

Marco Sentinelli, L'usabilità dei nuovi media, Carocci,
Roma 2003

Prefazione	Errore. Il segnalibro non è definito.
1. Cos'è l'usabilità	5
La definizione di usabilità degli standard ISO.....	6
La definizione di usabilità dei diversi autori	6
L'evoluzione del concetto di usabilità	9
Prima del 1980, l'usabilità non è necessaria.....	9
Anni '80, i laboratori di usabilità.....	9
Fine Anni '80, inizio Anni '90, design iterativo.....	10
Anni '90, design partecipativo	11
Il concetto di usabilità oggi	11
I tre settori dell'usabilità.....	12
L'architettura informativa.....	13
La progettazione dell'interfaccia grafica	13
L'analisi dei flussi interattivi.....	14
Principi del buon design.....	14
Fornire un buon modello concettuale del sistema	15
Rendere visibili le cose	15
Il mapping	16
Il feedback	16
Le caratteristiche del processo d'interazione	17
Usabilità e accettabilità	18
I 5 Aspetti della Usability	18
Learnability	19
Efficiency	21
Memorability	21
Error.....	22
Satisfaction.....	22
Categorie di utenti e differenze tra utenti.....	24
I principi di usabilità	26
1. Realizzare un dialogo semplice e naturale.....	26
2. Semplificare la struttura dei compiti.....	27
3. Favorire il riconoscimento piuttosto che il ricordo.....	28
4. Rendere visibile lo stato del sistema attraverso un feedback	29

5. Prevenire e limitare gli errori di interazione	30
6. Essere coerenti.....	31
7. Facilitare la flessibilità d'utilizzo e l'efficienza dell'utente.....	32
8. Fornire help e manuali.....	33
2. Storia delle interfacce.....	34
Batch Systems.....	35
Interfaccia riga di comando	35
Interfacce full screen	36
Gerarchia dei menu	37
Interfacce grafiche	37
Interfacce della prossima generazione	39
Trend di lungo termine.....	40
3. Come: le tecniche dell'usabilità.....	42
Il processo user centered	42
Le attività del processo user centered	44
Conoscenza degli utenti	44
Analisi comparativa	44
Definizione dei requisiti di usabilità	44
Progettazione parallela	45
Prototipazione.....	45
Validazione di usabilità attraverso test.....	45
Indagine follow-up	45
Valutazione di usabilità: metodi e metriche.....	46
Heuristic evaluation	46
Cognitive walkthrough	47
Pluralistic walkthrough.....	48
Thinking aloud	48
Task analysis	49
Metriche e misure	50
Analisi qualitativa vs misurazione quantitativa.....	52
Flusso di procedura	53
4. Dove: le applicazioni dell'usabilità	59
Prodotti informativi	59
Prodotti emozionali	59
Manifesto della usability.....	60
5. Perché applicare i processi di usabilità.....	61

I 10 slogan dell'usabilità	61
La tua migliore ipotesi non è la migliore.....	61
Gli utenti hanno sempre ragione	61
Gli utenti non hanno sempre ragione	61
Gli utenti non sono designer	62
I designer non sono utenti.....	62
Gli amministratori delegati non sono utenti	62
Meno è più	63
Problemi di dettaglio	63
L'aiuto non aiuta	63
Il progetto dell'usabilità è un processo	64
Vantaggi e costi	64
Il paradosso dell'usabilità in azienda	65
L'usabilità sul Web non è un bene di lusso	65
Usability: reason why.....	66
Ma quali sono i vantaggi che derivano dall'usabilità?	67
6. Chi: le nuove figure professionali.....	69
I diversi ruoli.....	69
Test monitor	69
Data logger	69
Timer	69
Operatore video	70
Esperto.....	70
Ruoli aggiuntivi	70
Osservatori.....	70
Ruolo del test monitor	71
Le competenze del test monitor	71
Specialista di fattori umani	71
Specialista di marketing	71
Comunicatori tecnici	71
Rotazione dei ruoli.....	72
Consulenti esterni.....	72
Caratteristiche di un buon test monitor	72
Formazione di base per la usability engineering.....	72
Apprendere facilmente.....	72
Capacità di instaurare ottimi rapporti	72
Eccellente memoria	73
Ottimi ascoltatori.....	73
A proprio agio con l'ambiguità.....	73

Flessibilità.....	73
Capacità di concentrazione	74
Capacità di dare sguardi generali.....	74
Capacità comunicative	74
Capacità organizzative e coordinative.....	74
Tipici problemi del test monitor.....	74
Condurre piuttosto che disporre.....	74
Troppo coinvolto nella registrazione dei dati.....	75
Comportarsi come ultra-competente.....	75
Troppo rigido con il test plan	75
Incapace di relazionarsi con ogni partecipante.....	75
Saltare alle conclusioni	75
Gestire lo stress	76
Come migliorare le abilità di un test monitor	76
Imparare i principi base dei fattori umani e dell'ergonomia.....	77
Imparare osservando gli altri.....	77
Osservarsi nella ripresa.....	77
Pratica	77
Abituarsi alla consapevolezza.....	77
Selezionare i partecipanti al test.....	77
Decidere il numero dei partecipanti al test	78
Qualche parola sull'uso di utenti interni	79
Per contattare gli utenti	80
7. I laboratori di test dell'usabilità.....	81
La zona dei test.....	81
Stanza singola	81
Stanza singola modificata.....	83
Stanza di osservazione elettronica	85
Laboratorio classico.....	86
Laboratorio classico con specchio unidirezionale.....	87
Laboratorio mobile	88
Conclusione	88
Bibliografia	90
Risorse web.....	95

1. Cos'è l'usabilità

Per comprendere cosa sia l'usabilità di un prodotto interattivo, partiamo da un'esperienza molto comune.

Immaginiamo di doverci sedere su una sedia per lavorare al nostro pc: con questo abbiamo già definito i concetti chiave di utente (noi stessi), prodotto con cui interagire (sedia), bisogno (il sedersi), contesto (postazione da lavoro con computer).

L'azione di sedersi sulla sedia non ci occuperà più di qualche decimo di secondo se il prodotto sarà riconoscibile per i suoi aspetti fondanti (seduta, zampe, eventuale schienale), cioè sarà cognitivamente chiaro per l'utente.

Se il prodotto non sarà in grado di comunicare la propria struttura all'utente, l'utente sarà chiamato a svolgere uno sforzo di riconoscimento e fruizione che nei casi estremi potrebbe costringerlo ad abbandonare il prodotto.

Il prodotto inoltre dovrà essere in grado di soddisfare i bisogni principali (il sedersi) e secondari (assicurare una posizione comoda ed efficace per l'uso del pc, essere esteticamente piacevole, essere provvisto di braccioli e rotelle...) dell'utente.

Il prodotto dovrà essere anche coerente con il contesto d'utilizzo: immaginiamo di affrontare un viaggio in nave stando seduti una sedia con rotelle o di guidare un'automobile seduti su uno sgabello.

Infine il prodotto dovrà essere adeguato alle caratteristiche di specifici utenti: ad esempio una persona molto alta, in aereo avrà bisogno di una poltrona con molto spazio davanti a sé.

In sintesi, per essere usabile un prodotto interattivo dovrà:

- essere adeguato ai bisogni e alle aspettative di specifici utenti che lo utilizzano in specifici contesti d'uso;
- risultare facile da capire, da imparare, da usare ed essere gradevole;
- consentire di eseguire le specifiche attività lavorative in maniera corretta, veloce e con soddisfazione;
- generare pochi errori non critici.

Ma l'usabilità è davvero un valore fondamentale?

Per rispondere a questa domanda, ritorniamo alla nostra sedia e interveniamo su di essa, capovolgendola con la seduta a terra e la zampe in aria: il prodotto sarà fisicamente lo stesso, composto dagli stessi elementi (seduta, schienale e zampe), ma

non adeguato ai bisogni degli utenti perché di fatto non usabile...a meno di un certo slancio di masochismo...

Quindi: un prodotto non usabile sarà un prodotto che di fatto non esisterà; un costo inutile per il produttore e per l'utente, che ne sceglierà un altro e percepirà un'immagine negativa non solo del prodotto ma anche e soprattutto del produttore.

La definizione di usabilità degli standard ISO

Lo standard ISO/IEC 9126 "*Information technology - Software product evaluation - Quality characteristics and guidelines for their use*"¹ definisce l'usabilità come "la capacità del *software* di essere compreso, appreso, usato e gradito dall'utente quando usato in determinate condizioni".

Lo standard ISO 9241-11 "*Ergonomic requirements for office work with visual display terminals - Guidance on usability*"² invece, definisce l'usabilità come "il grado in cui un prodotto può essere usato da classi di utenti per raggiungere specifici obiettivi con efficacia, efficienza e soddisfazione in un contesto d'uso determinato".

La definizione di usabilità dei diversi autori

I diversi autori che si sono occupati di chiarire il concetto di usabilità si sono trovati in difficoltà a tracciarne i confini.

Attualmente, l'autore che sembra incontrare i favori della comunità della HCI (*Human Computer Interaction*) è Shackel³, secondo il quale l'usabilità di un artefatto consiste ne "la sua capacità, in termini di caratteristiche cognitive umane, di essere utilizzato facilmente ed efficacemente da una specifica categoria di utenti, posto uno specifico esercizio e supporto all'utente, per svolgere specifiche categorie di compiti, all'interno di specifici scenari ambientali".

Kunkel, Bannert e Fach⁴, affermano che il contesto generale dell'usabilità abbraccia le quattro principali componenti di una situazione di lavoro: utente, compito, sistema ed ambiente. Progettare in vista dell'usabilità significa riuscire ad armonizzare l'interazione di questi quattro fattori.

1 ISO/IEC 9126 *Information technology - Software product evaluation - Quality characteristics and guidelines for their use*, 1992-1998.

2 ISO 9241 *Ergonomic Requirements for Office Work with Visual Display Terminals, part 11, Guidance on specifying and measuring usability*, 1995.

3 MORRIS, G., DILLON A., *The importance of usability in the establishment of organizational software standards for end user computing*. International Journal of Human - Computer Studies, 1996, vol. 45, 243-258.

4 KUNKEL K., BANNERT M., FACH W., *The influence of design decisions on the usability of direct manipulation user interface*. Behaviour and Information Technology, 1995, vol. 14, no. 2, 93-106.

Un modo più semplice per definire il concetto di usabilità ci viene fornito da Preece⁵ il quale afferma che "gli obiettivi della *Human Computer Interaction* sono quelli di sviluppare e migliorare sistemi che prevedano l'utilizzo di calcolatori in modo che gli utenti possano svolgere i loro compiti senza problemi, efficacemente, efficientemente e apprezzando il loro strumento di lavoro. Questi quattro fattori, assieme, compongono il concetto di usabilità."

Secondo Shneiderman⁶, nel concetto di usabilità vanno identificate quattro dimensioni principali: l'efficienza (*efficiency*), la facilità di apprendimento (*learnability*), la facilità di ricordare i comandi principali (*memorability*), la soddisfazione nell'uso (*satisfaction*). Il concetto di usabilità, inoltre, va considerato prettamente pratico, e la sua analisi deve fornire linee guida operative per la progettazione. Infatti, al centro del concetto di usabilità c'è la consapevolezza che ogni alternativa di progettazione deve essere valutata il più presto possibile con gli utenti potenziali del prodotto stesso. L'obiettivo della valutazione è quello di assicurare che i prodotti *software* siano caratterizzati da: brevi tempi di apprendimento, rapida esecuzione dei compiti, basso tasso di errore, facilità nel ricordare le istruzioni di base, alta soddisfazione dell'utente.

Anche Morris e Dillon⁷ sono consapevoli della difficoltà di definire il concetto di usabilità: essa viene spesso identificata solamente come un attributo dell'interfaccia, piuttosto che come una qualità propria del prodotto considerato nella sua totalità. Se così fosse, sarebbe sufficiente seguire delle linee guida valide per ogni prodotto, per ogni ambiente, per ogni contesto; al contrario, seguire dei principi di questo tipo non garantisce l'usabilità del prodotto. Anche Morris e Dillon riportano la definizione di Shackel, apprezzandone l'attenzione per il fattore umano e per l'ambiente cui un prodotto è destinato, anziché per la presenza o assenza di questa o quella opzione.

Jakob Nielsen¹, uno degli autori di riferimento per l'usabilità del Web, definisce l'usabilità come la misura della qualità dell'esperienza dell'utente in interazione con qualcosa, sia esso un sito Web o un applicazione *software* tradizionale o qualsiasi altro strumento con il quale l'utente può operare. Secondo Nielsen, un prodotto è usabile quando è facile da apprendere, consente una efficienza di utilizzo, è facile da ricordare, permette pochi errori di interazione e di bassa gravità, è piacevole da usare.

5 LIN X., CHOONG, Y., SALVENDY G., *A proposed Index of usability: a method for comparing the relative usability of different software systems*. Behaviour and Information Technology, 1997, vol. 16, n. 4/5, 267-278.

6 SHNEIDERMAN, B., *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Addison-Wesley Publishing Co Reading, MA, 1992.

7 MORRIS, G., DILLON A., *The importance of usability in the establishment of organizational software standards for end user computing*. International Journal of Human - Computer Studies, 1996, vol. 45, 243-258.

1 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

Concettualmente, l'usabilità di un prodotto, e più specificamente di un prodotto *software*, misura la distanza cognitiva fra il modello del progettista (modello del prodotto e delle sue modalità d'uso, possedute dal progettista ed incorporate nel prodotto) e il modello dell'utente (modello di funzionamento del prodotto che l'utente si costruisce e che regola l'interazione col prodotto): quanto più i due modelli sono vicini, tanto meno l'usabilità costituirà un problema⁸. L'usabilità va ricercata seguendo due principi fondamentali⁹:

- Compatibilità cognitiva uomo-computer: un'interfaccia deve essere non solo fisicamente compatibile con le caratteristiche della percezione e dell'azione umana, ma deve essere anche cognitivamente compatibile con le caratteristiche della comunicazione, della memoria e della soluzione di problemi umani. In questo senso, lo scopo del design orientato all'usabilità è quello di far incontrare il modello del sistema che l'utilizzatore forma e il modello dell'utilizzatore che il progettista incorpora nel sistema.

- Contesto del compito: l'azione degli esseri umani non può essere presa in considerazione senza fare riferimento al contesto generale in cui essa si svolge, poiché l'agire nasce, appunto, dall'incontro dell'attore umano con l'ambiente.

Obiettivo finale dell'usabilità è quello di rendere la tecnologia sottostante invisibile, trasparente all'utilizzatore, il quale deve potersi concentrare esclusivamente sul compito, anziché sul mezzo¹⁰.

In sintesi, quindi, per essere usabile, un prodotto deve:

essere adeguato ai bisogni e alle aspettative degli specifici utenti finali che lo usano in determinate condizioni;

risultare facile da capire, da imparare, da usare, ed essere gradevole;

consentire di eseguire le specifiche attività lavorative in modo corretto, veloce e con soddisfazione;

generare pochi errori non critici.

Affinchè un prodotto sia adeguato ai bisogni e alle aspettative degli utenti finali, occorre dunque conoscere bene le caratteristiche degli utenti, le attività che svolgono e il contesto organizzativo e sociale nel quale sono inseriti e operano.

8 NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

9 MANTOVANI G., *L'interazione Uomo - Computer*, il Mulino, Bologna, 1995.

10 NORMAN, D.A., *The invisible computer*, MIT Press Cambridge, MA 1998, tr. it. PARRELLA B., *Il computer invisibile*, Apogeo, Milano 2001.

L'evoluzione del concetto di usabilità

L'evoluzione del concetto di usabilità dei prodotti ipermediali è parallela a quella dei prodotti stessi; in particolare è in funzione del tipo di utenti dei sistemi. Possiamo individuare quattro epoche evolutive diverse¹¹:

Prima del 1980, l'usabilità non è necessaria.

Fino agli Anni '80, il *software* veniva prodotto e utilizzato prevalentemente dagli ingegneri informatici; gli utilizzatori di questi prodotti erano poi gli stessi progettisti: se l'utente di un sistema è la stessa persona che lo ha sviluppato, ovviamente l'usabilità è un problema che non esiste.

Inoltre l'attenzione era rivolta più al mezzo (il computer) che allo scopo (coadiuvare l'attività delle persone); in questo caso, il modello del progettista e il modello dell'utente coincidevano. In quegli anni, però, nacque la *Human Computer Interaction*¹² (HCI), una scienza fondata sul design che cercava di capire ed aiutare le persone a interagire con e per mezzo della tecnologia. In questo periodo, l'affermazione secondo cui i sistemi dovevano essere progettati tenendo conto delle esigenze, delle abilità e dei desideri degli utenti finali non veniva presa molto sul serio. La serie di studi che costituiscono la pietra miliare della fondazione dell'HCI venne allora definita Psicologia del *Software*. Obiettivo di questa disciplina era quello di provare l'utilità di un approccio comportamentale alla comprensione del design del *software*, della programmazione e dell'uso dei sistemi interattivi, e di motivare gli sviluppatori di sistema a tenere in considerazione le caratteristiche degli esseri umani. Obiettivo della psicologia del *software* era quello di produrre una descrizione generale dell'interazione degli esseri umani con il computer che fosse riassumibile in linee guida per gli sviluppatori e quello di verificare direttamente l'usabilità dei sistemi e del *software*.

Anni '80, i laboratori di usabilità.

Nel 1983, la Apple, realizza il primo *personal computer* con interfaccia grafica e *mouse*, destinato alla diffusione su larga scala. Da questo momento inizia l'introduzione sempre crescente del computer (con interfaccia grafica o a riga di comando) e contemporaneamente iniziano i problemi di usabilità: l'utente del sistema non appartiene più alla stessa classe di chi lo ha sviluppato. In questa fase, che corrisponde alla prima diffusione delle tecnologie informatiche sia negli uffici che nelle famiglie, gli utenti non avevano più competenze comuni con i progettisti. L'usabilità cominciava ad essere un problema e si verificarono i primi episodi di rifiuto della nuova

11 ANCeschi, G., (a cura di), *Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali*, Domus Academy, Milano, 1992-1993.

12 ANCeschi, G., (a cura di), *Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali*, Domus Academy, Milano, 1992-1993.

tecnologia, causati sia dagli alti costi per la formazione, sia dai mancati successi di molte esperienze di automazione di uffici. Si impose l'urgenza di avvicinare i due mondi, quello del progettista e quello dell'utente e l'usabilità, a partire dalla seconda metà degli anni '80, divenne l'obiettivo principale della HCI¹³. Per andare incontro a queste nuove esigenze, da un lato vennero introdotte nella progettazione linee guida sul fattore umano, dall'altro vennero allestiti i primi laboratori di usabilità. Obiettivo principale di un laboratorio di usabilità era quello di testare i prodotti con utenti potenziali, prima del lancio commerciale. Lo studio dell'usabilità assunse un carattere prettamente empirico¹⁴: promosse l'uso di prototipi, la partecipazione degli utenti alla fase di progettazione, enfatizzò l'uso di metafore per presentare le nuove funzionalità offerte dal sistema. Questi studi diedero l'impulso alla creazione di linee guida sulla base delle quali la progettazione di un'interfaccia poteva garantire al prodotto finale i requisiti di usabilità.

Il metodo utilizzato veniva definito design iterativo e consisteva nella realizzazione di prototipi sui quali venivano condotti test di usabilità, che a loro volta indicavano le modifiche da effettuare per migliorare il progetto. Questo nuovo approccio divenne ben presto largamente accettato e in molti si convinsero della necessità di progettare mediante l'utilizzo di prototipi; aumentò la partecipazione degli utenti alla progettazione e il metodo della creazione rapida di prototipi divenne, di fatto, uno standard. In considerazione di questo approccio pratico alla soluzione dei problemi, si cominciò a parlare di Ingegneria dell'Usabilità¹⁵.

Fine Anni '80, inizio Anni '90, design iterativo

L'Ingegneria dell'Usabilità introdusse tre nozioni chiave¹⁶:

- venne proposto che la progettazione iterativa venisse condotta inseguendo obiettivi pratici definiti e misurabili, detti specifiche di usabilità;
- si cercò di ampliare il campo d'azione degli studi empirici e delle tecniche di design cooperativo, per arrivare a definire un nuovo approccio, detto anche design contestuale, che tenesse conto sempre più dei reali bisogni dell'utente e del contesto reale in cui il prodotto veniva utilizzato, data la sterilità dei risultati ottenuti in laboratorio in condizioni decontestualizzate;
- la terza nozione chiave dell'ingegneria dell'usabilità divenne il rapporto costo/ricavo; infatti, uno dei problemi maggiori dei metodi di progettazione iterativa consisteva appunto negli alti costi che i vari cicli di riprogettazione richiedevano.

13 NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

14 NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993.

15 Ibidem.

16 Ibidem.

Nel 1986 viene pubblicato il libro di Norman e Draper "*User centered system design: New perspective on human-computer interaction*"¹⁷. Ci si rende conto che l'utente non può essere preso come un singolo individuo, ma visto come soggetto che appartiene a una cultura e a una organizzazione. Il modo stesso di utilizzare le tecnologie è influenzato dallo specifico contesto all'interno del quale le tecnologie si trovano. Per questi motivi l'introduzione di linee-guida sul fattore umano o la definizione di specifiche generiche non poteva portare alla reale usabilità dei prodotti. Con il modello *user centered*¹⁸, che inizia ad affermarsi su larga scala alla fine degli Anni '80, si riconosce l'importanza non solo delle capacità e dei vincoli fisici e cognitivi dei singoli utenti, ma anche delle relazioni culturali, sociali e organizzative, nonché degli artefatti cognitivi distribuiti nell'ambiente che influenzano il modo di lavorare dell'uomo.

Anni '90, design partecipativo

La consapevolezza dei costi e degli sforzi di formazione, nonché l'alto numero di errori generati dall'interazione, costringe gli sviluppatori a mettere al centro della progettazione l'utente e le sue esigenze. Si passa perciò dal diretto coinvolgimento degli specialisti al diretto coinvolgimento degli utenti¹⁹. L'utente partecipa a tutte le fasi definitorie del processo assumendo il ruolo di corresponsabile, insieme con il progettista, del prodotto finito. La produzione del *software* non è più un processo lineare ma un processo iterativo in cui si perviene al risultato finale attraverso aggiustamenti successivi guidati dalla continua verifica delle esigenze e delle necessità dell'utente finale.

Il concetto di usabilità oggi

Attualmente il concetto di usabilità sta subendo delle trasformazioni dovute all'ingresso nella discussione sull' HCI di figure nuove, provenienti dall'antropologia e dalle scienze sociologiche, che spingono l'attenzione verso orizzonti più ampi, ai confini con l'etnografia. Sono emersi problemi nuovi, legati non più all'aspetto tecnico ma alle implicazioni di carattere sociale, alle relazioni di potere tra utilizzatori di status diverso e al pericolo che l'exasperazione dell'usabilità possa portare a un impoverimento dei compiti e delle qualifiche richieste ai lavoratori. Ciò implica un

17 NORMAN, D. A., DRAPER, S., *User Centered System Design: New Perspectives on Human-Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale NJ, 1986.

18 NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993.

19 Ibidem.

mutamento della definizione di utente da fattore umano (tradizionalmente inteso come agente passivo e spersonalizzato), ad attore umano, vale a dire un individuo attivo, capace di controllo e scelta. L'esigenza diventa, perciò, quella di abbandonare la ricerca di tecniche e linee guida valide per tutti, in qualsiasi occasione e di accettare di confrontarsi con la complessità che deriva dalla progettazione di strumenti dedicati a persone diverse tra loro, ognuna impegnata nei propri scopi e immersa nel proprio ambiente²⁰.

Attualmente il punto di riferimento della comunità che si occupa degli studi di usabilità per i prodotti ipermediali è Jakob Nielsen. L'approccio di Nielsen è però oggetto di critiche provenienti da più parti: l'opinione dei designer di interfacce²¹ è che mettere l'usabilità al centro di tutti i prodotti, indipendentemente dal loro scopo, porterebbe a un rischio di omologazione delle interfacce che limiterebbe la creatività. Altra accusa mossa a Nielsen è quella di un approccio integralista che lo porta a scagliarsi contro gli stessi *tool* di sviluppo di prodotti ipermediali.

I tre settori dell'usabilità

L'usabilità di un prodotto ipermediale, racchiude tre grandi settori di indagine ed analisi²²:

- l'architettura informativa;
- la progettazione dell'interfaccia grafica;
- l'analisi dei flussi interattivi.

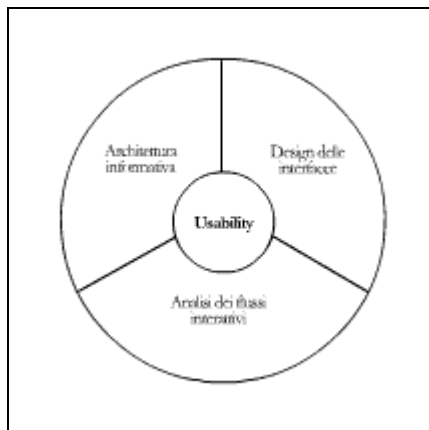


FIGURA 1 I tre settori dell'usabilità.

²⁰ RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

²¹ <http://www.k10k.net> e <http://www.flashkit.com>, i due siti di riferimento della comunità di sviluppatori Flash si sono scagliati più volte contro Nielsen e il suo concetto di usabilità.

²² RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

L'architettura informativa²³

L'architettura informativa di un prodotto multimediale interattivo è funzionale ad alcuni obiettivi specifici:

chiarire la *mission* e la *vision* del prodotto, bilanciando le esigenze della produzione con quelle degli utenti;

determinare quali contenuti e funzioni deve avere il prodotto;

specificare come gli utenti saranno in grado di raggiungere le informazioni all'interno del prodotto, definendone l'organizzazione, la navigazione, il *naming* e il sistema di ricerca;

prevedere come il prodotto possa evolvere e crescere nel corso del tempo.

Tutto questo può sembrare ovvio; probabilmente lo è. Rimane il fatto che gli utenti notano l'architettura informativa di un sistema solo quando essa effettivamente funziona male; quando non la notano significa che si sposa perfettamente con la loro mappa concettuale: il prodotto è quindi ben strutturato.

L'architetto informativo/informazionale è perciò la persona che:

organizza i percorsi informativi in dati, rendendo chiara la complessità;

disegna la mappa informativa che permette agli utenti di trovare il proprio percorso personale verso la conoscenza;

soddisfa le esigenze di una società centrata e concentrata sull'organizzazione dell'informazione e sulla comunicazione.

La progettazione dell'interfaccia grafica²⁴

Il design di un prodotto informatico interattivo deve tenere conto non solo dell'estetica dell'interfaccia, ma anche e soprattutto delle regole grafiche di usabilità. L'uso dei colori, delle forme, dei *font*, delle animazioni deve tenere conto dell'utente: il designer non deve disegnare per sé ma deve disegnare per l'utente. L'aumentata capacità di calcolo dei sistemi informatici e la sempre più ampia larghezza di banda disponibile per i sistemi in rete lasciano molto più spazio alla creatività dei designer. Sempre più spesso però assistiamo a delle vere e proprie "indigestioni multimediali": *home page* di siti Web colme di immagini, loghi, animazioni e testi, collocati insieme senza alcuna gerarchia informativa, con l'unico risultato di disorientare l'utente.

Dall'altro lato, una buona interfaccia grafica non è sempre graficamente essenziale: si tratta solamente di un'interfaccia ordinata, nella quale ogni elemento grafico ha il suo posto in funzione della gerarchia percettiva riservata.

²³ ROSENFELD, L., MORVILLE, P., *Information Architecture for the World Wide Web*, O'Reilly, Sebastapol CA, 1998.

²⁴ LAUREL, B., *The Art of Human-Computer Interface Design*, Addison-Wesley, Reading, MA, 1990.

Nella progettazione dell'interfaccia grafica convogliano più competenze e discipline di studio: dal design alla psicologia percettiva, dallo studio dei linguaggi e tool di sviluppo come Html e Flash, alla tipografia per lo studio dei *font*.

L'analisi dei flussi interattivi

Lo studio delle interazioni uomo-macchina collega l'usabilità con altre discipline, l'ergonomia e la HCI (*Human Computer Interaction*).²⁵

L'ergonomia è lo studio dell'interazione uomo-strumenti-ambiente nel suo complesso. La Società Italiana di Ergonomia definisce l'ergonomia come "un corpus di conoscenze interdisciplinari in grado di analizzare, progettare e valutare sistemi semplici o complessi in cui l'uomo figura come operatore o utente. Persegue coerenza e compatibilità tra il mondo che ci circonda - oggetti, servizi, ambienti di vita e di lavoro - ed esigenze di natura psicofisica e sociale, anche con l'obiettivo di migliorare l'efficienza e l'affidabilità dei sistemi".²⁶

Molto simile a quella di ergonomia è la definizione di HCI che si ritrova nel capitolo 2 del Curricula for HCI dello Special Interest Group on Computer-Human Interaction (SIGCHI): "L'interazione uomo-calcolatore è una disciplina che riguarda la progettazione, la valutazione e l'implementazione di sistemi interattivi per l'uso da parte degli esseri umani e lo studio dei più importanti fenomeni ad essi collegati"²⁷.

La differenza tra l'ergonomia cognitiva e HCI riguarda l'ambito di intervento. Infatti, mentre l'ergonomia spazia su oggetti, servizi, ambienti di vita e di lavoro, l'HCI si focalizza principalmente sui sistemi ipermediali²⁸.

Per quanto riguarda l'usabilità, l'analisi dei flussi interattivi si svolge attraverso l'applicazione del processo *user centered design* all'interno del ciclo di sviluppo del prodotto ipermediale, con varie tecniche di indagine e analisi.

Principi del buon design²⁹

A queste domande conseguono i principi del buon design, validi per tutti i prodotti interattivi: buon modello concettuale, visibilità, *mapping*, presenza del *feedback*.

25 MANTOVANI G., *L'interazione Uomo - Computer*, il Mulino, Bologna, 1995.

26 Il sito della Società Italiana di Ergonomia è raggiungibile on line all'indirizzo <http://www.informatique.it/sic/lecht.html>

27 Curricula for Human Computer Interaction, a cura della Association for Computer Machinery, Acm, New York, 1992. Il Curricula è consultabile on line all'indirizzo <http://www.acm.org/sigchi/cdg/cdg2.html>

28 MANTOVANI G., *L'interazione Uomo - Computer*. il Mulino, Bologna, 1995.

29 NORMAN, D.A., *The invisible computer*, MIT Press Cambridge, MA 1998, tr. it. PARRELLA B., *Il computer invisibile*, Apogeo, Milano 2001.

Fornire un buon modello concettuale del sistema

Un buon modello concettuale permette di prevedere gli effetti delle azioni. Senza un modello adeguato si può operare meccanicamente, alla cieca. Finché le cose funzionano come si deve, non si hanno problemi. Ma quando sorge un imprevisto o una situazione nuova, ecco che si ha bisogno di una migliore comprensione: un buon modello concettuale del sistema.

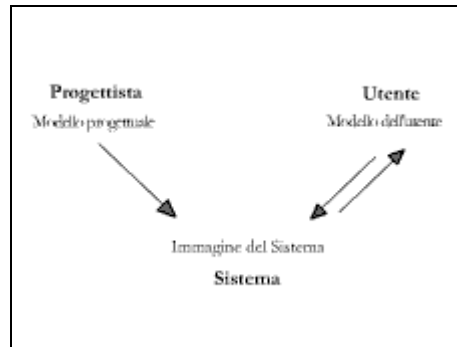


FIGURA 2 Modello progettuale e modello dell'utente. *Tratto da* NORMAN, D.A., *The invisible computer*, MIT Press Cambridge, MA 1998, tr. it. PARRELLA B. *Il computer invisibile*, Apogeo, Milano 2001, p.184-191.

Il modello progettuale è il modello concettuale del progettista. Il modello dell'utente è il modello mentale sviluppato attraverso l'interazione con il sistema. L'immagine del sistema risulta dalla struttura fisica che è stata costruita, la parte visibile del dispositivo. Il progettista si aspetta che il modello dell'utente sia identico al modello progettuale. Ma il progettista, anche se dovrebbe, non parla direttamente con l'utente; tutta la comunicazione avviene attraverso l'immagine del sistema. Se l'immagine del sistema non rende chiaro e coerente il modello progettuale, l'utente finirà per formarsi un modello mentale sbagliato.

Rendere visibili le cose

La visibilità funge da efficace richiamo mnemonico di ciò che si può fare e permette al comando stesso di specificare come deve essere eseguita l'azione. La relazione chiara e motivata fra la posizione del comando e la funzione cui assolve rende facile trovare il comando giusto per la manovra da eseguire. Di conseguenza, non c'è molto da ricordare.

I problemi causati dall'insufficiente attenzione alla visibilità sono illustrati da un apparecchio semplicissimo: il telefono moderno. Le corrispondenze sono arbitrarie: non c'è senso alcuno nella relazione fra le azioni da eseguire e i risultati che si ottengono; ad esempio, per accedere alla casella vocale "digitare cancelletto, quattro,

cinque, sei, tre, di seguito il proprio codice segreto e infine cancelletto”. Perché quattro, cinque, sei, tre e non un’altra sequenza numerica?

I rapporti fra le intenzioni dell’utente, le azioni richieste e i risultati finali sono completamente arbitrari. I comandi hanno funzioni multiple. Non c’è un’adeguata informazione di ritorno, così l’utente non è mai certo di aver ottenuto il risultato sperato. Il sistema, in generale, non è comprensibile; le sue capacità non sono evidenti. Ogni volta che il numero delle azioni possibili eccede il numero dei comandi, è facile che ci siano difficoltà.

Il mapping

Mapping è un termine tecnico usato per indicare la relazione fra due cose, in questo caso fra i comandi e il loro azionamento e i risultati che ne derivano nel mondo esterno. Un *mapping* naturale, che cioè sfrutta analogie fisiche o modelli culturali, porta alla comprensione immediata. Per esempio, il progettista può utilizzare l’analogia spaziale: per sollevare un oggetto, muovere il comando verso l’alto. Alcune di queste correlazioni naturali sono di natura culturale, convenzionale o biologica, come il modello universale secondo cui un livello che sale rappresenta più. Altre correlazioni derivano dai principi della percezione: ad esempio il blu è percepito come un colore di sfondo, il rosso come un colore di primo piano; l’occhio è attirato dal movimento e nota prima la luminosità di un oggetto che la sua tinta.

Il feedback

Il *feedback*, l’informazione di ritorno che dice all’utente quale azione ha effettivamente eseguito e quale risultato si è realizzato, è un concetto ben noto nella cibernetica e nella teoria dell’informazione. Immaginate di cercare di parlare a qualcuno senza poter udire la vostra voce o di disegnare con una matita che non lascia segni: non ci sarebbe nessun *feedback* e, ad esempio, non sareste in grado di regolare il vostro volume di voce o controllare la correttezza del tratto.

Un dispositivo è facile da usare quando c’è visibilità dell’insieme di azioni possibili, quando i quadri di comando e controllo sfruttano correlazioni naturali, quando c’è un’informazione di ritorno che tiene al corrente l’utente dello stato del sistema.

Le caratteristiche del processo d'interazione³⁰

Nel nostro Paese stiamo assistendo proprio in questi anni a un'esplosione della *net economy*. Gli investimenti del settore IT sono in continuo aumento. Un utente medio trova sempre più spesso occasioni di interazione con sistemi ipermediali: dal telefono cellulare al sito Web per l'*home banking*, dallo *shopping on line* alla consultazione di Cd-rom multimediali.

Tutti questi oggetti sono caratterizzati da un processo di interazione uomo-macchina: da un lato l'utente, dall'altro lato l'interfaccia del sistema che media il rapporto con la macchina.

Moltiplicandosi le occasioni di interazione con questi sistemi, necessariamente diventa importante assicurare all'utente un processo con questi sistemi, facile, potente, soddisfacente ed efficiente³¹.

Un processo **facile** perché, soprattutto per certi sistemi con un uso sporadico e casuale, l'utente non deve perdere troppo tempo a capirne il funzionamento, ma deve concentrarsi sul proprio compito; quando si vuole scrivere un documento con un programma di elaborazione testi, ci si deve concentrare sul contenuto del documento e non si possono perdere risorse preziose a capire come utilizzare il programma: in questo caso si tornerebbe a scrivere con la penna.

Un'interazione **potente** ovvero con alta potenzialità: usare un sistema per eseguire molti compiti. Non è pratico avere un telefono cellulare che non abbia la possibilità di memorizzare i numeri telefonici: si avrebbe la necessità di un altro sistema (dalla semplice agenda, fino al computer palmare) da imparare e utilizzare.

Un processo **soddisfacente ed efficiente**: l'utente non deve spendere troppe risorse per raggiungere il proprio obiettivo; e queste risorse devono essere spese per assicurare all'utente un'interazione piacevole. Per effettuare un acquisto *on line*, non si dovrebbe essere costretti a riempire *form* e a rispondere a domande per molti minuti: è un'esperienza frustrante che obbligherebbe a sprecare troppo tempo in funzione del risultato da ottenere; si potrebbe preferire spendere queste risorse in altro modo, ad esempio nel traffico, per effettuare l'acquisto di persona.

30 COOPER, A., *About Face: The Essentials of User Interface Design*, IDG Books, Foster City, CA, 1995; FLEMING, J., *Web Navigation: Designing the User Experience*, Songline Studios, Sebastopol CA, 1998; LEVIALDI, S., *Interazione Uomo Macchina, dispense del corso*, Roma, 2000; NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

31 NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

Usabilità e accettabilità³²

L'usabilità è un concetto molto più specifico di quello di accettabilità di un sistema. L'accettabilità di un sistema racchiude sia la sua dimensione sociale (per esempio, per quanto valido, un sistema che permetta di usare carte di credito rubate non è socialmente accettabile), sia la sua dimensione pratica.

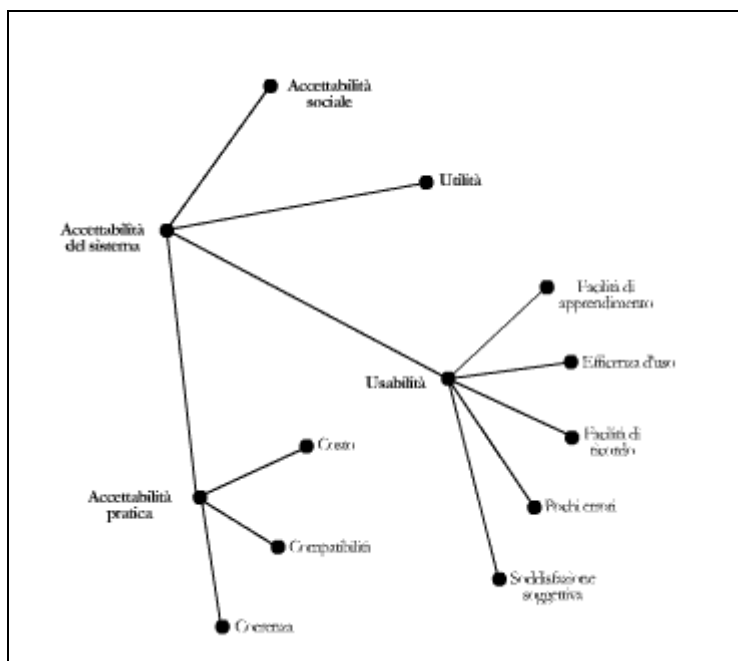


FIGURA 3 Accettabilità di un sistema. Tratto da NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993, p.25.

La dimensione pratica dell'accettabilità di un sistema è data da una serie di attributi (v. FIG. 9) quali il costo, la compatibilità, l'affidabilità, l'utilità. L'usabilità è una parte dell'accettabilità. Comprende ogni aspetto del sistema con cui l'utente potrebbe interagire, anche l'installazione e la manutenzione.

I 5 Aspetti della Usability

L'usabilità è una caratteristica pluridimensionale di un'interfaccia con una moltitudine di componenti tradizionalmente associati con 5 aspetti fondamentali³³.

1. *Learnability*: il sistema deve essere semplice da imparare in modo che l'utente possa velocemente iniziare a lavorarci.

³² NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993.

³³ NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993, p.26-37.

2. *Efficiency*: il sistema deve essere efficiente da utilizzare in modo che, una volta imparato, l'utente possa raggiungere un alto livello di produttività.
3. *Memorability*: il sistema deve essere facile da ricordare, in modo che l'utente casuale sia in grado di tornare a utilizzare il sistema anche dopo un lungo periodo di inutilizzo, senza la necessità di dover nuovamente imparare qualcosa.
0. *Error*: il sistema deve avere un basso livello di errori, in modo che gli utenti compiano solo pochi errori durante l'utilizzo. Deve essere sempre possibile tornare indietro velocemente da percorsi errati e ovviamente non devono esistere errori di percorso irreversibili.
0. *Satisfaction*: il sistema deve essere soddisfacente per l'utente che l'utilizza.

Individuare queste cinque dimensioni della *usability* permette un approccio più sistematico alla questione, e permette anche di effettuare delle misure quantitative e qualitative. Generalmente l'usabilità è misurata attraverso l'utilizzo di una serie di utenti (selezionati tra gli utenti potenziali) che svolgono differenti compiti prestabiliti. E' ovviamente importante notare come la *usability* possa essere misurata per certe categorie di utenti e per certi compiti specifici: ad esempio, un utente che con un programma di elaborazione testi voglia scrivere una lettera avrà bisogno di un'interfaccia diversa da chi debba scrivere un saggio di centinaia e centinaia di pagine. Proprio per queste differenze fra utenti, sarebbe preferibile non solo prendere in considerazione la media dei risultati, ma l'intera distribuzione delle rilevazioni.

Learnability

Per certi aspetti questo aspetto è la caratteristica determinante della *usability*. Certamente esistono dei sistemi altamente specializzati per cui vale la pena utilizzare molte risorse per imparare un'interfaccia molto complessa e ricca, ma la maggior parte dei sistemi devono essere facili da imparare.

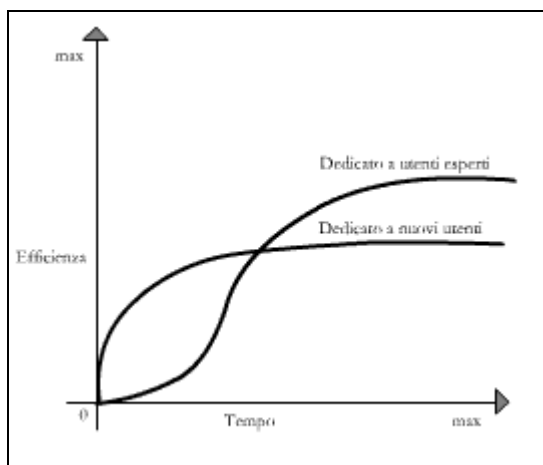


FIGURA 4 Rapporto tra efficienza e tempo di utilizzo per sistemi dedicati a nuovi utenti e a utenti esperti. Tratto da RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p. 128.

Come mostrato dalla FIG. 10, i sistemi con un'alta *learnability* permettono agli utenti, anche non esperti, di raggiungere un buon livello di utilizzo del sistema entro un periodo di tempo breve. Al contrario, i sistemi destinati a un'utenza avanzata all'inizio non presentano buoni risultati anche per gli stessi utenti esperti.

Quasi tutte le interfacce prevedono quindi un punto di partenza nel quale a tempo zero l'utente non è in grado di assolvere alcun compito³⁴. L'eccezione è costituita senz'altro da quei sistemi informativi (es. chioschi ipermediali di musei, sistemi di prenotazione e acquisto di biglietti ferroviari) che prevedono un tempo di *learnability* praticamente nullo, in grado di permettere all'utente di utilizzarle con successo già dalla prima volta. Questa curva standard della *learnability* non si applica nel caso in cui gli utenti trasferiscano il proprio bagaglio di conoscenze da versioni precedenti dello stesso sistema. In questo caso i tempi di apprendimento risulteranno essere molto più brevi.

La *learnability*, assieme alla soddisfazione individuale, è probabilmente la caratteristica di *usability* più semplice da misurare³⁵. Basta testare degli utenti che non hanno mai utilizzato il sistema e misurare quanto tempo impiegano a raggiungere un buon livello di confidenza con il sistema. Per il test vengono usati anche utenti che non hanno mai utilizzato un computer.

Nella curva di apprendimento sono rappresentate una serie continua di prestazioni utente migliorate e non una serie di dicotomie "appreso/ non appreso". Quindi si definisce un certo livello di prestazioni che l'utente deve raggiungere per passare dall'apprendimento del sistema all'utilizzo dello stesso.

³⁴ ANCESCHI, G., (a cura di), *Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali*, Domus Academy, Milano, 1992-1993.

³⁵ RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

Analizzando la *learnability* si deve comprendere che quasi mai l'utente si dedica al completo apprendimento dell'interfaccia prima del suo utilizzo³⁶. Al contrario, l'utente comincia a utilizzare il sistema non appena ha acquisito le competenze minime per farlo. Questo comporta la misurazione non di quanto tempo è necessario all'utente per avere una completa padronanza del sistema, quanto la minima competenza per poter svolgere un lavoro.

Efficiency

L'*efficiency* può essere definita come il livello di performance dell'utente, nel momento in cui la curva di *learnability* inizia ad appiattirsi³⁷. Ovviamente i tempi per il raggiungimento di questo livello variano in funzione dell'utente e del sistema. Per misurare l'efficiency di utenti esperti si deve quindi definire l'utente "esperto".

L'esperienza può essere definita in funzione della quantità di tempo spesa sul sistema o sul livello raggiunto rispetto la learning curve. Una tipica misura dell'*efficiency* è data ad esempio dalla scelta di un insieme rappresentativo di utenti esperti e la misura dei tempi mediamente necessari per il raggiungimento di un obiettivo.

Memorability³⁸

Gli utenti casuali sono la terza grande categoria di utenti, oltre a quelli nuovi e a quelli esperti. L'utente casuale utilizza il sistema in maniera intermittente e non frequente. Al contrario dei nuovi utenti, quelli casuali hanno però già utilizzato il sistema, e le loro azioni si basano sulle competenze acquisite nelle occasioni d'uso precedenti. Le applicazioni che si prestano a un utilizzo casuale sono tipicamente di supporto alle applicazioni principali che l'utente usa nel proprio lavoro quotidiano o che prevedono un utilizzo a intervalli di tempo molto lunghi.

Avere un'interfaccia semplice da comprendere e ricordare permette quindi agli utenti di tornare a utilizzare il sistema senza difficoltà anche dopo un periodo lungo di inutilizzo.

Una buona *memorability* è ovviamente favorita da un modello del sistema di facile comprensione. Questo attributo dell'usabilità è però quello che generalmente viene più trascurato dalle misurazioni. Due sono i metodi fondamentali per misurare la *memorability*: il più semplice consiste nel sottoporre, ad alcuni utenti che non utilizzano il sistema da tempo, alcuni compiti tipici, misurando il tempo necessario per il

36 Ibidem.

37 NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993; NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

38 NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993; NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

raggiungimento degli obiettivi. Altrimenti si può testare l'utente che ha finito una sessione di prova del sistema, chiedendo di ricordare i processi e i procedimenti necessari per svolgere alcune funzioni (ad esempio, spiegare le conseguenze di alcuni comandi).

I test di *memorability* devono però tenere conto di una caratteristica comune alla maggior parte delle interfacce: sono costruite sul principio di rendere visibile la gran parte del sistema agli utenti. Questo significa che l'utente può anche non ricordare i comandi decontestualizzati, ma è in grado perfettamente di assolvere i compiti e ricordare i procedimenti necessari una volta seduto di fronte al computer.

Error³⁹

Un errore può essere definito come un'azione che non raggiunge il fine desiderato; il tasso di errori indotto da un sistema può essere misurato come il numero di queste azioni che l'utente compie prima di raggiungere lo scopo. Il calcolo del tasso di errori può essere anche parte dell'esperimento condotto per misurare altre caratteristiche dell'usabilità.

Definire così l'errore non permette però di analizzare con attenzione i differenti impatti.

Alcune errori possono essere corretti rapidamente dagli utenti e hanno un minimo impatto sul ritmo di lavoro dell'utente.

Altri errori sono invece più catastrofici, sia perché non vengono scoperti dall'utente, conducendolo a un lavoro errato, sia perché distruggono lo stesso lavoro prodotto dall'utente, rendendo difficile il suo recupero. Questo tipo di errori va considerato separatamente dal primo tipo e il loro numero deve essere diminuito il più possibile.

Satisfaction⁴⁰

L'attributo finale della *usability* è la soddisfazione individuale, definibile come il piacere che l'utente ha di utilizzare il sistema. Questa caratteristica è particolarmente importante per quei prodotti che non hanno una destinazione di lavoro, ma per lo più di intrattenimento, come i videogiochi, o i programmi di *edutainment*. Per questo tipo di prodotti, il valore di intrattenimento è sicuramente superiore alle altre caratteristiche, poiché l'utente non è interessato al tempo necessario per assolvere un compito, dato che vuole spendere molto tempo nel suo intrattenimento.

Questa caratteristica individuale è molto differente dall'attitudine generale pubblica nei confronti del computer, che include oltre alla *usability* anche una considerazione dell'accettabilità sociale del sistema. Il livello di soddisfazione individuale può essere

³⁹ Ibidem.

⁴⁰ RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994..

misurato attraverso un approccio fisio-psicologico: misura della dilatazione delle pupille, livello del battito cardiaco, pressione sanguigna. Seppur valido da un punto di vista scientifico, questo tipo di test può essere falsato dall'effetto intimidatorio dell'approccio stesso. Alternativamente, la soddisfazione dell'utente può essere semplicemente misurata chiedendo agli utenti la loro soddisfazione⁴¹. Unendo le risposte di più utenti, si può avere una misura ragionevolmente accettabile della soddisfazione.

La misurazione prevede che venga sottoposto agli utenti, dopo la sessione di test, un breve questionario che preveda domande relative alla loro soddisfazione. E' fondamentale che per gli utenti nuovi il questionario venga sottoposto solo dopo che abbiano usato il sistema per assolvere alcuni compiti specifici. Poiché molti utenti rifiutano di utilizzare un programma perché il manuale è voluminoso, è buona norma che il sistema sia facilmente approcciabile. Per valutare questa caratteristica è quindi necessario porre questa domanda all'utente: "Quanto pensa sia difficile imparare a utilizzare questo programma?". Poiché la risposta a questa domanda è strettamente correlata con il grado di difficoltà incontrato dagli utenti nello svolgere i loro compiti, è preferibile che il sistema, piuttosto che con grandi potenzialità che prevedono uno sforzo per essere apprese, si presenti con meno potenzialità, ma mettendo così più raramente l'utente in condizioni di difficoltà. Generalmente i questionari per la misura della *subjective satisfaction* sono molto brevi, anche se ne sono stati sviluppati alcuni tipi lunghi. Normalmente vengono usati dei punteggi 1-5 o 1-7, con Scale Likert o scale di differenziale semantico⁴².

La scala Likert prevede alcune domande (es. "Ho trovato il sistema molto piacevole da utilizzare") e invita gli utenti a segnare il loro grado di accordo-disaccordo. Usando una scala da 1 a 5 avremo quindi: 1= fortemente in disaccordo, 2= parzialmente in disaccordo, 3= né d'accordo né in disaccordo, 4= parzialmente d'accordo, 5= fortemente d'accordo.

Al contrario, la scala del differenziale semantico prevede una lista di dicotomie (es. facile da utilizzare - difficile da utilizzare), e chiede all'utente di segnare la propria opinione come una posizione tra i due estremi.

Ovviamente si dovranno prevedere dei test pilota per verificare l'attendibilità delle domande utilizzate nelle scale e si dovranno prevedere anche una serie di domande a polarità invertita per ovviare alla tendenza degli utenti a dare sempre risposte troppo gentili.

41 Ibidem.

42 MARRADI, A., *Concetti e metodi per la ricerca sociale*, La Giuntina, Firenze, 1995.

Categorie di utenti e differenze tra utenti⁴³

Le due questioni più determinanti sull'usabilità sono le differenze fra i compiti e le caratteristiche individuali: una questione fondamentale per l'usabilità è dunque conoscere l'utente.

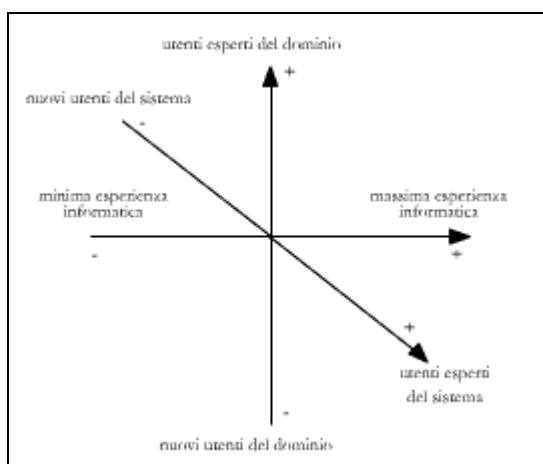


FIGURA 5 Le tre dimensioni del cubo utente: dominio, sistema, esperienza informatica. *Tratto da* NIELSEN, J., *Usability Engineering*. Academic Press, Boston, 1993, p44; LINDGAARD, G., *Usability Testing and System Evaluation: A Guide for Designing Useful Computer Systems*, Chapman and Hall, London, U.K., 1994.

La FIG. 11 mostra il cubo utente e le sue tre dimensioni fondamentali, secondo le quali gli utenti possono essere differenziati: esperienza con il sistema, con i computer in generale, e con i compiti specifici. L'esperienza dell'utente con il sistema porta alla differenziazione fra utenti esperti e utenti nuovi. La transizione tra le due categorie di utente segue il processo mostrato nella FIG. 11.

Oltre alla *learnability* del sistema, esistono altre caratteristiche che descrivono questo passaggio. Un esempio classico è quello di mostrare come parte dello stesso menu gli *shortcut* per i relativi comandi. Anche un buon sistema di aiuto in linea permette agli utenti di approfondire le proprie conoscenze.

L'esperienza dell'utente con i computer in generale ha un impatto sul design dell'interfaccia: ad esempio, un utente con esperienza di database e fogli elettronici cercherà anche nel suo *word-processor* il comando "ordina".

Sempre per differenziare gli utenti è possibile suddividere i menu in due livelli: il primo destinato agli utenti nuovi, con le funzioni principali; per gli utenti avanzati è

⁴³ NIELSEN, J., *Usability Engineering*. Academic Press, Boston, 1993, p44; LINDGAARD, G., *Usability Testing and System Evaluation: A Guide for Designing Useful Computer Systems*, Chapman and Hall, London, U.K. 1994, NIELSEN, J., *Designing User Interfaces for International Use*, Elsevier Science Publishers, Amsterdam, 1990.

invece prevista la possibilità di estendere il menu. Il sistema presenta così le proprie potenzialità, ben distinguendo quelle destinate agli utenti esperti dalle altre. C'è da considerare anche che i sistemi attuali prevedono una gamma così ampia di funzioni che, per un singolo utente, è possibile approfondire l'uso solo di alcune funzioni e quindi essere considerato avanzato solo per certi aspetti.

Infine anche la conoscenza da parte dell'utente dell'ambiente relativo ai compiti (*task domain*) incide sull'interfaccia: nel caso di una buona conoscenza sarà possibile usare una terminologia specifica nei menu e nei dialoghi e aumentare il carico informativo in una singola finestra. Si pensi all'interfaccia di un programma utilizzato per la gestione finanziaria: dipendentemente dalla conoscenza dell'utente del sistema finanziario vanno adeguati i dialoghi e le voci dei menu.

Esistono ovviamente anche fattori più generici per la differenziazione degli utenti: il sesso, l'età o anche fattori di più difficile osservazione come la capacità di ragionamento, di memorizzazione, di apprendimento...

E' poi importante anche la differente attitudine e motivazione: utenti che amano utilizzare il computer saranno maggiormente portati a sforzarsi per imparare l'utilizzo del programma. Questo gruppo di "super-utenti", che esplorano il sistema andandone a scovare gli aspetti più ignoti, svolge un ruolo fondamentale, permettendo agli sviluppatori di implementare funzioni nuove o semplicemente scoprire in anticipo esigenze nuove.

Come avevamo detto, un metodo per soddisfare le necessità di vari gruppi consiste nel prevedere un'ampia personalizzazione dell'interfaccia. Seppur valido, questo sistema deve essere usato con cautela, anche perché non è raro che un'interfaccia che soddisfi a pieno un gruppo crei notevoli problemi a un altro gruppo.

Dunque è fondamentale cercare di definire "tutti" i profili dei possibili utenti e rendere usabile l'interfaccia per tutti o la maggior parte di loro.

Tutte queste caratteristiche dell'usabilità spesso non possono essere soddisfatte contemporaneamente, poiché si tratta sempre di compromessi (ad esempio, volendo evitare degli errori catastrofici, si dovrà far passare l'utente per un numero maggiore di finestre di dialogo, diminuendone l'efficienza).

Nel caso in cui il compromesso possa essere trovato senza snaturare alcuna caratteristica del sistema, si procederà; altrimenti si dovranno scegliere le caratteristiche dell'usabilità fondamentali in funzione degli obiettivi del progetto di usabilità.

I principi di usabilità⁴⁴

Realizzare un dialogo semplice e naturale

Semplificare la struttura dei compiti

Favorire il riconoscimento piuttosto che il ricordo

1. Rendere visibile lo stato del sistema

Prevenire e limitare gli errori di interazione e facilitarne il recupero

Essere coerenti

Facilitare la flessibilità d'utilizzo e l'efficienza dell'utente

Fornire *help* e manuali

1. Realizzare un dialogo semplice e naturale

Come dicevamo, il sistema deve presentare un modello che l'utente sia facilmente in grado di capire, apprendere e utilizzare.

Il dialogo semplice e naturale è ovviamente basato sulla considerazione delle caratteristiche dell'utente, dei suoi compiti, della necessità di fornire un *feedback* soprattutto in relazione agli errori.

Il linguaggio utilizzato a livello di interfaccia deve essere semplice e familiare per l'utente e rispecchiare i concetti e la terminologia a lui noti.

Vanno evitati il più possibile un linguaggio tecnico e orientato al sistema che utenti non esperti di informatica possono non comprendere e termini stranieri: parole come *default*, *directory*, o frasi del tipo "il documento è disponibile in formato .rtf" o "documento .zip trasferibile via ftp".

Alcuni modi per rispettare il principio del dialogo semplice e naturale sono⁴⁵:

tenere conto dell'utente e del suo modello dell'attività nella:

organizzazione dei contenuti e della struttura del sistema,

implementazione della logica dell'interazione,

definizione dell'ordine dei menu,

definizione dell'ordine di presentazione delle finestre o delle pagine,

organizzazione dei contenuti e degli oggetti all'interno delle finestre o delle pagine;

fornire un buon modello concettuale del sistema e renderne evidenti l'organizzazione,

la logica d'interazione, affinché l'utente possa prevedere gli effetti delle proprie azioni:

interagire con un prodotto senza avere compreso e sviluppato un modello di come funziona è come agire alla cieca, senza sapere cosa succederà successivamente;

rendere evidenti le informazioni rilevanti di cui l'utente ha effettivamente bisogno per svolgere l'attività, evitando di presentare informazioni superflue o che si usano

⁴⁴ NIELSEN, J., *Usability Engineering*. Academic Press, Boston, 1993.

⁴⁵ Tratto da LAUREL, B., *The Art of Human-Computer Interface Design*, Addison-Wesley, Reading, MA 1990.

raramente: le informazioni irrilevanti possono entrare in "competizione" con quelle fondamentali e togliere a queste ultime la necessaria visibilità;
anticipare quanto possibile le informazioni che l'utente troverà nella pagina successiva o quello che succederà a seguito di una azione;
verificare con gli utenti la comprensibilità del linguaggio (etichette, istruzioni, elenchi);
usare preferibilmente icone e metafore già sperimentate e consolidate; se si vogliono usare simboli originali è sempre meglio sottoporli prima a test di comprensibilità con gli utenti finali del prodotto.

2. Semplificare la struttura dei compiti

I compiti o le attività che l'utente deve svolgere in interazione con il sistema devono avere una struttura semplice o devono essere semplificati.

Donald Norman⁴⁶ suggerisce quattro diversi approcci per semplificare i compiti:

mantenere il compito invariato, ma offrire sussidi mentali;

usare la tecnologia per rendere visibile quello che altrimenti sarebbe invisibile;

automatizzare, mantenendo il compito sostanzialmente invariato;

cambiare la natura del compito.

I primi tre approcci prevedono di non cambiare sostanzialmente il compito che l'utente deve affrontare.

Nel primo e nel secondo caso, il suggerimento è quello di offrire sussidi esterni (che supportano le capacità cognitive dell'utente per non costringerlo a ricordare tutto a mente) e *feedback* (che consentono all'utente di controllare le componenti non visibili del sistema, al fine di verificarne l'adeguato funzionamento). Ad esempio, nella scrittura a videoterminale, la struttura del compito è sostanzialmente identica alla scrittura manuale. In più la scrittura a videoterminale, grazie al sussidio del correttore automatico che segnala gli errori, consente di migliorare la qualità del lavoro.

Nel terzo caso, pur mantenendo invariata la struttura del compiti, alcune sue parti (quelle più rischiose o complicate) vengono eseguite dalla tecnologia e non più dall'utente.

L'ultimo approccio, quando il compito è intrinsecamente complesso per le abilità richieste, prevede una riprogettazione sostanziale del compito: l'obiettivo rimane lo stesso ma il modo in cui l'obiettivo viene raggiunto è totalmente diverso.

46 NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

3. Favorire il riconoscimento piuttosto che il ricordo

Osservando l'interfaccia, l'utente deve poter capire cosa deve fare, come può farlo e, una volta eseguita una azione, deve poter capire cosa è successo e quali sono stati i risultati.

Dal momento che è più facile riconoscere e ricordare una cosa vedendola direttamente, piuttosto che recuperare l'informazione dalla memoria, il modo più semplice per agevolare l'utente è quello di rendergli visibili le cose sull'interfaccia, ovvero fornirgli dei sussidi esterni che gli agevolano il ricordo. Un modo ulteriore (e forse il fondamentale), per facilitare il ricordo è dato dalla reale comprensione da parte dell'utente del modello concettuale del prodotto e della sua interazione.

Per facilitare la comprensione del modello è necessario sfruttare il "*mapping*" naturale, ovvero la correlazione naturale che esiste tra due cose, tra causa ed effetto, tra comandi, loro azionamento e risultati: un esempio di "*mapping*" naturale è la manipolazione diretta degli oggetti. Con queste tecniche, l'utente non è costretto a ricordare il modo di utilizzare degli oggetti o a descrivere le azioni da eseguire: semplicemente le esegue direttamente sullo schermo, per esempio spostando un documento dalla scrivania virtuale del suo PC al cestino, così come farebbe nel mondo reale.

Alcuni ostacoli al riconoscimento⁴⁷:

non capire, osservandolo, se un elemento (un campo per inserimento testo, un collegamento ipertestuale, un pulsante, ecc...) è selezionabile/modificabile oppure no;
navigando su un sito Internet, non capire più dove ci si trova e da dove si era partiti;
non capire se l'azione richiesta è stata eseguita dal sistema;
non capire perché l'azione richiesta non è stata eseguita dal sistema;
essere costretti a imparare a memoria certi comandi o informazioni (o ricorrere alla guida operativa o chiedere suggerimenti a colleghi e amici più esperti).

Alcuni suggerimenti⁴⁸ per agevolare il riconoscimento:

fare in modo che le azioni consentite sull'interfaccia siano chiaramente visibili;
rendere evidente lo stato del sistema ad ogni momento e ad ogni azione dell'utente (risultati dell'azione svolta, contesto nel quale ci si trova, ecc.);
utilizzare un linguaggio significativo per l'utente, in modo da non costringerlo a interpretazioni del significato;
dotare tutte le pagine di un titolo significativo che illustri adeguatamente il tipo di informazione visualizzata o le azioni da svolgere sulla finestra o sulla pagina;

47 ROSENFELD, L., MORVILLE, P., *Information Architecture for the World Wide Web*, O'Reilly, Sebastapol, CA, 1998.

48 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

non usare da sole icone e simboli grafici o almeno fornire delle descrizioni delle funzionalità associate all'icona stessa;
abilitare o disabilitare i comandi in base al contesto operativo, in modo da ricordare all'utente l'obbligatorietà di certe azioni o la relazione sequenziale tra certe altre;
rispettare la consistenza nell'organizzazione dei contenuti e degli oggetti all'interno delle finestre e delle pagine, in modo da non costringere l'utente a continue osservazioni dell'intera finestra o pagina per ritrovare determinati oggetti o gruppi di informazioni.

4. Rendere visibile lo stato del sistema attraverso un feedback

Il *feedback* rappresenta l'informazione di ritorno, la reazione del sistema in risposta all'azione che l'utente ha eseguito sulla interfaccia; il *feedback* serve a segnalare all'utente lo stato corrente del sistema e l'esito della propria azione: comprende quindi anche i messaggi di errore.

Il *feedback* non si riferisce solo alle azioni scorrette dell'utente e alla relativa messaggistica di errore, ma comprende tutti i modi per trasmettere informazione all'utente lo stato del sistema⁴⁹:

quale azione ha eseguito o sta eseguendo l'utente;

qual è la reazione del sistema all'azione sull'interfaccia da parte dell'utente;

il nuovo stato del sistema a seguito dell'azione effettuata.

Oltre alla messaggistica, prevalentemente usata per segnalare errori di interazione, le occasioni per fornire *feedback* sullo stato del sistema sono molteplici⁵⁰:

la comparsa della finestra che mostra il passaggio da una cartella a un'altra quando l'utente effettua una azione di *drag & drop*;

il cambio di forma del cursore del *mouse* dopo aver eseguito la selezione di qualche strumento indica il tipo di operazioni possibili;

l'uso di un indicatore di avanzamento informa del tempo necessario ad eseguire una certa azione da parte del sistema;

la visualizzazione della non disponibilità di un oggetto indica che quello strumento non è disponibile o che quell'azione non è consentita;

la visualizzazione su una pagina Web del *path* seguito all'interno del sito;

Il cambio di colore di un *link* rispetto allo stato *hover* che segnala la disponibilità a essere seguito, o il cambio di colore dopo averlo effettivamente seguito.

Ma entro quanto tempo deve essere fornito il *feedback*? Jakob Nielsen⁵¹ riporta le seguenti indicazioni:

49 DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

50 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

un decimo di secondo è approssimativamente il tempo massimo per dare all'utente la sensazione che il sistema ha reagito istantaneamente;

1 secondo è approssimativamente il tempo massimo per mostrare i risultati dell'azione dell'utente senza interrompere il suo flusso di ragionamento, anche se noterà il ritardo nella risposta del sistema;

10 secondi è il tempo massimo per mantenere l'attenzione dell'utente focalizzata sul dialogo. Oltre i 10 secondi, generalmente, l'utente inizia una altra attività mentre il computer sta lavorando: se si vuole mantenere l'attenzione dell'utente sull'applicazione o sul sito, la risposta all'azione dell'utente deve essere mostrata non oltre 10 secondi.

5. Prevenire e limitare gli errori di interazione

Commettere errori nell'interazione con un prodotto è naturale. Ogni azione dell'utente va concepita come un tentativo verso una giusta direzione. Si tratta di una componente naturale del dialogo utente-sistema che va tollerata, garantendo la giusta flessibilità di utilizzo che consente agli utenti di navigare liberamente senza entrare in vicoli ciechi e in situazioni critiche.

Ci sono alcuni tipi di errori che sono difficilmente eliminabili, come le sviste: inconsapevolmente, viene eseguita un'azione diversa rispetto a quella che ci si proponeva nelle intenzioni, a causa di una distrazione o di un'interruzione.

Altri tipi di errore, invece, si possono prevenire con una buona progettazione dell'interfaccia: sono gli errori commessi a seguito di un'applicazione sbagliata di regole di interazione o per la mancanza di sufficienti e adeguate informazioni e conoscenze. Rientrano tra questi tipi di errore quelli dovuti a un modello di dialogo che l'utente non capisce a pieno o che non incontra le sue aspettative: per questo, l'utente può applicare regole di interazione "sbagliate" rispetto a quelle richieste dal prodotto.

Il contributo fondamentale alla prevenzione degli errori d'interazione deriva dal rispetto dei principi sul dialogo, i compiti, il riconoscimento, il *feedback* sulla cui base è consentito agli utenti di individuare, riconoscere e adeguare le proprie azioni alle possibilità offerte attraverso l'interfaccia⁵².

Altri modi di prevenire gli errori prevedono l'utilizzo di funzioni bloccanti, che impediscono la continuazione di azioni sbagliate o che possono portare a risultati distruttivi.

Tuttavia, poiché gli errori sono sempre possibili, è importante che il sistema sia progettato in modo da diagnosticarli quando occorrono e facilitarne la correzione.

51 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

52 NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

I modi più semplici per raggiungere questo obiettivo sono⁵³:
fornire funzionalità di annullamento delle operazioni, come *undo* e *redo* o di ripristino delle condizioni di *default* o di *back*;
fornire una messaggistica efficace;
fornire comandi per interrompere le operazioni lunghe;
rendere sempre disponibili le funzioni per uscire dal programma o per ritornare alla *home page*.

6. Essere coerenti

La coerenza permette all'utente di trasferire agevolmente la conoscenza da un'applicazione all'altra, aumenta la predicibilità delle azioni e dei comportamenti del sistema e ne favorisce l'apprendibilità.

Un problema di coerenza che caratterizza il mondo Web è relativo ai *font* e ai *link*.

Molto spesso, all'interno dello stesso sito, si vedono pagine con *font* diversi per dimensione, stile e colori. Analogamente, i *link* vengono proposti in svariati formati: *standard* (sottolineato), in grassetto (ma senza sottolineatura), colorati senza limiti alla fantasia.

Le pagine Web sono da immaginare come pagine di un libro: non si è mai visto un libro che usa caratteri diversi per ogni paragrafo, titoli di dimensioni e colori differenti posizionati una volta centrali e una volta a sinistra sulla testata della pagina, rimandi ad approfondimenti ogni volta presentati con uno stile diverso senza una logica di fondo. L'uso di font e stili diversi deve avere una sua motivazione

L'incoerenza nell'uso dei *font*, nella struttura della pagina, nella grafica genera una situazione di confusione.

La coerenza in generale deve essere garantita a diversi livelli. Vediamoli in sintesi⁵⁴:

architettura informativa: disposizione e *naming* delle informazioni;

coerenza del linguaggio e della grafica: la stessa parola, la stessa icona, lo stesso colore devono identificare lo stesso tipo di informazione o lo stesso tipo di azione entro tutto il prodotto;

coerenza degli effetti: gli stessi comandi, le stesse azioni, gli stessi oggetti devono avere lo stesso comportamento e produrre gli stessi effetti in situazioni equivalenti; non devono essere associati agli stessi comandi, azioni e oggetti comportamenti diversi;

53 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

54 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000; ROSENFELD, L., MORVILLE, P., *Information Architecture for the World Wide Web*, O'Reilly, Sebastopol CA, 1998; FLEMING, J., *Web Navigation: Designing the User Experience*, Songline Studios, Sebastopol CA, 1998.

coerenza nella presentazione: gli stessi oggetti o lo stesso tipo di informazioni devono essere collocati tendenzialmente nella stessa posizione, avere la stessa forma e lo stesso ordine;

coerenza tra ambienti applicativi: un'applicazione, un sito non sono mondi isolati.

7. Facilitare la flessibilità d'utilizzo e l'efficienza dell'utente

Nella definizione degli strumenti in grado di agevolare la flessibilità e l'efficienza, va considerato che le esigenze degli utenti variano in relazione al loro livello di esperienza rispetto al compito e alle tecnologie informatiche. Ne consegue che, in relazione a questi due aspetti, il livello di supporto richiesto, gli strumenti utilizzati e le strategie di interazione messe in atto dagli utenti possono essere diverse.

Gli utenti non esperti, ad esempio, amano essere guidati passo per passo, mentre gli utenti più esperti preferiscono utilizzare scorciatoie, delle quali anche utenti non esperti, man mano che aumenta il loro livello di esperienza, possono usufruire.

Si può agevolare la flessibilità e l'efficienza d'uso fornendo, ad esempio:

assistenza intelligente (l'anticipazione da parte del sistema nell'inserimento di un termine) e *shortcut* (una combinazione di tasti, come ad esempio, CTRL+C per il comando Copia);

salto nella navigazione che evitano di passare in punti intermedi;

funzioni di personalizzazione dell'interfaccia, ovvero la possibilità di modificare alcuni aspetti del sistema in base alle esigenze del compito, alle caratteristiche dell'utente e sue preferenze personali.

Da notare che, una volta personalizzato, il sistema deve mantenere, alle successive riaperture, le impostazioni date dall'utente. Alcuni esempi di aspetti dell'interfaccia che dovrebbero essere personalizzabili sono:

la lingua,

la dimensione dei caratteri,

le impostazioni di default,

il formato dei dati presentati e livello di dettaglio,

la disposizione di alcuni oggetti grafici.

Un altro aspetto dell'efficienza è anche il tempo di risposta del sistema alle azioni dell'utente, che rappresenta uno dei problemi più critici nelle applicazioni per il Web.

I tempi di risposta a ogni chiamata dell'utente possono dipendere da molte cose (performance dei server e della rete, tipo e velocità di connessione, quantità di informazione che deve essere trasferita, ecc...), ma gli utenti non sono interessati a queste motivazioni: se l'attesa è lunga, pensano solo che non viene offerto loro un buon servizio e il loro livello di fiducia è destinato a diminuire.

8. Fornire help e manuali

L'argomento della documentazione (*help* in linea o manuali utente) è piuttosto controverso per diversi motivi⁵⁵:

un buon prodotto, teoricamente, non dovrebbe richiedere la consultazione della documentazione;

la documentazione viene spesso usata per compensare eventuali problemi di usabilità del prodotto;

nella maggior parte dei casi, gli utenti ignorano questi strumenti di supporto.

Per quanto riguarda l'ultimo punto, infatti, generalmente gli utenti ricorrono all'*help* in linea o alla documentazione solo come ultimo tentativo, cercando (e non trovando quasi mai) la soluzione al proprio caso specifico. Nella lettura delle informazioni riportate, inoltre, tendono a non approfondire gli argomenti, leggendo rapidamente solo poche righe. Infine, come per tutti i testi scritti, la comprensibilità di *help* e manuali, se non accuratamente verificata con gli utenti finali, non è sempre garantita.

Considerati questi aspetti, quando la documentazione può essere necessaria, essa va realizzata con l'obiettivo di garantire:

facilità di consultazione;

comprensibilità e brevità dei testi;

orientamento all'attività dell'utente;

efficacia nella risoluzione del problema.

⁵⁵ Per l'uso degli *help* e manuali vedi <http://www.useit.com/alertbox/>.

2. Storia delle interfacce

Come mostrato nella TAB. 2, lo sviluppo storico delle interfacce è interessante poiché ogni nuova generazione contiene molte caratteristiche della generazione precedente, integrandole con nuove potenzialità favorite dallo sviluppo tecnologico e dalle indagini di *usability*. Dunque un buon progettista di interfacce ha la necessità di conoscere lo sviluppo storico delle interfacce per imparare le tecniche di dialogo dalle generazioni precedenti.

Generazione	Hardware	Modalità operativa	Programmazione	Terminale	Utenti tipo	Immagine promozionale	Interfaccia
-1945 <i>Preistoria</i>	Meccanico, elettromeccanico (Babbage)	Non possono essere "utilizzati", se non per i calcoli.	Azione fisica sull'hardware.	Schede perforate.	Gli inventori stessi	Nessuna: il computer non hanno lasciato i laboratori.	Nessuna, accesso diretto all'hardware.
1945-1955 <i>Pioniera</i>	Enormi apparecchi a valvole.	Time sharing: un utente utilizza la macchina solo per un breve lasso di tempo.	Linguaggio macchina binario.	Macchine da scrivere.	Esperti, pionieri	Computer come calcolatore.	Comandi batch.
1955-1965 <i>Storica</i>	Transistor: i computer iniziano ad essere utilizzati fuori dai laboratori.	Computer centrale non accessibile in modo diretto.	Assembler.	Terminali a riga di comando.	Elite tecnica	Computer che processa informazione.	Linguaggi di comando.
1965-1980 <i>Tradizionale</i>	Circuiti integrati.	Utilizzo in rete.	Linguaggio di alto livello, Fortran, Pascal, C.	Terminali alfanumerici full screen.	Gruppi specializzati	Automazione e produttività professionale.	Menu gerarchici full screen e form da compilare.
1980-2000 <i>Moderna</i>	Circuiti integrati: nascono i personal computer.	Utilizzo singolo e personale.	Linguaggio orientato ai problemi e agli oggetti.	Display grafici, workstation desktop e portatili pesanti.	utenti professionali, appassionati	Produttività personale.	WIMP (Windows, Icons, Menus and Pointing Devices).
2000- <i>Futura</i>	Architettura a Wafer.	Utilizzo di rete singolo e collettivo; reti neurali.	Grafica orientata agli oggetti.	Palmari, portatili, cellulari.	tutti	Computer come protesi.	Interfacce non command.

TABELLA 1 Le diverse generazioni di interfacce. Trad. da NIELSEN, J., *Usability Engineering*. Academic Press, Boston, 1993, p.50.

Di seguito viene presentata una breve storia dell'evoluzione delle interfacce, a partire dalle prime interfacce, per arrivare ai possibili sviluppi futuri⁵⁶.

Batch Systems

Le prime generazioni di interfacce non erano ancora interattive. Il *batch* system prevedeva un'interfaccia zero dimensionale nella quale l'interazione fra l'utente e il sistema si limitava a un singolo momento: l'invio del comando *batch* come unità singola. Tutti i comandi utente dovevano essere specificati prima di conoscere il risultato dello stesso. Questo stile di interazione non si dimostra valido per molti scopi; può avere una sua utilità quando viene utilizzato per inviare comandi ripetitivi senza la necessità di una supervisione dell'utente. Quindi molti sistemi moderni prevedono il supporto di alcune istruzioni *batch*; rimane però la necessità di prevedere un intervento dell'utente in qualsiasi momento per interrompere o modificare le operazioni: sarebbe molto frustrante dover attendere la fine del comando prima di poter intervenire. Esempio attuale di sistemi *batch* è costituito dai messaggi per i server di posta elettronica: ad esempio per sottoscrivere o per escludersi da una mailing list.

Interfaccia riga di comando

I sistemi *time sharing*, furono inventati attorno agli Anni '60 per permettere a più utenti di accedere simultaneamente allo stesso *mainframe*: questo per non sprecare la capacità di calcolo e ammortizzare i costi dei primi sistemi. In questo modo però non rimanevano molte risorse disponibili per l'interfaccia utente che era così *line oriented*. Negli Anni '60, non esistevano monitor o tastiere: gli unici dispositivi di input erano le schede perforate, gli unici dispositivi di output le stampanti. Nel 1960 J.C.R. Licklider, che giocò un ruolo fondamentale nel primo periodo del *time sharing*, scrisse "Mancomputer Symbiosis"⁵⁷, nel quale si spiegava la necessità di avere sistemi che riflettessero le necessità dell'utente e le sue abilità.

Le interfacce *line oriented* sono principalmente interfacce unidimensionali, nella quale l'utente può interagire solo con l'ultima riga che serve come *command line*. Una volta che l'utente preme il tasto di invio comando, l'istruzione non è più modificabile. Ugualmente la risposta del sistema non può essere modificata. Dato che le interfacce

⁵⁶ NIELSEN, J., *Usability Engineering*. Academic Press, Boston, 1993; ANCESCHI, G., (a cura di), *Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali*, Domus Academy, Milano 1992, 1993.

⁵⁷ LICKLIDER, J., *Man Computer Symbiosis*, Electronics HPE, Boston, n.1 Marzo 1960.

line oriented non permettevano all'utente di muoversi sullo schermo, la tecnica di interazione era per lo più limitata a una struttura di domanda e risposta: il dialogo era condotto dal computer.

Queste strutture di domanda e risposta sono tuttora valide per quei sistemi in cui i comandi siano pochi e ben definiti, e dove il ruolo guida del sistema possa aiutare l'utente: sistemi per utilizzo casuale o per utenti nuovi.

Ma due sono i problemi fondamentali con questo tipo di interfaccia:

- gli utenti potrebbero avere voglia di cambiare le loro risposte precedenti;
- gli utenti si trovano nella condizione di dover rispondere senza conoscere le domande successive: per la compilazione di un *form on line*, alla domanda "Digitare l'indirizzo", l'utente potrebbe inserire anche il numero civico o il CAP, senza sapere cosa gli verrà richiesto dopo.

La maggior parte delle interfacce riga di comando sono costruite sulla base di linguaggi di comando vari e la maggior parte degli sviluppi dell'usabilità per questo tipo di interfacce è destinata alla individuazione dei nomi migliori per i comandi (*naming*); se è vero che molti linguaggi hanno potenzialità molto grandi grazie all'utilizzo di ampi set di istruzioni, è altrettanto vero che gli stessi linguaggi perdonano raramente gli errori degli utenti, poiché richiedono una sintassi molto precisa. Per velocizzare l'interazione e ridurre il rischio di errori di scrittura, molti linguaggi prevedono delle abbreviazioni per le loro istruzioni. Molte sono le regole che guidano l'abbreviazione dei nomi: troncature le parole o anche rimuovere le vocali.

La regola fondamentale indica che si deve seguire un criterio unico per l'abbreviazione.

Interfacce full screen

La nascita di interfacce che prevedevano l'utilizzo di tutto lo schermo per l'interazione, determinò la nascita di interfacce bidimensionali⁵⁸. Un utilizzo classico delle interfacce *full screen* è dato dalla compilazione dei *form* che hanno la forma di finestre di dialogo, nelle quali possono anche intervenire dei menu *pop up* a comparsa che assistono l'utente nella compilazione dei *form*. Oltre ai menu, molte interfacce *full screen* utilizzano anche tasti funzione; non solo questi tasti svolgono con un solo comando istruzioni complesse, ma sono così pochi e semplici, che possono essere facilmente memorizzati dall'utente.

⁵⁸ NIELSEN, J., *Usability Engineering*. Academic Press, Boston, 1993; ANCESCHI, G., (a cura di), *Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali*, Domus Academy, Milano 1992, 1993.

Gerarchia dei menu

Le interfacce *full screen* dipendono fortemente dalla gerarchia dei vari menu. Questa caratteristica delle interfacce *full screen* è sicuramente uno degli aspetti più importanti. Il rischio di questo tipo di menu è però che la comparsa di varie finestre possa coprire le varie opzioni e costringere l'utente a passare attraverso finestre di dialogo non richieste. Quindi spesso può essere più utile sovraccaricare un unico menu, piuttosto che creare una serie di sottomenu. Il compromesso tradizionale nei menù gerarchici è tra la profondità e l'ampiezza.

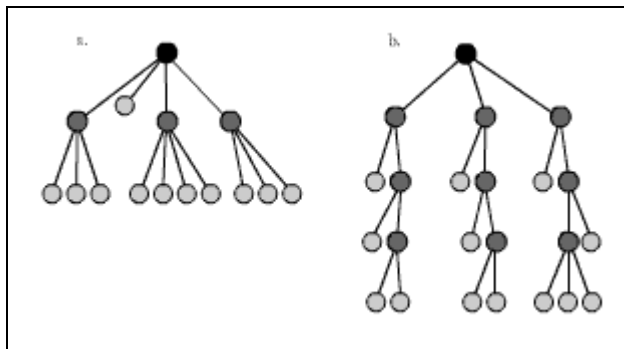


FIGURA 6 Gerarchia dei menù. Tratto da NIELSEN, J., *Usability Engineering*. Academic Press, Boston, 1993, p.56.

Come mostrato dalla FIG. 6 un menù ampio (a.) non obbliga l'utente a passare attraverso una lunga serie di scelte e navigazioni. Nella seconda ipotesi (b.), l'utente è costretto però a scegliere tra una serie di opzioni allo stesso livello, allungando i suoi tempi di risposta.

Ultimamente molte interfacce di sistemi telefonici usati, ad esempio, per ottenere informazioni sul proprio conto bancario, utilizzano quello che appare come un'interfaccia a menu di tipo b.

Interfacce grafiche

Anche se le interfacce grafiche hanno una storia che risale allo Sketchpad System⁵⁹ di Sutherland del 1963 e al *mouse* di Douglas Engelbart del 1963, non sono state utilizzate su scala commerciale prima degli Anni '80.

La maggior parte delle attuali interfacce grafiche possono essere definite WIMP (*Windows, Icons, Menus and Pointing Devices*), secondo quelli che sono i loro componenti principali.

59 SUTHERLAND, I., *Sketchpad: A Man Machine Graphical Communication System*, MIT, Boston, 1963; PEDEMONTE, E., *Personal Media. Storia e futuro di un'utopia*, Bollati Boringhieri, Torino 1998.

Queste interfacce aggiungono una terza dimensione, grazie alla possibilità di sovrapporre più finestre. Ovviamente questa possibilità di sovrapporre le finestre non rende l'interfaccia realmente tridimensionale, poiché non è possibile vedere il contenuto di una finestra oscurata senza muoverla in primo piano. Lo stile di interazione primario utilizzato è quello della manipolazione diretta, basata sulla rappresentazione visuale degli oggetti di dialogo. Il costante aggiornamento degli oggetti manipolati permette all'utente di muoverli sullo schermo con il *mouse*. Ad esempio, il metodo più semplice per specificare la larghezza di una cella in un programma di foglio di calcolo consiste nello spostare il selettore di margine direttamente, fino a impostare la larghezza voluta. Ma così come dimostra lo stesso esempio, questo tipo di interfaccia non è valida per tutti i compiti: per ottenere un'esatta larghezza di margine per esempio è necessario ricorrere a una finestra nella quale inserire il numero esatto di larghezza.

Molte interfacce utente sono dette *object oriented*, piuttosto che *function oriented*, come era invece per i sistemi precedenti.

In un'interfaccia *function oriented*, l'interazione è incentrata attorno a un set di comandi utilizzati dall'utente in vari modi per raggiungere lo scopo desiderato. La questione più importante dell'interfaccia consiste quindi nel rendere facilmente accessibili i comandi, sia attraverso abbreviazioni, sia attraverso menù a pieno schermo.

Nelle interfacce *object oriented* invece, l'informazione è rappresentata come un oggetto visuale, che viene gradualmente modificato dall'utente finché non raggiunge sullo schermo l'aspetto desiderato.

La maggior parte degli specialisti sostiene che le interfacce grafiche hanno migliori caratteristiche di usabilità di quelle testuali, specialmente rispetto al criterio di *learnability* per i nuovi utenti. Gli studi sperimentali svolti⁶⁰ hanno messo a confronto due interfacce, una testuale e una grafica, per lo stesso sistema: hanno dimostrato una maggiore usabilità dell'interfaccia grafica. Tutto ciò che può essere fatto con un'interfaccia testuale può essere replicato con un'interfaccia grafica. Ma non avviene il contrario. Inoltre, l'utilizzo di alcune periferiche che danno la sensazione di un vero e proprio movimento sullo schermo, come il *mouse*, agevolano l'utilizzo di interfacce grafiche.

Proprio questa naturalezza, che a volte può essere eccessiva, porta l'utente a degli errori: l'abitudine di interagire con gli oggetti semplicemente trascinandoli per lo schermo portava alcuni utenti a interagire usando lo stesso meccanismo anche, per esempio, con gli elaboratori testi, cercando di trascinare porzioni di testo: gli sviluppatori hanno quindi introdotto anche sui *word processor* la possibilità del *drag and drop*.

60 ANCESCHI, G., (a cura di), *Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali*, Domus Academy, Milano 1992, 1993.

Un altro esempio è fornito dalle calcolatrici usate nei sistemi operativi *Win* e *Mac*: l'abitudine degli utenti a interagire con la manipolazione diretta, porta gli stessi a utilizzare la calcolatrice attraverso il *mouse*, non utilizzando la più semplice e veloce alternativa del tastierino numerico.

L'interfaccia grafica necessita poi di una costante attenzione da parte dell'utente: dunque non può essere destinata a utenti con problemi o esigenze particolari.

Interfacce della prossima generazione⁶¹

Le interfacce della prossima generazione sono già in sviluppo, e probabilmente estenderanno il numero di dimensioni dalle attuali 2 dimensioni, a 3 o più. Le vie più comuni per aggiungere dimensioni alle interfacce sono l'aggiunta del tempo (in forma di animazioni) dei suoni o della voce, così come ovviamente la creazione di un vero e proprio spazio fisico tridimensionale con l'utilizzo di sistemi per la realtà virtuale. L'obiettivo principale è però quello di rendere le interfacce portatili e semplici da utilizzare come i libri.

Le due caratteristiche su cui più si punta sono la multimedialità, includendo informazioni di varia natura, e la connettività, unendo fra loro i vari sistemi. Un altro obiettivo è quello dell'unione tra applicazioni: permettere all'utente di lavorare allo stesso compito con strumenti diversi. Due metodi semplici ma efficaci, ormai attivi da anni, sono la possibilità del copia incolla fra applicazioni diverse e la possibilità di modificare un oggetto utilizzato in un'applicazione diversa da quella che l'ha creato: ad esempio, modificare direttamente un'immagine allegata a un documento di testo. L'ultimo sviluppo di questa tendenza prevede che l'utente non ragioni più in termini di applicazione aperta, poiché il sistema automaticamente considera un documento complesso come formato da vari sotto-oggetti, ognuno creato da un'applicazione diversa e quindi considerabile separatamente.

Un altro possibile sviluppo potrebbe consistere nella creazione di interfacce che non necessitano di un dialogo esplicito con l'utente, ma che agiscono in funzione di messaggi non espliciti dell'utente, come movimento degli occhi, riconoscimento dei gesti e della postura, analisi semi intelligente delle azioni. Ad esempio, il sistema *Portholes*⁶² per la connettività remota dei gruppi di lavoro visualizza delle miniature dei vari utenti, a seconda della loro presenza/assenza e a seconda delle attività che stanno svolgendo. Senza violare la privacy, come succederebbe ad esempio con un sistema video, il sistema autonomamente informa i colleghi, senza che l'utente sia impegnato a trasmettere il proprio status (l'esperienza con altri sistemi simili ha

61 NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993.

62 LEE, A., *Nynex Portholes: Initial User Reaction*, ACM SIGGROUP, New York, 1997.

dimostrato come gli utenti non spendano volentieri le proprie energie per tenere informati gli altri)⁶³.

Un'altra caratteristica importante potrebbe essere quella di avere un'interfaccia che si adegua a seconda della posizione dell'utente: ad esempio potrebbe ingrandire le dimensioni del testo quando percepisce l'utente lontano dal monitor; potrebbe avvertire l'utente con messaggi di allarme sonori nel caso in cui sia dall'altro lato della stanza; infine, nel caso in cui l'utente riceva un messaggio di posta e non sia al computer, il sistema potrebbe decidere autonomamente di inviare messaggi al suo cellulare, mandare un fax all'hotel dove si trova o inviare il messaggio al computer dell'ufficio.

Una nuova caratteristica dell'interfaccia che sta iniziando a essere implementata è quella che tiene conto delle ultime operazioni dell'utente. Ad esempio, nelle versioni recenti di *word processor*, i file usati di recente vengono visualizzati nella finestra file poco sotto il comando apri, poiché è ragionevolmente probabile che l'utente continui a lavorare sugli ultimi documenti aperti.

Trend di lungo termine⁶⁴

Al contrario di altri settori tecnologici, l'usabilità non è stata monitorata nel corso degli anni e quindi risulta difficile fare previsioni sui suoi futuri sviluppi.

Non è possibile paragonare, dato l'enorme sviluppo di mercato degli utenti dei sistemi informatici dai primi *mainframes* agli ultimi *personal computer*, le interfacce righe di comando del 1950 e le interfacce grafiche del 1990. E' però possibile, stabilendo degli obiettivi da raggiungere, concentrarsi su alcune caratteristiche: pensando per esempio che la *learnability* sia uno degli aspetti più importanti e ponendosi l'obiettivo di diminuire i tempi di apprendimento di un 10% in 2 anni, potremmo concentrarci su determinati aspetti⁶⁵. Sicuramente gli utenti sono in grado di svolgere un numero di compiti decisamente superiore, ma non sempre riescono a svolgere gli stessi compiti in maniera più veloce o più semplice. In altre parole, sono cresciute le potenzialità degli utenti ma non la produttività. Questo generale aumento di usabilità dei programmi ha fatto sì che le aspettative degli utenti siano aumentate e sia diminuita la loro pazienza nei confronti dei sistemi non usabili. Uno dei punti di forza delle future interfacce sarà perciò la possibilità di avere uno stile di interazione multilivello con la possibilità di unire alla manipolazione diretta degli oggetti sullo schermo anche tutti gli altri stili precedenti (riga di comando, *shortcut*, interfaccia testuale...) già appresi dagli

63 Ibidem.

64 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

65 Ibidem.

utenti. Ovviamente ogni stile di interazione cambierà in funzione dell'ambiente in cui sarà implementato.

Stile d'interazione	Interfaccia grafica	Caratteristiche
Batch	Operazioni batch, server di email	Non richiede l'intervento dell'utente.
Question Answer	Riga di comando	Il computer controlla l'utente, indicato per un uso casuale.
<i>Command Language</i>	Riga di comando	Un linguaggio potente può supportare operazioni complesse.
Tasti funzione	Full screen, WIMP	Utilizzo di pochi comandi veloci con limitata flessibilità.
Compilazione form	Full screen, WIMP	Possono essere visualizzati vari campi contemporaneamente.
Menu	Full screen, WIMP, telefonia e domotica	Libera l'utente dal ricordo delle opzioni al prezzo di un sistema potenzialmente lento e con una gerarchia confusa.
Manipolazione diretta	WIMP, Realtà Virtuale	L'utente controlla il sistema attraverso l'uso di metafore del mondo reale.
Non command	Realtà Virtuale, sistemi futuri	L'utente è libero di concentrarsi sul dominio e non sull'utilizzo del computer; i sistemi sono in grado di comprendere le azioni e i comportamenti degli utenti.
Linguaggio naturale	Sistemi futuri	Idealmente permette un input senza limitazioni.

TABELLA 2 Le diverse generazioni di interfaccia. *Tratto da Anceschi, G., (a cura di), Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali, Domus Academy, Milano, 1992-1993, Cap 1.*

La TAB. 3 presenta una sintesi degli stili di interazione presentati. Si noti come i vari stili di interazione non siano stati abbandonati nel corso degli anni, ma integrati, pesando ogni volta la loro importanza e presenza nell'interfaccia.

Sono state volutamente ignorate le interfacce vocali, a causa dei loro problemi di usabilità: nonostante la loro importanza in termini di universalità⁶⁶, un eccessivo uso di istruzioni renderebbe difficile la corretta identificazione del comando stesso; al contrario, pochi comandi ne limiterebbero la potenzialità.

⁶⁶ NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

3. Come: le tecniche dell'usabilità

Il processo user centered

Alla base dell'usabilità c'è il processo *user centered*⁶⁷. L'essenza di questo processo, che prevede il coinvolgimento dell'utente finale del prodotto in tutto il ciclo di ideazione, progettazione e sviluppo, può essere definita come “la pratica di disegnare i prodotti in modo da permettere all'utente di assolvere i propri compiti con il minimo stress e la massima efficienza”⁶⁸.

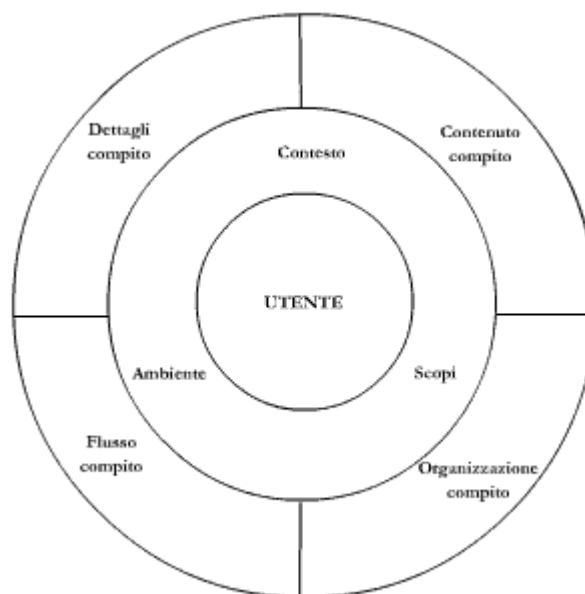


FIGURA 2 I componenti del Processo User centered. *Tratto da RUBIN, J., Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.11.

I quattro principi del UCD⁶⁹, che ne riassumono le caratteristiche, sono:

- una reale focalizzazione sugli utenti e i loro compiti: è necessario un approccio sistematico e strutturato agli utenti, che permetta di registrare tutte le informazioni relative ai loro compiti e che li coinvolga in tutte le fasi del ciclo di vita del prodotto;

⁶⁷ RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994, p.10-24.

⁶⁸ Ibidem.

⁶⁹ Tratto da RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

- misure quantitative e qualitative circa le caratteristiche d'utilizzo del prodotto: queste misure saranno il risultato di metodologie di analisi, ricerca e sperimentazione illustrate in seguito;
- design iterativo, basato sulla struttura "primo design > test > secondo design", da applicarsi fin dalle prime fasi di ideazione in maniera ciclica lungo tutto l'arco di sviluppo del prodotto,
- un approccio multidisciplinare del *team di usability*, che sia in grado di avere una conoscenza trasversale di campi anche molto diversi come marketing, formazione, fattori umani, multimedia.

Questo approccio produttivo comporta⁷⁰:

Coinvolgimento diretto dell'utente finale in tutti le fasi produttive, dall'analisi delle caratteristiche e degli obiettivi di prodotto alla valutazione finale ex-post: l'utente finale diventa il centro di tutti i processi produttivi.

Processo iterativo: una volta completato un prototipo, si attua un processo di valutazione-modifica-nuovo prototipo del prototipo; questo percorso permette di raggiungere elevati standard di usabilità già nelle prime fasi di produzione e permette interventi più frequenti e quindi meno costosi. Inoltre, il percorso attraverso piccoli aggiustamenti progressivi, permette la scoperta di un più alto numero di incongruenze. Alla base del ciclo di produzione di un sistema interattivo entrano in gioco tre elementi: il modello del progettista, l'immagine del prodotto e il modello dell'utente⁷¹.

Il modello progettuale è il modello concettuale del progettista. Il modello dell'utente è il modello mentale sviluppato attraverso l'interazione con il sistema. L'immagine del sistema risulta dalla struttura fisica che è stata costruita (comprese documentazioni, istruzioni, etichette). Il progettista si aspetta che il modello dell'utente sia identico al modello progettuale. Ma il progettista non parla direttamente con l'utente; tutta la comunicazione avviene attraverso l'immagine del sistema. Se l'immagine del sistema non rende chiaro e coerente il modello progettuale, l'utente finirà per formarsi un modello mentale sbagliato.

Per una buona progettazione sono quindi necessarie due condizioni:

- fornire un buon modello concettuale che permetta facilmente all'utente di prevedere i risultati delle proprie azioni;
- rendere visibili le cose e creare delle corrispondenze tra strumenti, azioni e reazioni.

Fondamentale diventa quindi il principio del *mapping*: la metafora che indica la relazione fra due cose: i comandi e il loro azionamento e i risultati che ne conseguono.

A facilitare questo processo interviene il *feedback*: l'informazione di ritorno che segnala all'utente l'esito della propria azione e il conseguente stato del sistema.

⁷⁰ NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993.

⁷¹ NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997, p.25.

I problemi di usabilità e le difficoltà di interazione esistono quando non vengono rispettati questi principi: i comandi diventano arbitrari e non giustificati, costringendo l'utente a un inutile sforzo di memoria.

Le attività del processo user centered⁷²

Ecco l'insieme di attività che dovranno entrare a far parte del processo produttivo per garantire una buona usabilità del prodotto e una facile fruibilità.

Conoscenza degli utenti

Se l'utente deve essere al centro del processo, si deve essere in grado di conoscere le sue caratteristiche.

Spesso l'utente *target* è identificato dagli obiettivi di marketing, che definiscono a chi è indirizzato il prodotto, costituendo il profilo del target principale e del target secondario: età, sesso, livello di esperienza rispetto al dominio di conoscenza, livello di esperienza rispetto alle tecnologie informatiche...

Non solo deve essere considerato il profilo dell'utente, ma soprattutto deve essere definito l'insieme di compiti e attività che l'utente svolgerà o desidera svolgere. Questi compiti dovranno poi essere ordinati per importanza, frequenza, criticità (induzione all'errore e difficoltà).

Oltre a questi due aspetti si dovranno valutare il contesto d'utilizzo del prodotto nei suoi vari aspetti sociali e tecnologici e gli sviluppi futuri d'utilizzo ipotizzati.

Analisi comparativa

Se si è nelle condizioni di sviluppare un prodotto per il quale esistono già *competitor*, diventa fondamentale definire dei *benchmark* che potranno essere utilizzati come indici di riferimento. Ovviamente con il nuovo prodotto, si dovrà fornire agli utenti un livello di prestazioni superiore a quello del prodotto *competitor* che compensi lo sforzo per il passaggio dall'uno all'altro.

Definizione dei requisiti di usabilità

Si devono definire quali sono le priorità dei vari aspetti dell'usabilità del prodotto. Questa gerarchia dovrà guidare nella scelta fra soluzioni diverse: ad esempio, se il prodotto prevede un utilizzo saltuario, l'accento dovrà essere posto più sulla facilità di apprendimento e utilizzo, piuttosto che sulla personalizzazione.

⁷² RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994; DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993.

Per ciascuna caratteristica vanno individuate delle misure di riferimento (se possibile sfruttando l'analisi comparativa) e i limiti di accettabilità.

Progettazione parallela

L'utilizzo di diversi progettisti che lavorano in maniera autonoma e indipendente a diverse soluzioni, permette all'inizio di esplorare vie diverse senza allungare i tempi di produzione. Una volta completate le diverse prototipazioni, che hanno un costo molto basso poiché sono realizzate in maniera schematica, spesso semplicemente su carta, il passo successivo consiste nell'unione in un nuovo prototipo unificato delle caratteristiche migliori di ogni soluzione. Ovviamente si dovrà fare attenzione a mantenere la coerenza del nuovo prototipo.

Una possibilità ulteriore è offerta dallo sviluppo di diversi prototipi specializzati: ogni soluzione progettata pone l'accento su una caratteristica; ad esempio, per lo stesso prodotto potremmo avere 3 interfacce rispettivamente destinate a nuovi utenti, a utenti esperti o a un'interazione non verbale. La specializzazione e diversificazione di ogni prototipo permette di approfondire quell'aspetto specifico e riunire le caratteristiche migliori nel nuovo prototipo unificato.

Prototipazione

La prototipazione è quindi fondamentale in due fasi:

nella progettazione parallela permette di esplorare varie strade a costi limitati;

nelle prime fasi di produzione permette di correggere già le prime incongruenze avendo la possibilità di arrivare in fase di sviluppo con delle indicazioni precise.

Le valutazioni dei prototipi possono essere condotte attraverso vari metodi (*heuristic evaluation, cognitive walkthrough*, valutazione cooperativa, ecc...) in funzione delle risorse e dei tempi a disposizione e al tipo di risultati che si vogliono ottenere.

Una possibilità molto efficace è quella di modificare il prototipo *on the fly*, in sede di valutazione, in modo da applicare immediatamente le modifiche e valutarle nuovamente: c'è la possibilità che le nuove modifiche introducano ulteriori problemi.

Validazione di usabilità attraverso test

L'insieme di attività svolte permettono di arrivare al test finale, precedente il rilascio del prodotto, già con un buon livello di usabilità validato durante il ciclo produttivo.

Indagine follow-up

Una volta avviata la commercializzazione del prodotto, si dovranno condurre delle indagini che permettano di monitorare le reazioni dei clienti, il loro grado di

soddisfazione ed eventuali nuovi errori di usabilità. Le nuove osservazioni verranno utilizzate per le future versioni del prodotto.

Valutazione di usabilità: metodi e metriche⁷³

La valutazione di usabilità è una delle attività che contraddistingue il processo *user centered*.

I metodi di indagine di usabilità di un prodotto sono numerosi: ognuno è caratterizzato da obiettivi diversi e quindi viene utilizzato nelle varie fasi del ciclo di vita del prodotto.

Con le attività di prototipazione e progettazione parallela è possibile effettuare i primi controlli di usabilità non appena sono disponibili le prime bozze di interfaccia.

Non è necessario avere un prodotto funzionante per applicare test di usabilità, bastano anche i prototipi cartacei.

L'altra attività del processo *user centered* è il design iterativo: questo ci porta ad applicare varie volte i test di usabilità all'interno del ciclo di vita.

E' da notare, infatti, che le modifiche effettuate sull'interfaccia dopo la prima verifica di usabilità possono non essere efficaci ai fini della risoluzione dei problemi riscontrati. E c'è sempre il rischio che i cambiamenti introducano altri problemi di usabilità.

I principali metodi di valutazione dell'usabilità sono⁷⁴:

Heuristic evaluation

Cognitive walkthrough

Thinking aloud

Task analysis

Questionario

Heuristic evaluation

Con la *heuristic evaluation* si rileva la fedeltà e l'aderenza del prodotto ai principi di usabilità.

Questo metodo prevede il coinvolgimento solamente degli esperti di usabilità e non chiama in causa gli utenti finali: per questo motivo è facilmente eseguibile, economico e rapido.

⁷³ Bailey, R., *Usability Testing versus Heuristic Evaluation: A Head-to-Head Comparison*. Proceedings of the Human Factors Society 36th Annual Meeting, 409-413, New York, 1992; DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR, 1993; Lindgaard, G., *Usability Testing and System Evaluation: A Guide for Designing Useful Computer Systems*, Chapman and Hall, London, U.K. 1994; Nielsen, J., *Usability Engineering*, Academic Press, Boston, 1993; NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

⁷⁴ Ibidem.

Dal punto di vista pratico, la *heuristic evaluation* consiste in una serie di navigazioni del prodotto effettuate separatamente da ciascun esperto. Durante l'utilizzo, il prodotto viene valutato sia per gli aspetti statici dell'interfaccia (*layout* delle finestre, etichette, pulsanti, ecc.) sia per gli aspetti di interazione (logica e processi, flussi) rispetto alle linee guida di riferimento. Una volta terminate le indagini da parte di ogni esperto, si riuniscono i risultati e si arriva a delle conclusioni comuni.

Gli studi statistici effettuati da Nielsen⁷⁵ indicano che un solo esperto di usabilità riesce a individuare circa il 35% dei problemi di usabilità, mentre cinque esperti diagnosticano circa il 75% dei problemi: dunque è necessario effettuare una *heuristic evaluation* con almeno tre esperti di usabilità, se le risorse lo permettono con cinque esperti.

Cognitive walkthrough

Esistono diverse varianti di *usability walkthrough*, un metodo di valutazione dell'interfaccia basato sul giudizio di esperti di usabilità, progettisti, utenti e altre figure professionali.

Anche nel caso del *cognitive walkthrough* è possibile iniziare la valutazione quando siano disponibili i prototipi.

Questo metodo si basa sulla teoria dell'apprendimento esplorativo di Polson e Lewis⁷⁶, secondo la quale il processo di *problem solving* di un utente si basa su un'attività in quattro fasi:

Partire con una descrizione grossolana del compito che si vuole compiere;

- Esplorare l'interfaccia e selezionare le azioni che si eseguiranno (o parte di essa).
- Osservare le reazioni dell'interfaccia per vedere se le azioni hanno avuto l'effetto desiderato.
- Determinare quale azione compiere successivamente.

Applicando questa teoria, le varie caratteristiche di un prodotto sono correlate con la facilità di apprendimento: se il modello del prodotto è difficile da comprendere e apprendere, allora sarà anche difficile da imparare e ricordare.

Il *cognitive walkthrough* si concentra quindi sugli errori di progettazione dell'interfaccia che potrebbero rendere difficile o impossibile l'apprendimento delle modalità di utilizzo da parte dell'utente finale.

⁷⁵ NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

⁷⁶ POLSON, P.G., LEWIS, C.H., *Theory-based design for easily learned interfaces*. Human-Computer Interaction, 5, 191-220, 1990.

Una sessione di *cognitive walkthrough* coinvolge il progettista, chiamato a illustrare il prodotto a un gruppo di valutazione composto da altri progettisti, esperti di usabilità, utenti esperti. In particolare sarà indicata la:

descrizione degli utenti di prodotto;

descrizione delle attività e dei compiti;

descrizione della corretta sequenza di azioni da compiere per assolvere il compito.

Si eseguono i vari compiti e il gruppo di valutatori giudica le caratteristiche delle schermate ponendosi queste domande:

- gli utenti cercheranno di ottenere il giusto effetto? Il loro compito è stampare il documento: la prima cosa che devono fare è selezionare una stampante - sapranno che devono fare ciò?
- all'utente è noto che l'azione corretta è disponibile? Se l'azione è selezionare da un menù visibile, non ci sono problemi. Ma se c'è da fare un triplo click sull'icona della stampante l'utente non lo saprà mai;
- l'utente assocerà la corretta azione con l'effetto che vuol raggiungere? Se c'è una voce di menù che dice "seleziona una stampante" le cose andranno lisce. Non andranno bene se il menù dice "Sys P");
- se l'azione corretta è eseguita, l'utente vedrà che è stato fatto un progresso verso la soluzione del compito? Se dopo aver selezionato la stampante salta fuori un box di dialogo, tutto bene; male quando non c'è *feedback*.

I tratti critici di una interfaccia, allora sono quelli che forniscono legami tra la descrizione del compito dell'utente e l'azione corretta, e quelli che forniscono *feedback*, indicando che la prima azione fatta dall'utente è in esecuzione.

Pluralistic walkthrough

Un'altra tipologia di *walkthrough* è il *pluralistic walkthrough*, al quale partecipano principalmente tre tipologie di valutatori: utenti finali, esperti di usabilità, progettisti *software*.

A differenza di altri metodi *walkthrough*, nel *pluralistic walkthrough* tutti i partecipanti assumono il ruolo degli utenti finali e, prima di eseguire le azioni sull'interfaccia, queste vengono scritte su fogli, in modo che la strategia di interazione sull'interfaccia possa essere analizzata nel dettaglio per il miglioramento dell'usabilità.

Thinking aloud

Questo metodo, che mutua i propri strumenti dalla ricerca psicologica, consiste nel far verbalizzare agli utenti quello che pensano durante l'esecuzione di una attività o

compito. Obiettivo del *thinking aloud* è quello di far emergere le logiche di interazione e il modello dell'utente.

Accanto all'utente o comunque in grado di osservarlo, troviamo il ricercatore che può rivestire un ruolo passivo e dunque non intervenire durante la sessione di interazione; in alternativa troviamo la valutazione cooperativa nella quale utente e ricercatore collaborano all'indagine del prodotto; in questo caso il ricercatore è in grado di intervenire durante le fasi di difficoltà dell'utente, per aiutarlo e approfondire quindi l'indagine nelle fasi critiche di errore. Questa variante permette sicuramente di andare a fondo nella logica di interazione, ma il rischio è dato dalla possibilità che il ricercatore influenzi il comportamento dell'utente osservato.

Questo metodo di valutazione ha un lato critico poiché l'utente è chiamato a un duplice sforzo cognitivo: non solo deve utilizzare il prodotto, ma soprattutto deve raccontare la propria interazione. Non tutti gli utenti sono in grado di farlo o di farlo in maniera naturale.

Il numero di utenti utilizzati nel *thinking aloud* è in relazione alle risorse disponibili: in ogni caso almeno 3 campioni per classe di utenti.

Task analysis

Questa metodologia permette più delle altre una rilevazione quantitativa e comparativa delle caratteristiche di usabilità; inoltre è il metodo più indicato per valutare prototipi funzionanti o prodotti finiti prima del loro rilascio sul mercato.

Prevede il coinvolgimento diretto degli utenti finali che sono chiamati a usare il prodotto all'interno dei laboratori di usabilità o nel loro contesto di utilizzo originario. Generalmente i laboratori di usabilità prevedono la postazione utente e il posto del ricercatore (*test monitor*) e degli osservatori, a volte posizionati in una stanza gemella con specchi unidirezionali che permettono l'osservazione dell'utente a lavoro. Il rischio è però quello di creare delle condizioni di utilizzo non fedeli e di caricare l'utente di un'ansia da prestazione che renderebbe il suo comportamento non naturale. Prima dell'inizio dei test vengono definiti gli obiettivi tipici di utilizzo del prodotto e, in funzione di questi, dei *task* tipici da far eseguire nella sperimentazione: viene descritto il loro svolgimento corretto e vengono ordinati per importanza, frequenza e criticità.

Se possibile, vengono fissati degli indici, dei *benchmark* di riferimento e dei limiti di accettabilità.

Infine si selezionano gli utenti partecipanti alle sperimentazioni.

Durante l'effettuazione dei compiti, *test monitor* e osservatori misureranno le prestazioni e rileveranno/analizzeranno gli errori di utilizzo.

In sede di analisi si confronteranno i risultati sia dal punto di vista qualitativo che statistico-quantitativo.

Anche nel caso dell'analisi dei *task*, il numero di utenti utilizzati è in relazione alle risorse disponibili: per ottenere dati qualitativi dai 3 ai 5 utenti per categoria; per ottenere risultati quantitativi per comparazioni e indici, almeno 20 utenti.

Metriche e misure⁷⁷

La misurazione dell'usabilità si basa sull'analisi dei dati relativi all'interazione utente-prodotto. Si presta più facilmente per quelle tecniche di indagine che coinvolgono direttamente l'utente finale, come la *task analysis* o i questionari.

Ovviamente, per ottenere una validità statistica dei dati è necessario definire con attenzione sia gli obiettivi di usabilità sia i compiti da eseguire e le relative tecniche di valutazione delle prestazioni.

Le metriche principali dell'usabilità sono:

Efficacia: la misura in cui un utente è in grado di raggiungere l'obiettivo di un compito in modo corretto e completo;

Efficienza: la quantità di risorse spese in relazione all'efficacia;

Soddisfazione d'uso: la piacevolezza e il gradimento dell'utilizzo del prodotto;

Facilità di apprendimento: in relazione alla curva di apprendimento dell'utente che indica il tempo necessario per eseguire correttamente i compiti;

Facilità di ricordo: la facilità con cui le procedure di interazione del prodotto vengono memorizzate dall'utente.

Per quanto riguarda le misure, quelle tipiche sono le seguenti:

tempo necessario per eseguire i compiti;

compiti eseguiti in modo corretto;

compiti eseguiti completamente entro un tempo prestabilito;

numero di errori commessi nell'esecuzione dei compiti;

gravità e ricorrenza degli errori commessi;

tempo necessario per correggere gli errori commessi;

tempo speso per l'esplorazione del prodotto;

volte di utilizzo dell'*help*;

tipo di comandi usati dall'utente per l'esecuzione del compito;

numero e tipo di comandi ignorati dall'utente;

reazioni dell'utente.

Focus group

⁷⁷ BAILEY, R., *Usability Testing versus Heuristic Evaluation: A Head-to-Head Comparison*. Proceedings of the Human Factors Society 36th Annual Meeting, 409-413, New York, 1992; DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR, 1993; LINDGAARD, G., *Usability Testing and System Evaluation: A Guide for Designing Useful Computer Systems*, Chapman and Hall, London, U.K. 1994; NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993; NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

I focus group sono strumenti di approfondimento mutuati dal marketing. Prevedono il coinvolgimento all'interno di un gruppo di discussione di utenti reali o potenziali del prodotto, progettisti, responsabili di prodotto, esperti di usabilità. E' prevista la presenza di un moderatore in grado sottoporre all'attenzione dei partecipanti una serie di *item*.

Un focus group con una composizione così eterogenea ha lo scopo di apportare elementi di innovazione e problem-solving: ad esempio, definire nuovi servizi o modificare procedure esistenti; il focus group si rivela quindi particolarmente utile in fase di concept o restyling di un prodotto.

Questionario

I questionari sono degli strumenti di valutazione dell'usabilità molto economici in quanto possono essere somministrati a un numero molto ampio di utenti contemporaneamente e senza la necessità di disporre di grandi risorse. Allo stesso tempo, però, sono soggetti a tutte le limitazioni connesse con l'utilizzo dei questionari in generale (soggettività dei dati, difficoltà a controllare le condizioni di somministrazione in remoto, ecc...).

Dal momento che la compilazione di un questionario implica la conoscenza del prodotto sul quale si deve esprimere il giudizio, i questionari sono particolarmente adatti per la valutazione dell'usabilità di prodotti già rilasciati. Possono quindi essere utilizzati per rilevare e monitorare la percezione di usabilità che maturano gli utenti in relazione al tempo di utilizzo e decidere, quindi, i momenti di interventi di manutenzione sul prodotto.

I questionari più noti sono:

SUMI (Software Usability Measurement Inventory)

Sviluppato dallo Human Factors Research Group dell'Università di Cork ⁷⁸ (Irlanda), si compone di 50 domande alle quali l'utente risponde in termini di accordo, indeciso e disaccordo.

Le domande vertono su cinque aspetti dell'usabilità (*efficiency, affect, helpfulness, control, learnability*).

I punteggi forniti da SUMI sono standardizzati (mediana = 50 e deviazione standard ± 10).

WAMMI (*Website Analysis and Measurement Inventory*).

Sviluppato in collaborazione dalla Nomos Management ⁷⁹(Stoccolma) e dallo Human Factors Research Group dell'Università di Cork (Irlanda), è un questionario specifico per la rilevazione dell'usabilità dei siti Web.

⁷⁸ Il sito Web dello Human Factors Research Group dell'Università di Cork è raggiungibile all'indirizzo <http://www.ucc.ie/hfrg>.

⁷⁹ Il sito Web della Nomos Management è raggiungibile all'indirizzo <http://www.nomos.se/>.

Si compila *on line* e consente di rilevare il giudizio degli utenti sui seguenti aspetti di usabilità delle interfacce: *attractiveness, controlability, efficiency, helpfulness, learnability*. Come SUMI, i punteggi ottenuti attraverso WAMMI sono standardizzati.

QUIS (Questionnaire for User Interaction Satisfaction)

Sviluppato dallo Human-Computer Interaction Lab (HCIL) dell'Università del Maryland⁸⁰ a College Park.

Si compone di undici sezioni, customizzabili, ciascuna dedicata alla rilevazione del giudizio degli utenti su specifici aspetti dell'usabilità del prodotto: *screen factors, terminology and system feedback, learning factors, system capabilities, technical manuals, on-line tutorials, multimedia, voice recognition, virtual environments, Internet access, software installation*. Gli utenti rispondono alle domande delle varie sezioni all'interno di una scala da 1 a 9.

PUTQ (Purdue Usability Testing Questionnaire)

Sviluppato da A proposed index of usability: a method for comparing the relative usability of different software systems⁸¹.

Si compone di 100 domande a risposta affermativa/negativa, relative a consistency, compatibility, flexibility, learnability, minimal action, minimal memory load, perceptual limitation, user guidance.

Analisi qualitativa vs misurazione quantitativa⁸²

Compiere un'analisi quantitativa comporta sicuramente dei costi superiori a causa del numero maggiore di utenti coinvolti e della necessità di una fase di analisi più lunga, ma fornisce risultati con rilievo statistico. Un'analisi qualitativa necessita di minori risorse: possono essere sufficienti 5 *usability engineer* o 5 utenti per i test.

Se lo scopo dell'indagine è semplicemente quello di valutare il prodotto per migliorarlo, può bastare un'analisi qualitativa; nel caso in cui volessimo ottenere delle misure di riferimento, effettuare comparazioni tra prodotti diversi o fra versioni diverse del prodotto sarà preferibile un'analisi quantitativa.

⁸⁰ Il sito Web dello Human-Computer Interaction Lab (HCIL) dell'University del Maryland è raggiungibile all'indirizzo <http://www.cs.umd.edu/projects/hcil/>

⁸¹ LIN X., CHOONG, Y., SALVENDY G., *A proposed Index of usability: a method for comparing the relative usability of different software systems*. Behaviour and Information Technology, 1997, vol. 16, n. 4/5, 267-278.

⁸² LINDGAARD, G., *Usability Testing and System Evaluation: A Guide for Designing Useful Computer Systems*, Chapman and Hall, London, U.K. 1994.

Flusso di procedura

Ecco presentata la procedura comprendente i vari step per lo svolgimento di un'indagine di usabilità. In questa procedura è stato ipotizzato un rapporto tra un cliente, un'agenzia creativa che voglia testare il proprio prodotto, e un fornitore, un'agenzia specializzata in usabilità. Questa procedura è stata sviluppata in collaborazione con Yousable, dove viene correntemente applicata.

- **Contatto con il cliente**

Keywords: colloquio, analisi, posizionamento, target, servizi, partner, obiettivi; task principali e secondari; team di progetto e sviluppo; competitor.
Attori: usability engineer, test monitor; management, responsabile marketing.
Deliverable: meeting report, tasks' list, competitor's list, market analysis.

L'incontro serve al coordinatore e ai *test monitor* per conoscere a fondo le esigenze del cliente e soprattutto per capire quali sono i *task* principali e secondari del prodotto che si vuole valutare. Ad esempio, per un sito di prenotazioni di viaggi, i *task* principali potranno essere: ricercare un pacchetto di viaggio per data, per fascia di prezzo o per destinazione; azioni secondarie potranno essere: leggere le schede dei paesi di destinazione, entrare in contatto con partecipanti a viaggi precedenti. Questa conoscenza avverrà attraverso un colloquio con il management del cliente, che illustrerà la storia, le caratteristiche e gli obiettivi del prodotto. Verrà definito inoltre il target di riferimento e i target secondari, di cui verranno tracciati dei profili e, specialmente nel caso di un target di nicchia, verranno anche forniti dei nominativi e dei contatti.

Un altro punto dell'incontro sarà la descrizione del *team* di progetto e del *team* di sviluppo del prodotto, figure con le quali si entrerà in contatto successivamente e che verranno coinvolte nella sperimentazione. Infine verrà illustrata dal responsabile marketing del cliente una breve analisi del mercato e dei *competitor*.

- **Esplorazione preliminare interfaccia**

Keywords: task tipici, familiarizzazione.
Attori: usability engineer, test monitor.
Deliverable: exploration report.

Si diventa utenti del prodotto: si familiarizzerà con gli strumenti di navigazione e con il contenuto, si eseguiranno i *task* principali e secondari.

- **Definizione scenario**

Keywords: task, benchmark, utenti-test, metodologie, coinvolgimento team di progetto.
Attori: usability engineer, test monitor.
Deliverable: test plan.

Si passa a una fase di definizione delle procedure. Prima di tutto si elaborerà la lista dei *task*, categorizzandoli per frequenza, criticità, importanza e infine raggruppandoli in primari e secondari. Se possibile, si produrrà un *benchmark* di questi *task* su prodotti *competitor*. Per quanto riguarda la sperimentazione, si dovrà scegliere: il numero dei *test monitor*, generalmente uno per gruppo di sperimentazione; il numero di utenti-test, tipicamente 2 o 3 batterie da 5 utenti; si dovrà poi elaborare il loro profilo tipo, eventualmente scegliendone più di uno. Si definisce e descrive anche il ruolo del *team* di progetto e sviluppo del cliente e le figure professionali chiamate a partecipare in qualità di osservatori.

Dovrà essere anche elaborata l'analisi qualitativa a cura dell'*usability engineer* e del *test monitor*.

- **Elaborazione offerta**

Keywords: offerta.

Attori: usability engineer, test monitor.

Deliverable: offerta commerciale, GANTT.

All'interno dell'offerta dovranno essere definite le attività da svolgere, le risorse impegnate e i relativi costi e i tempi necessari per svolgere le attività. Nel caso in cui sarà previsto un rimborso per gli utenti test, si dovrà inserire anche questo. Il costo sarà in funzione del numero di utenti e dei gruppi di sperimentazione, della scelta del design iterativo, del numero di analisi qualitative previste e dei tempi di realizzazione. Si definirà il GANTT di progetto.

- **Approvazione del cliente**

Keywords: cliente, risorse.

Attori: usability engineer, test monitor.

Deliverable: lettera d'incarico.

Il cliente sceglierà di accogliere la proposta commerciale ed eventualmente indicherà anche la variante preferita. Indicherà le risorse interne messe a disposizione del progetto e i loro tempi.

- **Utenti-test**

Keywords: nominativi, colloquio, scelta.

Attori: usability engineer, test monitor.

Deliverable: calendario sessioni.

Elenco dei nominativi dei possibili utenti-test; elaborazione scheda profilo da somministrare all'utente e colloquio con utenti-test per confronto con il profilo

definito; calendario delle sessioni di test con gli utenti e conferma della loro convocazione.

- **Preparazione logistica**

Keywords: laboratorio, schede di valutazione, misura, performance, strumenti.

Attori: usability engineer, test monitor.

Deliverable: schede registrazione dati, scheda briefing utenti test, scheda briefing osservatori.

Si dovrà ricreare la situazione di utilizzo del prodotto quanto più simile a quella reale; ad esempio, se un prodotto per il controllo vocale di un computer è concepito per essere utilizzato in un ufficio affollato, la situazione di test dovrà essere la più realistica possibile, eventualmente facendo partecipare gli osservatori. Quindi si dovrà approntare la logistica necessaria: postazione di lavoro, collegamento in rete, strumenti per l'attività. Si prepareranno le schede di valutazione e gli strumenti di misura delle performance degli utenti, come videocamere e cronometri. Si preparerà la scheda dei *task* da somministrare agli utenti test. Si preparerà la scheda per il briefing degli osservatori.

- **Briefing osservatori**

Keywords: osservazione, ruoli.

Attori: test monitor, osservatori, team di progetto e sviluppo.

Si dovrà spiegare il ruolo degli osservatori e le loro competenze.

- **Briefing utenti test**

Keywords: ruolo, naturalezza, niente attesa.

Attori: test monitor, utenti-test.

Si dovrà spiegare il loro ruolo e, cosa fondamentale, si dovrà evitare di far nascere l'ansia da prestazione: dunque si dovranno evitare termini quali misura, prestazione, confronto, analisi, osservazione. E' uno dei punti più delicati, se non il più delicato, perché rischia di falsare le prestazioni dell'utente. Si dovrà evitare di far attendere troppo a lungo, prima e dopo la sessione e in ogni caso si renderà piacevole questa attesa.

- **Esecuzione test**

Keywords: Thinking aloud, quotes, misura task, osservazione.

Attori: test monitor, utenti-test, osservatori.

Deliverable: log test, finding's report.

Nella conduzione del test, fondamentale è il ruolo del *test monitor* che dovrà analizzare il comportamento dell'utente appuntando le osservazioni; uno dei *test monitor* non

impegnato si occuperà della misura delle performance e della registrazione delle stesse. Gli osservatori andranno a integrare gli spunti raccolti dal *test monitor*.

- **Analisi esperta**

Keywords: expertise, check list, task.

Attori: usability engineer, test monitor.

Deliverable: expertise's report.

A cura dello *usability engineer* o di uno dei *test monitor* non attivo nella conduzione della sperimentazione, verranno condotte delle analisi esperte sulla base di una check list di *task*; le osservazioni raccolte dovranno essere più sistematiche e precise possibili.

- **Debriefing utenti-test**

Keywords: naturalezza.

Attori: test monitor, utenti test.

Deliverable: osservazioni.

Si congederà l'utente test; anche in questo caso si dovrà cercare di rendere il comportamento più naturale possibile.

- **Debriefing osservatori**

Keywords: condivisione, concertazione, annotazioni.

Attori: test monitor, usability engineer, osservatori, team di progetto e sviluppo.

Deliverables: observers' finding.

Gli osservatori del test sono invitati a condividere le loro annotazioni; verrà elaborata una lista di osservazioni categorizzate per frequenza, criticità, importanza e infine, facendo riferimento ai *task*, si raggrupperanno in primarie e secondarie. E' importante far uscire tutte le osservazioni, anche quelle che potrebbero sembrare fuori luogo.

- **Registrazione e analisi dati statistici**

Keywords: analisi qualitativa, analisi quantitativa, problemi, aree di forza.

Attori: test monitor, usability engineer.

Deliverable: test analysis, chart and matrix, test report (quantitative analysis + qualitative analysis).

Si passa all'analisi dei dati statistici che saranno stati già ordinati; si dovrà notare la loro frequenza, criticità, importanza. Verranno elaborati matrici e grafici per un'immediata evidenza dei dati.

In un secondo momento vengono comparati con i risultati qualitativi e viene prodotto un documento da presentare al *team* di progetto e sviluppo. E' importante evidenziare i punti di contatto tra analisi quantitativa e qualitativa. Per le osservazioni a bassa criticità vengono formulate le prime raccomandazioni.

- **Debriefing con *team* di progetto e sviluppo**

Keywords: Concertazione, soluzioni, raccomandazioni, aree di forza.

Attori: Usability engineer, test monitor, team di progetto e sviluppo.

Deliverable: meeting report.

Vengono presentati i risultati delle analisi e si stimola la discussione su quelle che possono essere le soluzioni e le raccomandazioni; si analizza anche la fattibilità degli interventi richiesti. Sono evidenziati anche i lati positivi e le aree di forza del prodotto.

- **Primo Report**

Keywords: sintesi, layout, appendici.

Attori: test monitor, usability engineer.

Deliverable: first report.

Viene elaborato un primo report sulla base delle analisi dei dati, dei colloqui e delle analisi qualitative.

Il report dovrà essere strutturato in questo modo: una prima sezione, sintesi delle aree di forza e di debolezza riscontrate. Poi una parte che indichi i *benchmark* relativi ai *task* primari e secondari comparati con quelli dei *competitor*; la descrizione per esteso di tutte le aree di debolezza con le relative soluzioni o raccomandazioni. La sezione seguente presenterà gli strumenti statistici e i dati relativi agli utenti-test. Fondamentale la cura della struttura del report: la *usability* di un *usability* report è importante. Dovrà quindi avere un aspetto professionale, attraente, esauriente, facile da comprendere e dovrà integrare anche degli screenshot d'esempio. Il report viene consegnato al *team* di progettazione e sviluppo che invia le proprie osservazioni.

- Eventuale **Seconda release del prodotto** (design iterativo)

Keywords: ridefinizione prodotto.

Attori: team di progetto e sviluppo.

Deliverable: new release.

Il *team* di progetto e sviluppo ridefiniscono il prodotto sulla base delle indicazioni emerse in questa prima parte; successivamente si ripetono i punti 6-16.

- **Secondo report**

Keywords: integrazione.

Attori: test monitor, usability engineer.

Deliverable: second report.

Al report originario si integrano le osservazioni del *team* di progetto e sviluppo al primo report.

- **Meeting con management**

Keywords: ascolto, fattibilità, soluzioni.

Attori: usability engineer, test monitor, management cliente.

Deliverable: meeting report.

Si presenta il secondo report; si ascoltano le osservazioni e le indicazioni del management, soprattutto in merito alla fattibilità delle soluzioni proposte.

- **Report definitivo**

Keywords: integrazione, sintesi, raccomandazioni, arre di forza, fattibilità.

Attori: usability engineer, test monitor, management, team di sviluppo e progetto.

Deliverable: final report.

Integrazione secondo report con le informazioni emerse dal colloquio con il management. Formulazione delle raccomandazioni e soluzioni effettivamente implementabili e delle osservazioni di carattere generale. Raccolta delle raccomandazioni e soluzioni proposte non implementate.

4. Dove: le applicazioni dell'usabilità

La “cultura dell'usabilità”, intesa come sensibilità alla funzionalità e alla piacevolezza dell'interazione frutto di un'architettura di progetto con l'utente al centro, dovrebbe essere alla base di ogni progetto di prodotto multimediale⁸³.

La cultura dell'usabilità non dovrebbe avere nemmeno un approccio integralista: non per tutti i prodotti ipermediali deve essere garantito necessariamente un accesso e un utilizzo immediato e semplice. La dimensione esplorativa dell'interfaccia, nella sua dimensione ludica, formativa e di intrattenimento, potrebbe essere la caratteristica determinante di alcune categorie di prodotti.⁸⁴

Per facilitare la comprensione di questa apparente contraddizione, può essere utile un paragone con i linguaggi televisivi: lo stile di comunicazione e comprensione, “l'usabilità informativa”, di un tg è radicalmente diversa dallo stile di un *videoclip*; non per questo il *videoclip* è non fruibile: è prevista un'interazione diversa, dove la frammentarietà e la velocità di comunicazione sono parte integrante del linguaggio utilizzato e del contenuto emozionale veicolato⁸⁵.

Anche per i prodotti ipermediali è possibile distinguere linguaggi e fruizioni diverse.

Prodotti informativi

Per alcuni tipi di prodotto, la priorità è lasciata alle informazioni: siti di *e-commerce*, *di news*, *e-banking*, *e-trading*, Cd-rom di consultazione e autoformazione devono garantire un accesso semplice, veloce e potente ai propri contenuti.

Prodotti emozionali

In altri casi la priorità è lasciata all'esperienza vissuta dall'utente, al rapporto empatico prodotto-utente e diventa quindi fondamentale stupire e coinvolgere il fruitore poiché non c'è necessità di un grande passaggio di informazioni, quanto più di emozioni.

Pensiamo quindi al mondo dei videogiochi.

Da un lato troviamo i giochi di strategia con le loro interfacce potenti e che necessitano di essere evidenti: l'utente non deve pensare a come usare l'interfaccia ma al compito da assolvere; ecco perché le interfacce pur essendo ben evidenti in superficie, sono “invisibili”, spariscono nell'utilizzo.

83 NORMAN, D.A., *The invisible computer*, MIT Press Cambridge, MA 1998, tr. it. PARRELLA B., *Il computer invisibile*, Apogeo, Milano 2001.

84 NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997; NORMAN, D.A., *Things That Make Us Smart: Defending Human Attributes in the Age of the Machine*, Addison-Wesley Publishing, MA, 1993.

85 CASTELLANI, L., *La Tv dell'anno zero. Linguaggi e generi televisivi*, Edizioni Studium, Roma, 1995.

Dall'altro lato troviamo le avventure grafiche; l'interfaccia è essa stessa il gioco e la sua esplorazione è la parte fondamentale della dimensione ludica: l'utente non sa dove cliccare per rendere attivo un oggetto, ma deve esplorare la schermata con il cursore che cambierà forma al passaggio sopra la zona sensibile.

E' importante anche sul Web comprendere fin dall'inizio l'obiettivo del prodotto e scegliere l'interfaccia migliore: un errore di usabilità classico sul Web è l'utilizzo di bottoni e *link* non chiaramente segnalati e privi dello stato *hover*.

Dunque: usabilità sì, ma non con un approccio integralista.

Manifesto della usability

Per un'usabilità dall'approccio pragmatico...

no regole ma norme

La usability non detta leggi da seguire ma indica norme a cui ispirarsi.

usability=creatività

La usability non limita la creatività, così come la teoria della percezione non limita un pittore.

no sempre e comunque usability

Per i prodotti emozionali la scarsa o inesistente usability potrebbe costituire l'essenza, ludica o esplorativa, del prodotto stesso.

+ test con utenti - valutazione euristica

Un esperto sarà in grado di *pensare come* un utente, ma *non è* un utente.

analisi del target e selezione utenti

Profilare il target di un prodotto e selezionare il campione rappresentativo costituisce l'80% della qualità di un test di usabilità.

5. Perché applicare i processi di usabilità

I 10 slogan dell'usabilità⁸⁶

La maggior parte delle motivazioni all'usabilità possono essere riassunte in 10 slogan. Ci sono delle apparenti contraddizioni: questo perché l'usabilità è un campo minato che ha bisogno di un'analisi approfondita. Alcune contraddizioni possono però essere risolte solo dalla figura professionale dello *usability engineer*, in grado di scegliere la soluzione migliore per le caratteristiche e le necessità specifiche del progetto.

La tua migliore ipotesi non è la migliore

Una ragione fondamentale dell'esistenza della *usability engineering* è data dal fatto che non è possibile disegnare un'interfaccia ottimale con sicurezza. Gli utenti hanno infinite potenzialità di svolgere compiti nelle maniere più inaspettate e condurre il proprio lavoro nei modi più complessi da immaginare. Il design dell'interfaccia però può essere decisamente migliorato se si pone alla base la comprensione degli utenti e dei loro compiti. Quindi si deve cercare di disegnare l'interfaccia nel miglior modo possibile e poi verificarla con una serie di test. Non è male che un'interfaccia venga quasi completamente ridisegnata dopo uno *user testing*.

Gli utenti hanno sempre ragione

Come abbiamo visto, è possibile che un tentativo iniziale di disegnare un'interfaccia sfoci in un insuccesso. Questo non perché gli utenti sono stupidi. Chi prepara la documentazione di *software* e si accorge che gli utenti hanno difficoltà su un certo punto, si limita a scrivere una casella di testo con "Leggere attentamente le istruzioni", invece di riscrivere la documentazione stessa.

Gli utenti non hanno sempre ragione

Sfortunatamente disegnare un'interfaccia non è solo dare agli utenti ciò che vogliono. Spesso gli utenti non sanno cosa sia meglio per loro; o semplicemente possono avere idee anche molto diverse.

⁸⁶ NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

Gli utenti non sono designer

La soluzione ideale al problema dell'usabilità, potrebbe essere la possibilità di lasciare agli stessi utenti il design dell'interfaccia, semplicemente prevedendo un'ampia flessibilità di personalizzazione dell'interfaccia; si può fornire all'utente proprio ciò che vuole.

L'esperienza ha mostrato come gli utenti inesperti accedano ai pannelli di personalizzazione *software* molto di rado. Al contrario gli utenti esperti utilizzano le caratteristiche di personalizzazione dei programmi.

Innanzitutto, la personalizzazione è semplice solo se inserita all'interno di un disegno coerente e solo se le opzioni si inseriscono in maniera coerente.

Inoltre la personalizzazione stessa necessita di una propria interfaccia che non aumenti la complessità del sistema.

Infine, una personalizzazione troppo ampia potrebbe generare interfacce non omogenee e molto diverse da utente a utente, rendendo difficile anche gli interventi di assistenza al cliente. Ovviamente poi, gli utenti potrebbero scegliere opzioni di personalizzazione non esatte. Una ricerca di Grudin e Barnard⁸⁷ dimostra come gli utenti che utilizzano combinazioni personalizzate di tasti tendano a compiere circa il doppio degli errori degli utenti che utilizzano combinazioni standard.

I designer non sono utenti

I designer di sistema sono umani e sicuramente usano il computer: due caratteristiche comuni con gli utenti. Sfortunatamente però i designer sono diversi dagli utenti per molti aspetti, compresi una differente esperienza informatica e un grado diverso di conoscenza dei concetti di base dell'architettura dei sistemi. Ad esempio un messaggio di errore può essere perfettamente esplicativo per un designer di sistema, ma completamente senza senso per un utente medio.

Gli amministratori delegati non sono utenti

Molti *CEO* hanno iniziato a comprendere l'importanza dell'interfaccia come valore aggiunto per i loro prodotti *software*. Dunque molti dirigenti hanno iniziato a interferire nel processo di design delle interfacce.

⁸⁷ GRUDIN, J., BARNARD, P., *When does an abbreviation become a word and related questions*, ACM, New York, 1985.

Amministratori e dirigenti sono molto diversi dalla maggior parte degli utenti e le loro intuizioni di cosa potrebbe essere il design potrebbero essere errate. Ovviamente però ogni proposta che possa servire da ispirazione è ben accetta.

Meno è più

Una soluzione tentatrice nel design dell'interfaccia potrebbe essere quella di inserire ogni caratteristica e opzione possibile. Se c'è ogni cosa, chiunque può essere soddisfatto. Sbagliato. Ogni elemento aggiuntivo nell'interfaccia aumenta il carico dell'utente nella valutazione di come usare quel dato elemento. Avere un numero inferiore di opzioni invece potrebbe far concentrare l'utente su quelle, spingendolo a comprenderle meglio. Alcune riviste di settore hanno coniato il termine *fatware*, per indicare quei programmi che ad ogni release successiva raddoppiano le loro dimensioni in termini di menù, bottoni, opzioni...

Problemi di dettaglio

Sfortunatamente, l'usabilità dipende da una serie di dettagli minori dell'interfaccia ed ecco perché il lavoro di ingegneria di usabilità deve essere così sistematico.

L'aiuto non aiuta

A volte l'*help* in linea e la documentazione non offrono un vero aiuto. Così spesso gli utenti o non trovano l'aiuto di cui necessitano o non lo capiscono. Inoltre proprio gli *help* introducono ulteriore complessità alla stessa interfaccia. In ogni caso, la possibilità di fornire un aiuto non deve essere vista come scusa per implementare un'interfaccia carente. E' sempre meglio che l'utente sia in grado di operare sull'interfaccia senza bisogno di consultare l'*help*. Quindi un *help* in linea brillante, non può rendere *user-friendly* un'interfaccia incomprensibile.

Il progetto dell'usabilità è un processo⁸⁸

Se è vero che esistono idee anche abbastanza diverse riguardanti dettagliate caratteristiche di usabilità, il processo del progetto di usabilità è invece un principio ben stabile e condiviso. Ogni progetto è diverso ed ogni interfaccia finale ha un aspetto diverso, ma le attività per arrivare a un buon risultato sono più o meno le stesse.

Vantaggi e costi⁸⁹

Ancora oggi, in diversi contesti produttivi, si tende a sottovalutare l'usabilità, ritenendo che essa comporti dei costi che non vale la pena affrontare dal momento che non si intravedono vantaggi immediati per l'azienda produttrice.

Spesso si crede anche di poter risolvere i problemi degli utenti nell'utilizzo dei prodotti allestendo e predisponendo moderni e sofisticati centri di assistenza o *call center*.

Quello che va considerato, invece, è che un prodotto inusabile può essere quasi del tutto inutile (vale la regola "se l'utente non la trova, la funzione non esiste") o, peggio, può rappresentare un costo notevole (elezioni presidenziali del 2000, costi dei *call center* o della manualistica⁹⁰).

Una produzione non orientata all'usabilità e, in definitiva, alla qualità del prodotto, non fa altro che spostare in avanti, nel ciclo di vita dei prodotti, i costi di produzione, spesso maggiorandoli e facendoli pesare quasi interamente sull'utente finale.

Quando un cliente di una *media agency* richiede un servizio di ingegneria dell'usabilità sul proprio prodotto, non è raro che scelga di includere questo servizio in funzione dei costi prospettati; spesso questi costi appaiono solamente come un aggravio inutile ai costi di progetto e vengono quindi scartati.

In realtà, la presenza di una voce di costo riferita all'usabilità non solo rappresenta un investimento anche per il futuro, ma è una chiara indicazione di qualità dell'agenzia: spesso si tende a dare per scontato che, nonostante l'assenza di questa voce sull'offerta, sia presente comunque una verifica di usabilità del prodotto: spesso non è così.

88 NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993; DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland, OR, 1993; LINDGAARD, G., *Usability Testing and System Evaluation: A Guide for Designing Useful Computer Systems*, Chapman and Hall, London, U.K., 1994; MAYHEW, D., *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design*, Morgan Kaufmann Publishers, San Francisco, 1999.

89 BIAS, R., and MAYHEW, D. J., *Cost-Justifying Usability*, AP Professional, Cambridge, MA, 1994.

90 <http://www.useit.com/alertbox> Febbraio 2001.

Il paradosso dell'usabilità in azienda⁹¹

Da un po' di tempo, tra gli operatori dell'IT è di moda la parola *usability*. Molti non comprendono come la *usability* possa avere effetto sulla produttività ed efficienza della propria organizzazione. Mentre le aziende investono in usabilità dei prodotti immessi sul mercato, spesso trascurano l'usabilità dei propri sistemi interni.

Pensiamo, ad esempio, a una società di servizi finanziari. Molte volte al giorno gli operatori accedono alla schermata per la gestione dei conti; ipotizziamo che un errore di *usability* nel design della pagina porti via circa 10 secondi ogni volta che si presenta quella schermata. Se questo errore si presenta 20 volte al giorno per 10 operatori con un costo orario di 40 Euro, abbiamo un costo per la società di 6700 Euro alla settimana, cioè di 70000 Euro all'anno con una perdita di 1750 ore di lavoro.

Il budget disposto per l'addestramento degli operatori è parte del "risparmio" di un pessimo design: un'adeguata *usability* del sistema permette di ridurre questi costi del 50% e di aumentare la produttività del 25%; il beneficio ulteriore: i dipendenti potrebbero iniziare ad apprezzare di più il proprio lavoro.

L'usabilità sul Web non è un bene di lusso⁹²

Su Internet esiste uno strano darwinismo digitale: la sopravvivenza del più semplice. Se un cliente non è in grado di trovare un prodotto, non è in grado di acquistarlo. E' più economico aumentare il budget destinato al design e alla *usability* che aumentare quello pubblicitario e un'attenzione all'usabilità può incrementare la percentuale dei visitatori del sito Web in grado di effettuare un acquisto.

La Rete pone in primo piano l'esperienza del sistema da parte dell'utente, e, solo in un secondo momento, l'acquisto e il pagamento. Fornire ai visitatori una buona esperienza del sistema e un buon impatto è il modo più facile per trasformarli in clienti fidelizzati.

La Rete offre però anche un basso costo di scelta: questa è la vera differenza nel comportamento di un cliente tra mondo reale e Web. Nel mondo fisico, il costo per passare da un negozio a un altro è alto: si deve guidare nuovamente, trovare un nuovo parcheggio...

Questo significa che spesso il processo d'acquisto vada a buon fine nonostante l'insoddisfazione del cliente, poiché il costo per il cambiamento è troppo alto; ovviamente questo cliente non sarà fidelizzato e non tornerà nel punto vendita.

⁹¹ Tratto da <http://www.zdnet.com/devhead/> del 14 settembre 2000.

⁹² Tratto da <http://www.informationweek.com/> del 14 gennaio 2000.

In Rete, se il cliente non trova subito ciò che cerca, il concorrente è lontano soltanto pochi clic.

Il successo su Internet dipende dalla capacità di moltiplicare il numero di visitatori del proprio sito per aumentare il numero di quelli che, in proporzione, possono diventare acquirenti. E' molto costoso per un'azienda, in termini di budget pubblicitari, aumentare il numero dei visitatori: ecco perchè gli investimenti pubblicitari sono così alti.

Per raddoppiare il numero dei visitatori, si può sia raddoppiare il numero dei visitatori o il tasso di conversione visitatore-cliente. Raddoppiare il numero dei visitatori significa almeno raddoppiare gli investimenti pubblicitari; raddoppiare il tasso di conversione può significare anche progettare nuovamente il sito Web e il processo di interazione, attorno al human centered design.

Può essere sufficiente avere un buon design e un buon budget pubblicitario? No. Molti studi dimostrano come gli utenti, pur riempiendo il carrello di prodotti, lascino il sito senza completare il processo di acquisto.

Questo perché il processo si dimostra troppo lungo e oneroso: le informazioni sulla merce fornite al momento dell'acquisto sono sufficienti?

L'utente deve passare attraverso la compilazione di un *form* lungo, con informazioni non necessarie e ripetitive?

I clienti sono rassicurati sulla privacy del trattamento dati?

L'usabilità sul Web non è un bene di lusso: è essenziale per la sopravvivenza. E' la chiave per creare un rapporto di fidelizzazione con il cliente. Poiché i costi per il cambiamento da un punto di vendita all'altro sono bassi, l'usabilità diventa il modo più economico per aumentare il numero delle conversioni visitatori-utenti.

Usability: reason why⁹³

1. *User friendly*
Progettare per l'utente.
2. *Effectiveness*
Raggiungere gli obiettivi aziendali.

93 NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di VANINI, W., Apogeo, Milano, 2000.

3. *Fidelization*
Far vivere un'esperienza positiva agli utenti.
4. *Think Different*
Differenziarsi sul mercato.
0. *Maturity*
Far crescere l'azienda.
0. *Innovation*
Cliente maturo, mercato maturo.

Ma quali sono i vantaggi che derivano dall'usabilità?⁹⁴

Per le aziende produttrici:
 incremento della qualità dei prodotti;
 incremento della vendite;
 incremento della produttività attraverso:
 riduzione dei tempi di sviluppo,
 riduzione degli interventi e dei costi di manutenzione,
 riduzione degli interventi e dei costi di assistenza (ad esempio, dei call center),
 riduzione dei costi della formazione per gli utenti finali,
 riduzione della documentazione utente (manuali e guide operative) e dei tempi per la sua realizzazione.

Per gli utenti finali i vantaggi si evidenziano in termini di:
 incremento della produttività attraverso:
 riduzione dei tempi di esecuzione delle attività,
 riduzione degli errori e dei tempi di recupero degli errori,
 riduzione dei tempi per richiedere supporto e assistenza,
 riduzione dei tempi di apprendimento (formazione, consultazione dei manuali);
 incremento della soddisfazione;
 incremento della qualità del lavoro, anche attraverso la riduzione della fatica e dello stress.

George M. Donahue e altri della Compuware Corporation, presentano nel documento "*Usability is a good business*"⁹⁵ un rapporto costi/benefici per gli studi di usabilità di 1 a 10-100: vuol dire che per ogni dollaro speso per implementare l'usabilità, si realizza un beneficio tra 10 e 100 dollari.

⁹⁴ BIAS, R., and MAYHEW, D. J., *Cost-Justifying Usability*, AP Professional, Cambridge, MA, 1994.

⁹⁵ DONAHUE, G., WEINSCHEK, S., NOWICKI, J., *Usability Is a Good Business*, Compuware Corporation, 1999.

Altri casi sono sul sito della Sun Microsystems⁹⁶: si dimostra che l'esecuzione di una analisi di usabilità, il cui costo si aggira in media sui 20,000 dollari, determina un risparmio economico di 153 milioni di dollari in termini di riduzione dei costi di call center e aumento della produttività degli utenti.

⁹⁶ Gli studi di usabilità della Sun Microsystems, sono disponibili on line su <http://www.sun.com/usability/>

6. Chi: le nuove figure professionali

Per condurre indagini di usabilità, bisogna disporre di uno staff professionale specializzato: un gruppo di persone che sono addestrate per programmare, disporre, effettuare, analizzare e riportare i risultati dei test. Sono previste figure professionali specifiche: questa divisione dei compiti viene intesa per poter registrare il maggior numero possibile di informazione. In ogni caso è possibile che anche solo un paio di persone possano svolgere le varie funzioni necessarie.

I diversi ruoli⁹⁷

Test monitor

Il *test monitor* è a capo del *team* di progetto e sviluppo del sito e partecipa alla definizione del test. E' la figura chiave attorno alla quale ruota tutto il test; svolge anche il *brief* e il *debrief* con i partecipanti al test.

Data logger⁹⁸

Ha il compito di classificare i vari eventi ed attività e codificarle entro determinate categorie. Ad esempio, se il partecipante avrà accesso durante il test a quattro differenti schermate, compito del *data logger* sarà quello di definirle con le lettere A, B, C, D. Durante il test, quando il partecipante accederà alla schermata, il *data logger* dovrà semplicemente registrare la lettera corrispondente ed eventualmente annotare il tempo in maniera da coordinarsi con la registrazione della videocamera.

Timer⁹⁹

Hanno il compito di registrare il tempo necessario per lo svolgimento del test e delle singole attività; può essere fatto sia in tempo reale, sia riguardando la videoregistrazione. Questo ruolo può essere svolto anche dall'operatore video, ma è fondamentale che non venga svolto dal *test monitor* che potrebbe distrarsi dall'osservazione del comportamento del partecipante.

⁹⁷ DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland, OR, 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994, p.49-59; NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

⁹⁸ Colui che registra i dati.

⁹⁹ Cronometrista.

Operatore video

E' responsabile della registrazione dell'intero processo di test. Tra i suoi compiti possiamo annoverare:

- preparare l'equipaggiamento per le audio e video registrazioni;
- valutare e ottimizzare l'angolo di ripresa delle videocamere per visualizzare sia il partecipante che il prodotto.

Durante il test, l'operatore ha il compito di seguire il flusso del test ed eventualmente concentrare la ripresa sulle attività critiche. Dopo il test sarà invece responsabile dell'archiviazione, del montaggio e della conservazione del girato. Nel caso non si abbiano a disposizione grandi risorse, le figure di *timer*, *data logger* e operatore video potranno coincidere.

Esperto

E' un individuo esperto del prodotto che deve essere testato. Per esempio, se il sistema si blocca e mette a rischio l'intero test, qualcuno dovrà essere in grado di ripristinare il sistema e fare un *back up* del lavoro svolto. Questo aspetto diventa cruciale quando si va a testare un prodotto ancora in fase di sviluppo con numerosi *bug*. Questo ruolo deve essere presente in ogni sessione di test ed eventualmente può essere coperto dal *test monitor*.

Ruoli aggiuntivi

Possono dipendere dalla natura del test e dalle esigenze. Ad esempio, si dovrà essere in grado di simulare l'operatore di un *call center* di supporto o il commesso di un negozio o il tecnico inviato dalla casa produttrice.

Osservatori

Non è un ruolo di per sé. Può essere svolto da chiunque stia assistendo al test, come ad esempio membri del gruppo di sviluppo, *project manager* o anche collaboratori dell'azienda che ha commissionato il test. Questa possibilità di assistere ai test può essere particolarmente di aiuto, poiché le professionalità che successivamente saranno impegnate nello sviluppo e nelle modifiche del prodotto possono da subito rendersi conto delle prospettive future; inoltre permette anche, vedendo gli utenti all'opera, di accettare di buon grado modifiche al prodotto sviluppato.

Ruolo del test monitor¹⁰⁰

Questo ruolo è fondamentale per tutto il *team*. E' il responsabile dell'intero test e dunque deve essere sempre presente in ogni momento della procedura. Dovrà curare ogni aspetto: accogliere e istruire i partecipanti, assicurarsi della loro attendibilità e, nel caso in cui questa venga meno, interrompere il test per non inficiare il risultato finale.

Chi deve svolgere questo ruolo?

E' impossibile rimanere obiettivi testando il proprio prodotto. E' troppo forte la tendenza a guidare i partecipanti nella direzione che vogliamo; è come pretendere dai genitori un giudizio obiettivo sulle capacità del figlio. Certamente, nel caso in cui non si abbia un disposizione un *team* per il test, è meglio testare da soli il proprio prodotto piuttosto che non testarlo affatto. Eventualmente il produttore può collaborare per lo sviluppo del materiale e delle procedure di test.

Le competenze del test monitor¹⁰¹

Specialista di fattori umani

Uno specialista in fattori umani, con una buona formazione di psicologia, design industriale o discipline simili e con una certa familiarità con le procedure sperimentali, rappresenta la professionalità migliore per il ruolo di *test monitor*. Questa formazione in psicologia cognitiva e analisi delle informazioni permette al *test monitor* di differenziare le osservazioni superficiali da quelle cruciali.

Specialista di marketing

Tipicamente *customer-oriented* e *user-oriented* con una buona capacità di comunicazione interpersonale; è una professionalità che può aiutare molto il miglioramento del prodotto e, sommando anche le altre competenze, può diventare un buon *test monitor*.

Comunicatori tecnici

Sicuramente una delle professionalità più indicate per il ruolo sono i comunicatori e i formatori tecnici che già sono abituati a pensare come utenti per definire e presentare le procedure e i materiali di supporto.

100 RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59.

101 DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland, OR, 1993.

Rotazione dei ruoli

Un'alternativa può essere quella della rotazione dei ruoli all'interno del *team*, il che permette una crescita professionale dei vari soggetti. Inoltre è possibile che, mentre una persona disegna e pianifica i test, il ruolo di sorveglianza venga svolto da un'altra: il test sarà così il prodotto di due persone.

Consulenti esterni

Questa soluzione può essere utile nel caso in cui si voglia creare un laboratorio interno all'azienda a lungo termine. Delle professionalità esterne potrebbero addestrare il personale per la costruzione ex-novo di un laboratorio e delle procedure.

Caratteristiche di un buon test monitor¹⁰²

Indipendentemente da chi conduce il test, interno o esterno, e indipendentemente dal percorso formativo, esistono delle caratteristiche chiave per il ruolo; nel caso in cui si decida di diventare *test monitor*, queste sono le capacità che si dovranno acquisire.

Formazione di base per la usability engineering

Conoscenze più o meno approfondite nel trattamento delle informazioni, della psicologia cognitiva, dello *user centered design*, del design delle interfacce aiutano moltissimo poiché contribuiscono a prevedere quali compiti o eventi potrebbero presentare delle criticità e quali aspetti dell'interfaccia dovranno essere migliorati prima dei test.

Apprendere facilmente

Il test monitor deve essere in grado di apprendere facilmente e velocemente il contesto specifico dei prodotti da testare. Ad esempio, se l'oggetto dei test è un *software* di *project management*, il *test monitor* non deve essere necessariamente un esperto del prodotto, ma deve essere in grado di apprendere facilmente l'ambiente e le funzioni del prodotto nonché il suo mercato e il suo posizionamento.

Capacità di instaurare ottimi rapporti

Nel caso in cui si decida di testare il prodotto solo con cinque soggetti, anche se solo uno di questi si comporta in maniera non spontanea rappresenta il 20% in meno di informazioni. Ecco perché è fondamentale che il *test monitor* sia in grado da subito di

¹⁰² RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

instaurare un clima sereno e tranquillo con i partecipanti, capire da subito le loro eventuali difficoltà. Si deve garantire il comfort dei partecipanti.

Eccellente memoria

Qualcuno potrebbe pensare che, dato che le sessioni di test sono videoriprese, il *test monitor* non debba ricordarle. E' invece fondamentale ricordare certi aspetti dopo la fine del test: ad esempio, si dovrà notare se il soggetto svolgerà nel corso della sessione lo stesso compito in maniera diversa. Inoltre, cercare nel girato una particolare sequenza, occupa decisamente più tempo che non ricordarla. Ecco perché si dovrà utilizzare il girato solo come conferma delle situazioni critiche.

Ottimi ascoltatori

La capacità di ascoltare del *test monitor* si risolve nell'abilità di seguire ogni sessione di test con "orecchie nuove", senza essere influenzato dalle sessioni precedenti. La spiccata sensibilità dovrà essere utilizzata anche per comprendere il linguaggio non verbale e comprendere i segni razionali del comportamento del soggetto.

A proprio agio con l'ambiguità

L'usabilità non è una scienza esatta costituita solo da formule in bianco o in nero. Anche se i test verranno condotti nel massimo rigore scientifico possibile, difficilmente si otterranno risultati chiari ed estendibili a tutta la popolazione. Ecco perché il *test monitor* deve essere in grado di leggere nell'ambiguità.

Per esempio, prima del test, si è sicuri che esistono solo due modi per assolvere un compito: durante il test, il partecipante scopre quattro nuovi modi per lo stesso compito. Ci si può trovare anche nella situazione che alla fine dei test non c'è nessuno o più di un *layout* valido: il *test monitor* deve essere in grado di gestire queste situazioni.

Flessibilità

Un'altra caratteristica è quella di saper gestire tutti quegli eventi che si discostano dal piano originale, anche se le conseguenze sono estremamente negative. Per esempio, ci si può accorgere anche solo dopo due test che una certa interfaccia presenta gravi lacune; si deve quindi interrompere il test per non sprecare tempo risolvendo i problemi minori e si dovrà subito sviluppare una nuova versione del prodotto per i test successivi.

Capacità di concentrazione

I test di usabilità possono essere sicuramente noiosi; ecco perché è fondamentale mantenere la concentrazione a lungo nella stessa sessione e anche in successive sessioni anche molto simili.

Capacità di dare sguardi generali

Ci sono talmente tanti dati da registrare durante un test che il rischio è quello di perdersi nel particolare, ignorando lo sguardo generale o di lasciarsi influenzare dagli ultimi utenti osservati. Il *test monitor* dovrà quindi essere in grado di:

- saper collegare insieme vari commenti, dati e impressioni dal singolo test per arrivare a uno sguardo generale delle performance dei partecipanti;
- saper collegare le varie conclusioni delle differenti sessioni di test, arrivando a delle conclusioni generali su tutto il prodotto.

Capacità comunicative

La capacità comunicativa è fondamentale: da un lato ci si deve interfacciare con i componenti del gruppo di test e con i partecipanti; dall'altro lato si deve anche convincere il cliente della necessità di apportare modifiche al prodotto nel loro stesso interesse. Inoltre si devono anche avere buone doti di scrittura per la preparazione dei materiali di supporto e *debriefing* del test.

Capacità organizzative e coordinative

Il test di *usability* è un progetto nel progetto: ecco perché anche una semplice sessione comporta la gestione di un grande numero di piccoli particolari. Il *test monitor* è la figura di riferimento per tutti gli altri componenti del *team* e partecipanti al test.

Tipici problemi del test monitor¹⁰³

Esaminiamo ora quei comportamenti che il *test monitor* deve evitare e tutti quegli errori tipici della conduzione di una sessione sperimentale.

Condurre piuttosto che disporre

Comportamenti che conducono l'utente a un certo comportamento sono generati da quei *test monitor* troppo vicini al prodotto che inconsciamente forniscono indizi ai soggetti. Anche semplicemente il tono di voce, un cenno della testa, o il tipo di

103 NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

domande usate nel briefing e nel test possono influenzare il risultato finale; questa è la ragione per la quale la maggior parte dei test in ambito professionale viene condotta attraverso una stanza di controllo separata da quella di test; nonostante questo, suggerimenti inconsci sono possibili anche da qui. Per evitare questo comportamento, è bene rendere il proprio comportamento e l'ambiente più neutrale possibile.

Troppo coinvolto nella registrazione dei dati

Mentre il compito principale del test è quello di collezionare e registrare dati, l'atto stesso della registrazione non deve interferire con l'osservazione diretta. Il *test monitor* deve essere attento a registrare ogni comportamento, anche rischiando di non appuntarne qualcuno. Ecco il motivo per cui si utilizzano strumenti di registrazione e codificazione dei comportamenti quali videocamere, schede di registrazione dati e di controllo.

Comportarsi come ultra-competente

Questo accade quando *test monitor* e partecipante sono nella stessa stanza: se i partecipanti lo vedranno troppo disponibile, tenderanno ad affidarsi troppo alle sue risposte e a rivolgergli molte domande; altrimenti, di fronte a un *test monitor* ultracompetente, potrebbero intimidirsi, specialmente gli utenti meno evoluti. Per evitare questi comportamenti è importante che il *test monitor* si limiti semplicemente a collezionare dati; il comportamento dei soggetti cambierà e si farà più intraprendente se sapranno di non poter ricevere più assistenza.

Troppo rigido con il test plan

Un esperto *test monitor* sa quando è il caso di deviare dal test plan, ad esempio perché il soggetto non si comporta come previsto; è importante saper definire dove il progetto del test presenta lacune ed apportare le giuste modifiche.

Incapace di relazionarsi con ogni partecipante

I partecipanti apparterranno ai profili più diversi. Il *test monitor* deve saper instaurare una relazione con tutti; un buon *test monitor* ha sempre chiaro che il partecipante ha sempre ragione.

Saltare alle conclusioni

I *test monitor* inesperti tendono a giungere subito alle conclusioni anche solo dopo pochi test iniziali. Questa è la ragione per cui si tende a testare vari soggetti: per avere un giudizio da vari profili, ognuno con comportamenti e formazione diversa.

Gestire lo stress

Un'altra caratteristica molto importante che deve possedere il *test monitor* è quella di saper gestire la frustrazione e il nervosismo dei partecipanti. Molti *test monitor*, quando percepiscono che un utente sta diventando nervoso perché non riesce a svolgere un compito, saltano direttamente al compito successivo per evitare che il soggetto si senta frustrato o sotto esame; passare oltre a delle criticità che svelano evidentemente i lati deboli di un'interfaccia può non essere una scelta felice: il comportamento dell'utente in questo momento è più rivelatore di qualsiasi altro momento e può aiutare il *team* a comprendere come il partecipante apprende l'utilizzo dell'interfaccia. E' importante che il *test monitor* incoraggi l'utente a completare il compito: il punto sta nel comprendere fino a quando incoraggiare il soggetto senza frustrarlo, comprendere cioè quando il soggetto è frustrato ma ancora volenteroso ad andare avanti e provare. Un'altra caratteristica importante per incoraggiare i partecipanti sta nel fatto che, per il cliente, assistere alla frustrazione e al disorientamento di un utente di fronte all'interfaccia può essere più convincente per la modifica del prodotto, piuttosto che discussioni di settimane con il gruppo di test.

Ma come incoraggiare gli utenti? Una maniera potrebbe essere quella di stabilire un rapporto empatico: "Mi rendo conto quanto sta diventando frustrante l'utilizzo del prodotto ma, poiché si tratta di un aspetto cruciale, potrebbe provare ancora un po?". Si può anche svelare che altri partecipanti hanno riscontrato lo stesso tipo di difficoltà: "Ho notato che ha delle difficoltà. Altri utenti hanno incontrato lo stesso tipo di problemi: posso chiederle di continuare altri 5 minuti?".

I migliori *test monitor* sono in grado di incoraggiare il lavoro con un sorriso. L'esperienza insegnerà il punto esatto nel quale incoraggiare il soggetto o passare al successivo compito.

Come migliorare le abilità di un test monitor¹⁰⁴

Condurre un test di usabilità è un compito estremamente difficile e composito. A un primo livello si deve lavorare a fianco di altre persone per migliorare un prodotto. A un livello più profondo è un'esperienza profonda che costringe a confrontare, senza influenzarli, il proprio modo di pensare con altri modi diversi. Nel caso in cui si voglia diventare un *test monitor* ecco come migliorare le proprie qualità.

104 RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

Imparare i principi base dei fattori umani e dell'ergonomia

Imparare le basi di discipline come l'elaborazione umana delle informazioni, la psicologia sperimentale, la psicologia cognitiva, la statistica, il design delle interfacce e l'ingegneria dell'usabilità.

Imparare osservando gli altri

Ovviamente osservare altri *test monitor* a lavoro può aiutare a imparare come svolgere al meglio il proprio compito. Dall'altro lato si cerchi sempre un contributo originale e l'esplorazione di nuove tecniche.

Osservarsi nella ripresa

Uno dei benefici di riprendere la sessione con la videocamera consiste nell'aver la possibilità di rivedere i propri comportamenti avendo la possibilità così di migliorare.

Pratica

Non cercare di essere perfezionista, ma partire con il giusto spirito e la voglia di imparare: all'inizio si faranno molti errori, si riveleranno informazioni e indizi agli utenti e si inventeranno nuovi metodi per invalidare una sessione di test.

Abituarsi alla consapevolezza

All'interno della propria routine giornaliera sarebbe importante avere la possibilità di dedicare 15-30 minuti per espandere la presenza a se stessi: concentrarsi su un compito molto semplice come, ad esempio, battere le dita sui tasti; provare a concentrarsi sulla sensazione delle dita che colpiscono i tasti e sforzare la propria mente a concentrarsi solo su questo evento; nel caso in cui ci si distraiga riportarla gentilmente al suo compito¹⁰⁵.

Questa pratica aiuterà a mantenere la concentrazione necessaria per la definizione delle procedure, per le sessioni di test e per il lavoro di analisi dei dati.

Selezionare i partecipanti al test¹⁰⁶

Questa fase è cruciale per i test, perché si devono scegliere con cura dei partecipanti rappresentativi. Selezionare i partecipanti comporta anche identificare e descrivere le

105 DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR, 1993.

106 NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993; NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

loro conoscenze principali e la loro competenza con il prodotto che andranno a testare.

Anzitutto, si dovranno caratterizzare gli utenti; per farlo si potrà seguire questo schema:

- storia personale
- età
- sesso
- attitudine verso la tecnologia
- modalità d'apprendimento (leggi e esegui, prova e poi leggi, impara facendo)
- formazione
- titolo di studio
- altri studi
- esperienza informatica
- tempo di utilizzo medio
- frequenza d'uso
- tipi di periferiche utilizzate
- OS utilizzato
- interfaccia utilizzata
- applicazioni utilizzate
- esperienza del prodotto
- tempo di utilizzo totale
- frequenza d'uso
- tipo di compiti eseguiti e frequenza
- occupazione

Questa scheda, non solo permette di conoscere l'utente e di dividere l'insieme di utenti in gruppi, ma permette anche di definire meglio il probabile utente finale medio. Infatti gli sviluppatori, solitamente, tendono a immaginare l'utente a cui viene rivolto il prodotto come molto simile, per capacità e conoscenze, a sé stessi.

Quindi, anche grazie all'aiuto di altri strumenti (ricerche di mercato, specifiche di progetto...) saremo in grado di definire l'utente medio a cui è rivolto il prodotto.

Decidere il numero dei partecipanti al test¹⁰⁷

Il numero dei partecipanti è in funzione di alcuni elementi:

¹⁰⁷ BIAS, R., MAYHEW, D. J., *Cost-Justifying Usability*, AP Professional, Cambridge, MA, 1994; tradotto da <http://www.useit.com/alertbox/> (visitato il 15 03 2001), sito ufficiale di Jakob Nielsen.

- il grado di attendibilità dei risultati;
- le risorse;
- la disponibilità di partecipanti dal profilo richiesto;
- la durata della sessione di test;
- il tempo necessario per approntare i test.

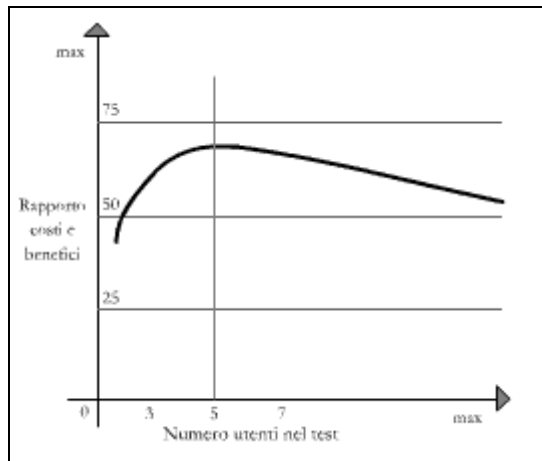


FIGURA 3 Il numero utenti test in funzione del rapporto costi benefici; 5 utenti la soluzione più conveniente. *Tratto da NIELSEN, J., and MACK, R. L., Usability Inspection Methods, John Wiley & Sons, New York, NY, 1994; RUBIN, J., Handbook of Usability Testing, John Wiley and Sons, New York, NY, 1994; BIAS, R., and MAYHEW, D. J., Cost-Justifying Usability, AP Professional, Cambridge, MA, 1994.*

In generale, con quattro o cinque partecipanti si è in grado di scoprire la maggior parte di problemi di usabilità in un tempo e con delle risorse ragionevoli. Se si hanno a disposizione risorse superiori, è preferibile condurre i test con otto partecipanti, suddivisi in due gruppi da quattro: “primo test> ridefinizione interfaccia> secondo test”. E' sempre buona norma includere una piccola parte di utenti esperti, anche se il profilo dell'utente finale non prevede utenti esperti. Questo perché un prodotto deve essere testato comunque da gruppi diversi.

Qualche parola sull'uso di utenti interni

Spesso vengono utilizzati per il test di prodotti, utenti interni alla società che ha sviluppato il prodotto. E' un metodo sicuramente comodo, ma sarebbe preferibile non utilizzarli come utenti principali del test. Ci sono altri momenti in cui possono essere utilizzati con profitto:

- testare il test : l'attenzione non è concentrata sulle scoperte di *usability*, ma per accertarsi della completezza del test stesso;

- condurre test esplorativi: per testare la struttura generale del prodotto (modello concettuale, tecniche di navigazione...) è possibile utilizzare utenti interni che non abbiano familiarità con il prodotto;
- per condurre i test con gli "utenti migliori": soprattutto nelle fase iniziali di sviluppo del prodotto, lo si può testare con degli utenti esperti sia del prodotto che dell'area di quel prodotto.

Ovviamente, se questi stessi utenti dovessero incontrare gravi problemi, l'interfaccia avrebbe bisogno di un serio ripensamento. Se invece l'interfaccia passasse il test con l'utente esperto, si potrebbe passare a testare con i nuovi utenti o gli utenti base per fare delle piccole correzioni.

In ogni caso è necessario scegliere gli utenti interni nei dipartimenti non familiari con il prodotto.

Per contattare gli utenti

Per prendere contatto e chiamare gli utenti per i test è possibile rivolgersi a varie strutture:

- ricerche di mercato
- clienti utenti del prodotto
- personale dell'azienda
- università
- annunci sul giornale
- comunità di utenti
- amici

7. I laboratori di test dell'usabilità

Per molti, implementare un programma per il test della *usability*, è diventato l'equivalente del disporre di un laboratorio ultra specializzato, costoso e articolato. Per molte organizzazioni, il laboratorio di *usability* è diventato più importante dello stesso processo di test. Non essendo consapevoli che istituire un programma di ingegneria dell'usabilità significa soprattutto e come prima cosa un forte cambiamento della cultura dell'organizzazione, queste aziende hanno tralasciato la parte più importante. Questo approccio è sicuramente miope e superficiale e con un alto rischio di fallimento. Eguagliare il laboratorio con il processo rischia di ridurre tutto a una collezione di videocamere e registratori¹⁰⁸.

La zona dei test¹⁰⁹

In questa sezione verranno descritti 5 diversi ambienti di test, da uno molto semplice ed economico, a un altro sofisticato e costoso. Ognuno presenta vantaggi e svantaggi e ognuno possiede un suo particolare stile di test.

Stanza singola

La stanza singola è il più piccolo ambiente di test possibile. Dentro questa stanza saranno presenti sia il partecipante, sia il *test monitor* e gli osservatori. L'importante è che il *test monitor* sia a una distanza tale da essere presente per il partecipante, senza distrarlo e senza provocare ansietà.

108 RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

109 DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland, OR, 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994, p.49-59; NIELSEN, J., MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

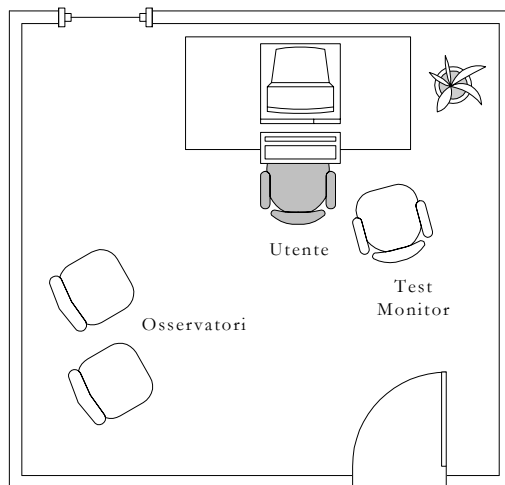


FIGURA 4 Laboratorio a stanza singola con osservatori. Elaborato da DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59; NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

L'implicazione fondamentale per questo tipo di test è data dal fatto che il *test monitor* potrebbe inavvertitamente influenzare il partecipante.

D'altra parte essere così vicino al partecipante diventa fondamentale per certi tipi di test che richiedono un'alta interazione tra utente e *test monitor*.

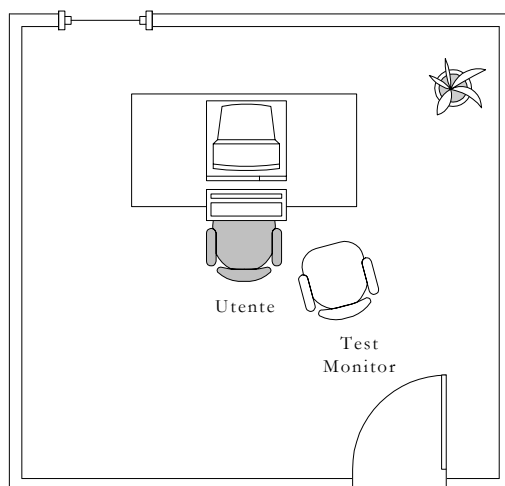


FIGURA 5 Laboratorio stanza singola senza osservatori. Elaborato da DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59; NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

Vantaggi

- il *test monitor* ha perfettamente sotto controllo il comportamento del partecipante, sia per il suo comportamento verbale, ma soprattutto per il suo comportamento non verbale;
- durante le prime fasi di sviluppo del prodotto, quando è necessaria una maggiore interazione fra i partecipanti, questa posizione accentua la sensazione di lavoro d'equipe. In questa fase della progettazione, il partecipante è per lo più un collaboratore al design del prodotto;
- diventa più facile e naturale per il partecipante, pensare a voce alta, piuttosto che parlare solo in una stanza ad alta voce.

Svantaggi

- il comportamento del *test monitor* può influenzare il comportamento del partecipante. Il partecipante potrebbe essere influenzato anche solo da una postura o da un movimento degli occhi. Inoltre si dovrebbe evitare di prendere appunti quando un compito viene effettuato, questo per non dare indicazioni all'utente e modificare il suo comportamento;
- c'è uno spazio limitato per gli osservatori; il numero degli osservatori è in funzione delle dimensioni della stanza: ovviamente è da evitare un alto numero di osservatori che potrebbero intimidire il partecipante.

Stanza singola modificata

La disposizione visualizzata nella FIG. 21, rispetto alla disposizione precedente, permette al *test monitor* di non invadere l'area del partecipante. Il *test monitor* è in grado di visualizzare il comportamento del partecipante via monitor; contemporaneamente è in grado di utilizzare del *software* per la registrazione dei dati senza essere nel campo visivo del partecipante, influenzando così il suo comportamento. Come per la disposizione precedente, il numero degli osservatori addizionali è dipendente dalla dimensione della stanza.

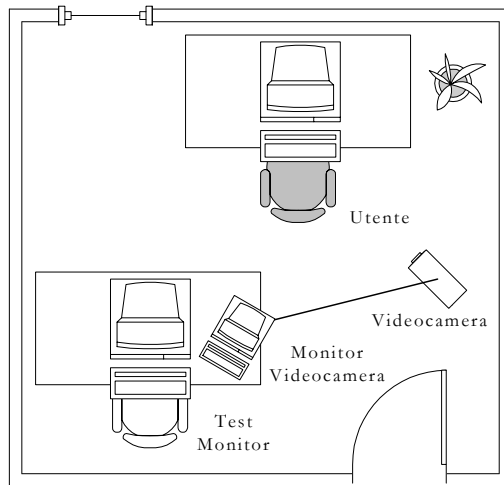


FIGURA 6 Laboratorio a stanza singola con videocamera. Elaborato da DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59; NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

Vantaggi

- il *test monitor* è più libero di muoversi e registrare dati mentre il test prosegue, stando in prossimità visiva con il partecipante e osservandone anche il linguaggio non verbale, senza influenzarlo;
- il partecipante non soffre di un senso assoluto di isolamento, poichè è presente nella stessa stanza il *test monitor*; questo facilita anche il processo di *thinking aloud*.

Svantaggi

- la perdita della prossimità con il partecipante limita la portata dei comportamenti non verbali osservabili. Per questo è fondamentale posizionare la videocamera mantenendo un angolo tale da permettere di vedere sia cosa l'utente sta eseguendo, sia come lo sta eseguendo;
- posizionare il *test monitor* dietro al partecipante entro un raggio di 10 metri potrebbe rendere il partecipante molto nervoso dandogli la sensazione di essere osservato. Le ricerche psicologiche sulla prossimità fisica ci indicano che, se ci si posiziona in un raggio di 10 metri, si deve essere presenti nel campo visivo del partecipante. Altrimenti ci si deve porre a distanze superiori;
- lo spazio per gli osservatori è limitato.

Stanza di osservazione elettronica

Questa disposizione permette di mantenere separati gli osservatori da un lato, il partecipante e il *test monitor* dall'altra. Ovviamente, la comunicazione tra osservatori e *test monitor* avverrà attraverso un auricolare non visibile per il partecipante.

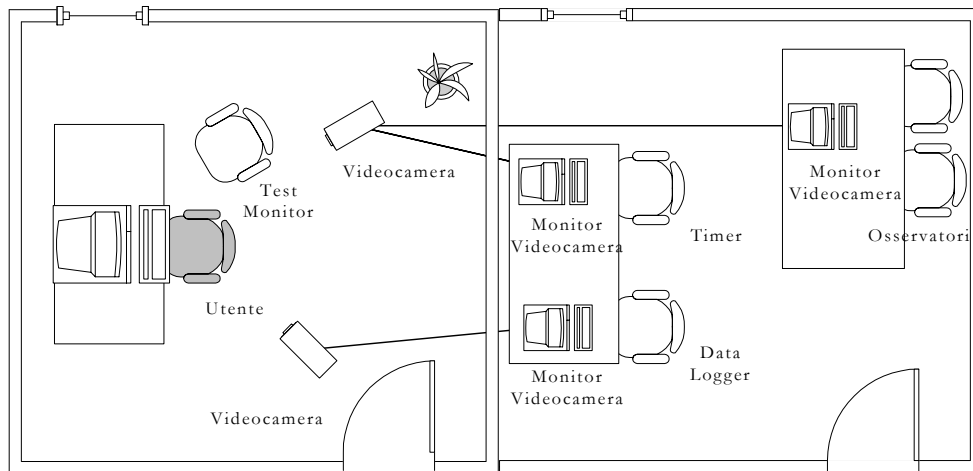


FIGURA 7 Laboratorio a due stanze con test monitor accanto all'utente. Elaborato da DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59; NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

Vantaggi

- tutti i vantaggi della stanza singola;
- gli osservatori possono esaminare i comportamenti del partecipante quanto desiderano senza aver timore di interferire con il suo comportamento.

Svantaggi

- il comportamento del *test monitor* può influenzare il partecipante;
- è logisticamente più impegnativa occupando due stanze.

Laboratorio classico

Questa disposizione prevede una stanza designata come stanza dei test e una seconda stanza per l'osservazione e il controllo. L'unica persona presente nella sala test è il partecipante. Tutto il personale restante è posizionato all'interno della stanza fornita di uno specchio unidirezionale. Le comunicazioni avvengono attraverso un sistema di altoparlanti.

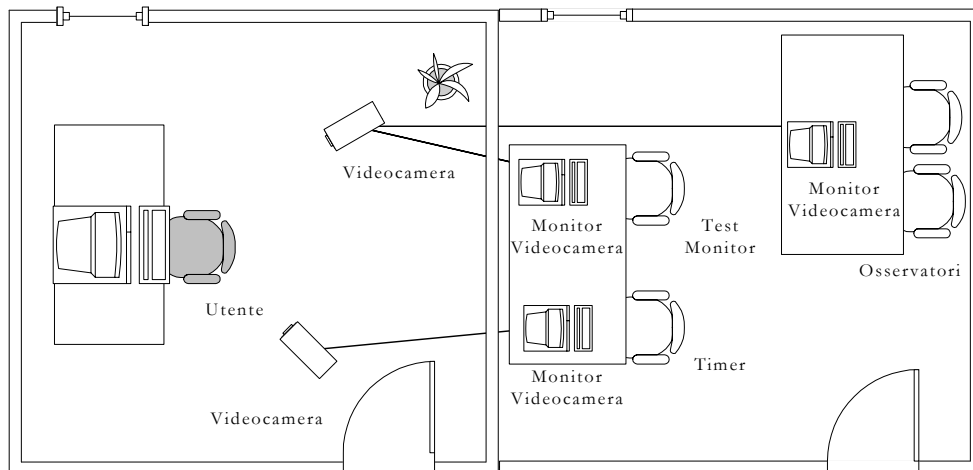


FIGURA 8 Laboratorio con due stanze, videoregistrazione, utente non affiancato dal test monitor. Elaborato da DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59; NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

A seconda del tipo di test, il *test monitor* può essere presente nella stanza insieme al partecipante.

Vantaggi

- poiché il *test monitor* non è nella stessa stanza, si eliminano possibili effetti di influenza;
- stando in una stanza isolata dai rumori, osservatori e *test monitor* possono comunicare tranquillamente ad alta voce;
- questo ambiente rende possibile a più osservatori di visionare un singolo *test monitor*.

Svantaggi

- in relazione all'abilità dello staff di test, questa disposizione potrebbe rendere troppo impersonale l'ambiente. Il partecipante potrebbe essere intimidito o addirittura impaurito dalla situazione, non comportandosi in maniera naturale. Addirittura la voce dell'altoparlante potrebbe apparire come una "voce dall'alto". Nel caso di utenti partecipanti sofisticati ed evoluti, questo effetto potrebbe essere

relativo, ma con utenti più timidi sarebbe il caso di avere il *test monitor* nella stessa stanza del partecipante;

- nonostante una disposizione delle videocamere ideale, il *test monitor* potrebbe non essere in grado di capire in quale parte della schermata sta guardando il partecipante. Ecco perché è fondamentale spingere il partecipante a pensare ad alta voce;
- questa disposizione viene sconsigliata nel caso di test esploratori nei quali è fondamentale l'osservazione dei comportamenti non verbali.

Laboratorio classico con specchio unidirezionale

Uguale alla disposizione precedente, ma con in più uno specchio unidirezionale che permette di seguire l'utente e il *test monitor*.

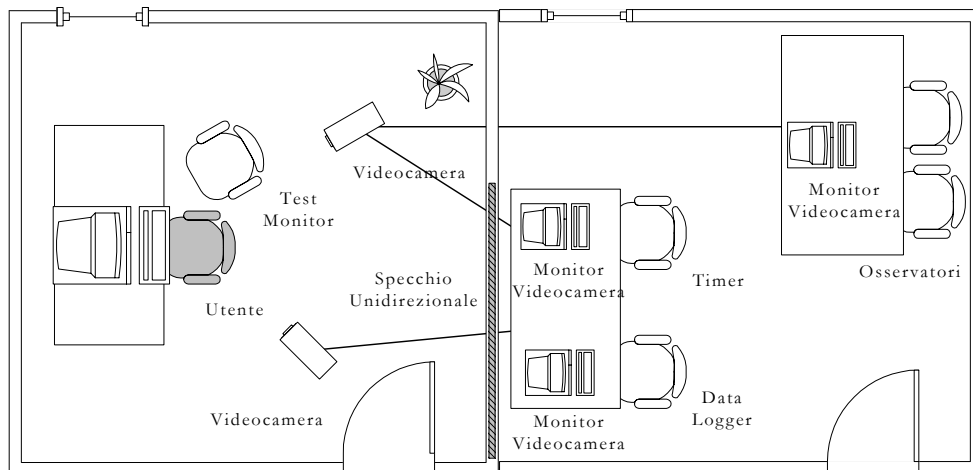


FIGURA 9 Laboratorio a due stanze, con specchio unidirezionale e test monitor accanto all'utente. Elaborato da DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59; NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

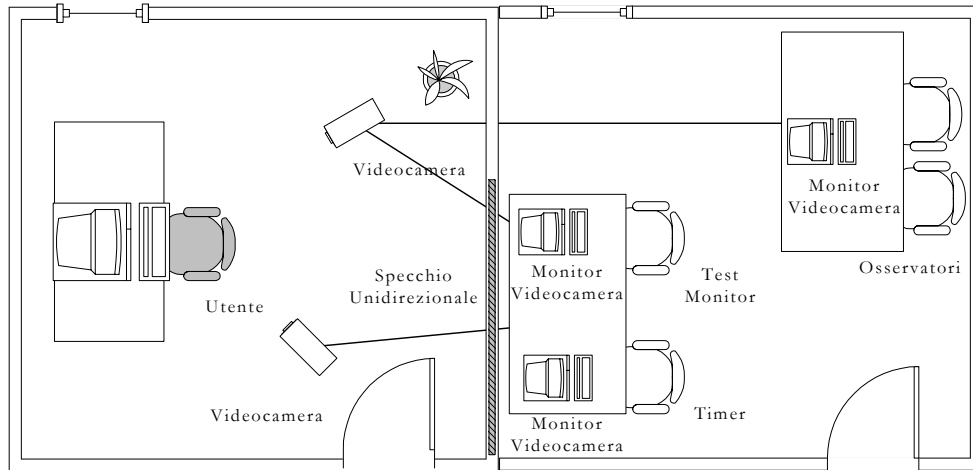


FIGURA 10 Laboratorio a due stanze, con specchio unidirezionale. Elaborato da DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993; RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY 1994, p.49-59; NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.

Laboratorio mobile

Nonostante l'equipaggiamento sia uguale a quello dei laboratori fissi, avendo la necessità di doverlo trasportare da un posto a un altro, si avrà cura di sceglierlo, piccolo, leggero e robusto. In ogni caso, non avendo la necessità di avere un stanza a disposizione, è sicuramente un modo economico per iniziare la sperimentazione.

Vantaggi

- e' sicuramente un soluzione economica poiché non ha i costi di logistica di un laboratorio fisso;
- si possono svolgere test sul campo, direttamente presso la sede del cliente.

Svantaggi

- ogni volta che si dovrà svolgere un nuovo test, si dovranno collaudare le apparecchiature; inoltre si deve ridefinire la loro posizione ogni volta.

Conclusione

Il compromesso migliore è rappresentato dalla stanza per l'osservazione elettronica. Questa disposizione fornisce il miglior compromesso fra interazione *test monitor* partecipante, disposizione degli osservatori e costi. Anche se sarebbe preferibile la

visione diretta del partecipante da parte degli osservatori, spesso è utile poter commentare a voce alta le azioni dell'utente.

Nel caso in cui l'azienda voglia dare vita a un programma su larga scala per i test, diventa forse preferibile una disposizione da laboratorio classica.

Bibliografia

La presente bibliografia raccoglie l'elenco delle fonti utilizzate per la stesura del volume.

L'elenco è presentato in ordine alfabetico per autore e, per ogni autore, le pubblicazioni sono presentate in ordine cronologico crescente.

A

- ANCESCHI, G., (a cura di), *Il progetto delle interfacce. Oggetto colloquiali e protesi virtuali*, Domus Academy, Milano, 1992-1993.

B

- BAECKER, R., & Buxton, W., *Readings in Human-Computer Interaction: A Multidisciplinary Approach*, Morgan-Kaufmann Publishers, Los Altos, CA, 1987.
- BAILEY, R., *Usability Testing versus Heuristic Evaluation: A Head-to-Head Comparison*. Proceedings of the Human Factors Society 36th Annual Meeting, 409-413, New York, 1992.
- BIAS, R., and MAYHEW, D. J., *Cost-Justifying Usability*, AP Professional, Cambridge, MA, 1994.
- BRIDGER, R. S., *Introduction to Ergonomics*, McGraw Hill, New York, 1995.
- BURGESS, J., *Designing for Humans: The Human Factor in Engineering*, Petrocelli Books, Princeton, NJ, 1986.
- BUSH, V., *As We May Think* in "Atlantic Monthly", luglio 1945 (tr. it in NELSON, T. H., *Literary Machine* 90.1. *Il progetto Xanadu*, Muzzio, Padova, 1992., p. 1/348-1/53).

C

- CARD, S., MORAN, T., NEWELL, A., *The Psychology of Human-Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale, NJ, 1983.
- CARLINI F., *Lo stile del web. Parole e immagini nella comunicazione di rete*, Einaudi, Torino, 1999.
- CASTELLANI, L., *La Tv dell'anno zero. Linguaggi e generi televisivi*, Edizioni Studium, Roma, 1995.
- CLAYTON L., RIEMAN, J., *Task-Centered User Interface Design: A Practical Introduction*, University of Colorado, Boulder, Colorado, 1993.
- CLEVELAND, W., *The Elements of Graphing Data*. Wadsworth, Monterrey, California, 1985.
- COOPER, A., *About Face: The Essentials of User Interface Design*, IDG Books, Foster City, CA, 1995.

D

- DE KERCKHOVE, D., *Brainframes. Tecnology, Mind and Business*, Bosch & Kerning, Utrecht, 1991, tr. it. di Bassi. B., *Brainframes. Mente, tecnologia e business*, Baskerville, Bologna, 1993.
- DUMAS, J., REDISH, J., *A Practical Guide to Usability Testing*, Intellect, Portland OR 1993.

F

- FLEMING, J., *Web Navigation: Designing the User Experience*, Songline Studios, Sebastopoli, CA, 1998.

G

- GRUDIN, J., BARNARD, P., *When does an abbreviation become a word and related questions*, ACM, New York, 1985.

H

- HACKOS, JO ANN, AND REDISH, GINNY, *User and Task Analysis for Interface Design*, John Wiley & Sons, Bethesda, MD, 1998.
- HELANDER, M., *Handbook of Human-Computer Interaction*, User Interface development, Amsterdam, 1988.

I

- ISO/IEC 9126 *Information technology - Software product evaluation - Quality characteristics and guidelines for their use*, 1992-1998.
- ISO 9241 *Ergonomic Requirements for Office Work with Visual Display Terminals*, Part 10, Dialogue principles, 1994.
- ISO 9241 *Ergonomic Requirements for Office Work with Visual Display Terminals*, part 11, Guidance on specifying and measuring usability, 1995.
- ISO 9241 *Ergonomic Requirements for Office Work with Visual Display Terminals*, Part 12, Ergonomic requirements for presentation of information, 1996.

J

- JOHNSON, S., *Interface Culture. How New Technology Transforms the Way We Create and Communicate*, HarperCollins, San Francisco, 1997.

K

- KAHN, P., KRZYSZTOF L., *Mapping Websites*, New Riders, Indianapolis, 2000.

- KAHN, P., KRZYSZTOF L., *Website Information Architecture*, New Riders, Indianapolis, 1998.
- KUNKEL K., BANNERT M., FACH W., *The influence of design decisions on the usability of direct manipulation user interface*. Behaviour and Information Technology, 1995, vol. 14, no. 2, 93-106.

L

- LANDOW, G. P., DELANY, P. (a cura di), *The Digital Word: Text-Based Computing in the Humanities*, MIT Press, London, 1993.
- LANDOW, G. P. (a cura di), *Hyper/text/theory*, Johns Hopkins University Press, Baltimore, 1994.
- LANDOW, G.P., *Hypertext 2.0. The Convergence of Contemporary Critical Theory and Technology*. Johns Hopkins University Press, Baltimore, 1997 (tr. it. MUSUMECI, V., L'ipertesto. Tecnologie digitali e critica letteraria, Bruno Mondadori, Milano, 1998).
- LAUREL, B., *Computers As Theater*, Addison-Wesley, New York 1991.
- LAUREL, B., *The Art of Human-Computer Interface Design*, Addison-Wesley, Reading, MA, 1990.
- LEE, A., *Nynex Portholes: Initial User Reaction*, ACM SIGGROUP, New York, 1997.
- LEVIALDI, S., *Interazione Uomo Macchina*, dispense del corso, Roma, 2000.
- LÉVY, P., *Il virtuale*, 1995, tr. It. M. Colò e M. Di Sopra, Raffaello Cortina, Milano, 2000.
- LÉVY, P., *Cybercultura. Gli usi sociali delle nuove tecnologie*, 1997, tr. It. B. Combatti, il Saggiatore, Milano, 1997.
- LEVY, P., *L'intelligenza collettiva. Per una antropologia del ciber spazio*, Feltrinelli, Milano, 1996.
- LEWIS, C., POLSON, P., *Theory Based designs for easily-learned interfaces*, Human Computer Interaction, Bethesda MD, 1998.
- LICKLIDER, J., *Man Coputer Symbiosis*, Electronics HPE, Boston, n.1 Marzo 1960.
- LIN X., CHOONG, Y., SALVENDY G., *A proposed Index of usability: a method for comparing the relative usability of different software systems*. Behaviour and Information Technology, 1997, vol. 16, n. 4/5, 267-278.
- LINDGAARD, G., *Usability Testing and System Evaluation: A Guide for Designing Useful Computer Systems*, Chapman and Hall, London, U.K. 1994.
- LUCCHINI, A., *Business Writing*, Sperling & Kupfer, Milano, 2001.

M

- MANTOVANI G., *L'interazione Uomo - Computer*. il Mulino, Bologna, 1995.

- MARCUS, A., *Graphic Design for Electronic Documents and User Interfaces*, Addison-Wesley Publishing Co. (ACM Press), Reading, MA 1992.
- MARRADI, A., *Concetti e metodi per la ricerca sociale*, La Giuntina, Firenze, 1995.
- MAYHEW, D., *The Usability Engineering Lifecycle: A Practitioner's Handbook for User Interface Design*, Morgan Kaufmann Publishers, San Francisco, 1999.
- MICROSOFT CORPORATION (TANDY TROWER) *The Windows Interface Guidelines for Software Design*, Microsoft Press, Redmond, WA, 1995.
- MORRIS, G., DILLON A., *The importance of usability in the establishment of organizational software standards for end user computing*. International Journal of Human - Computer Studies, 1996, vol. 45, 243-258.

N

- NIELSEN, J., *Coordinating User Interfaces for Consistency*, Academic Press, Boston, 1990.
- NIELSEN, J., *Designing User Interfaces for International Use*, Elsevier Science Publishers, Amsterdam, 1990.
- NIELSEN, J., *Hypertext and Hypermedia*, Academic Press, Boston, 1990.
- NIELSEN, J., *Usability Engineering*, Academic Press, Boston, 1993.
- NIELSEN, J., and MACK, R. L., *Usability Inspection Methods*, John Wiley & Sons, New York, NY, 1994.
- NIELSEN, J., *Advances in HCI Vol. 5*, Ablex, Norwood, NJ, 1995
- NIELSEN, J., *Multimedia and Hypertext: The Internet and Beyond*, AP Professional, Boston, 1995.
- NIELSEN, J., del Galdo, E., *International User Interfaces*, John Wiley & Sons, New York, NY, 1996.
- NIELSEN, J., *Designing Web Usability: The Practice of Simplicity*, New Riders Publishing, Indianapolis, 2000, tr. it. di Vanini, W., Apogeo, Milano, 2000.
- NIELSEN, J., and Tahir, M., *Homepage Usability: 50 Websites Deconstructed*, New Riders Publishing, Indianapolis, 2001.
- NORMAN, D.A., *Models of human memory*. Academic Press, New York, 1970.
- NORMAN, D.A., *Learning and Memory*, W.H. Freeman, San Francisco, 1982, tr. it. Di Valenitni Migliari, R., *Memoria e Attenzione*, Milano, Angeli, 1985.
- Norman, D. A., & Draper, S., *User Centered System Design: New Perspectives on Human-Computer Interaction*, Lawrence Erlbaum Associates, Hillsdale NJ, 1986.
- NORMAN, D.A., *The Design of Everyday Things*, Basic Books, New York, NY, 1988.
- NORMAN, D.A., *The Psychology of Everyday Things*, Basic Books, New York, NY, 1988, tr. it. DI NOFERI, G., *La caffettiera del masochista*, Giunti, Firenze 1997.

- NORMAN, D.A., *Turn signals are the facial expressions of automobiles*, Addison-Wesley Reading, MA 1992.
- NORMAN, D.A., *Things That Make Us Smart: Defending Human Attributes in the Age of the Machine*, Addison-Wesley Publishing, MA, 1993.
- NORMAN, D.A., *The invisible computer*, MIT Press Cambridge, MA 1998, tr. it. PARRELLA B. *Il computer invisibile*, Apogeo, Milano 2001.
- NEGROPONTE, N., *Being Digital*, New York, Alfred A. Knopf, 1995, trad. it. Di FILIPPAZZI, F., *Essere digitali*, Sperling & Kupfer, Milano 1995.

P

- PEDEMONTE, E., *Personal Media. Storia e futuro di un'utopia*, Bollati Boringhieri, Torino 1998.
- POLSON, P.G. AND LEWIS, C.H., *Theory-based design for easily learned interfaces*. *Human-Computer Interaction*, 5, 191-220, 1990.

R

- RHEINGOLD, H., *Virtual Reality*, Tauschstone, New York, 1991, tr. it. Di Saggini V., *La realtà virtuale. I mondi artificiali generati dal computer e il loro potere di trasformare la società*, 1991, Baskerville, Bologna 1993.
- RHEINGOLD, H., *The Virtual Community: Homesteading on the Electronic Frontier*, Revised Edition, Mit Press, Cambridge, MA 2000
- RIFKIN J., *The Age of Access*, Penguin, London, 2000, tr. it. Canton P., *L'era dell'accesso. La rivoluzione della new economy*, Mondadori, Milano, 2000.
- ROSENFELD, L., MORVILLE, P., *Information Architecture for the World Wide Web*, O'Reilly, Sebastapol CA, 1998.
- RUBIN, J., *Handbook of Usability Testing*, John Wiley and Sons, New York, NY, 1994.

S

- SANO D., *Designing Large-Scale Web Sites: A Visual Design Methodology*, Wiley, New York, 1996.
- SHNEIDERMAN, B., *Designing the User Interface. Strategies for Effective Human-Computer Interaction*, Addison-Wesley Publishing Co Reading, MA, 1992.
- SPOOL, J., *Web Site Usability: a Designer's Guide*, User Interface Engineering, North Andover, MA, 1997.
- SUTHERLAND, I., *Sketchpad: A Man Machine Graphical Communication System*, MIT, Boston, 1963.

T

- TOGNAZZINI, B., *Tog on Interface*, Addison-Wesley, Reading, MA, 1992.
- TOSELLI, L., *Il progettista multimediale*, Bollati Boringhieri, Torino, 1998.

Risorse web

Di seguito è proposto l'elenco delle fonti web utilizzate per la stesura del volume. L'elenco presenta i siti Web, corredati da una piccola spiegazione, in ordine cronologico crescente per data di visione.

- <http://www.yousable.com>
Società italiana specializzata in servizi di progettazione e valutazione di prodotti ipermediali con alte esigenze di usabilità.
- <http://www.macromedia.com>
Sito ufficiale della software house produttrice, tra gli altri, del tool di sviluppo Flash e dell'editor Html Dreamweaver.
- <http://www.sun.com/usability>
Sito della Sun Corporate, azienda produttrice di hardware e software, una delle prime ad occuparsi di usabilità a livello industriale. Jakob Nielsen effettuò le prime ricerche in collaborazione con la Sun.
- <http://www.adobe.com>
Sito ufficiale della software house produttrice dei programmi Photoshop per il fotoritocco, Illustrator, per l'illustrazione, Premier e After Effect, per il montaggio e produzione video.
- <http://www.informationweek.com/>
Magazine on line di informatica e nuove tecnologie.
- <http://hotwired.lycos.com>
Edizione on line della rivista di nuove tecnologie Wired.
- <http://www.wmtools.com> (Edizione on line della rivista di marketing e nuove tecnologie Web Marketing Tools.
- <http://www.useit.com>
Sito ufficiale di Jakob Nielsen.
- <http://www.microsoft.com/usability/>
Sezione del sito Microsoft specializzata in usability engineering.
- <http://www.ibm.com/usability/>
Sezione del sito Ibm specializzata in usability engineering.
- <http://www.k10k.net>
Uno dei siti della comunità on line degli sviluppatori Flash e Html.
- <http://www.flashkit.com>

Sito punto di riferimento per la comunità on line degli sviluppatori Flash.

- <http://www.zdnet.com/devhead/>

Sezione del portale USA dedicato agli sviluppatori per il Web.

- <http://www.eu.microsoft.com/usability/>

Sezione del sito Microsoft Europa specializzata in usability engineering.

- <http://www.usableweb.it>

Sito italiano dedicato all'usabilità.

- <http://www.informatique.it/sie/lecht.html>

Il sito della Società Italiana di Ergonomia

- <http://www.acm.org/sigchi/cdg/cdg2.html>

Curricula for Human Computer Interaction

- <http://www.ucc.ie/hfrg>

Il sito Web dello Human Factors Research Group dell'Università di Cork (Irlanda)

- <http://www.nomos.se/>

Il sito Web della Nomos Management (Svezia)

- <http://www.cs.umd.edu/projects/hcil/>

Il sito Web dello Human-Computer Interaction Lab (HCIL) dell'University del Maryland

- <http://web.tiscalinet.it/userware>

Sito italiano dedicato all'usabilità.

- <http://www.upassoc.org>

Sito della Usability Professional's Association.

- <http://www.hyperlab.net/ergonomia/>

Sito italiano che si occupa di ergonomia cognitiva.

- <http://www.acm.org/sigchi/>

Sito del Special Interest Group on Computer-Human Interaction.