

Interazione Uomo Macchina

L'Interazione

Struttura dell'interazione di D. Norman: pone particolare attenzione alla progettazione di oggetti usabili dall'utente. L'interazione viene descritta in termini di obiettivi ed azioni.

L'interazione è la comunicazione tra utente e sistema: l'utente stabilisce obiettivi, formula le intenzioni, specifica le azioni su un'interfaccia, esegue le azioni, percepisce e interpreta lo stato del sistema e lo valuta rispetto agli obiettivi.

Scopo del software è aiutare l'utente a **raggiungere** i suoi **obiettivi** nel dominio applicativo.

Il **modello di Norman** si basa sul susseguirsi di **due fasi**:

1. **esecuzione**: l'utente formula un piano d'azione e lo esegue mediante l'interfaccia;
2. **valutazione**: l'utente osserva l'interfaccia in modo da valutare, rispetto ai suoi obiettivi e intenzioni, il risultato del piano eseguito e determinare la successiva azione.

Questo modello si concentra solo su interfacce utente visuali.

Vengono applicati i seguenti concetti:

- **golfo dell'esecuzione**: è la differenza tra le azioni formulate dall'utente per raggiungere l'obiettivo e le azioni permesse dal sistema. Infatti le azioni formulate dall'utente possono essere diverse da quelle permesse dal sistema.
 - Più è piccolo più l'interazione è efficace.
- **golfo della valutazione**: è la differenza tra la presentazione fisica dello stato del sistema e le aspettative dell'utente. il cambiamento di stato previsto dall'utente può essere diverso da quello presentato dal sistema
 - Più è piccolo minore sarà lo sforzo richiesto all'utente.

Pregi: introduce il concetto di fruibilità e di visibilità

Difetti:

- troppo concentrato sull'utente;
- non considera il punto di vista del progettista e quindi non tiene presente vincoli dovuti alla macchina;
- analizza l'interazione a livello locale e non globale.

Struttura dell'interazione di Abowd e Beale:

Composta da *quattro parti* ognuna delle quali ha un suo linguaggio:

- utente;
- input;
- sistema;
- output;

vi è quindi la **necessità di tradurre tali linguaggi**, la cosa tuttavia può risultare difficile o impossibile.

Le *intenzioni* dell'utente sono tradotte in *azioni* dall'interfaccia e provocano un'*alterazione* dello stato del sistema; queste *alterazioni* sono *espresse* attraverso dispositivi di output che verranno *interpretati* dall'utente.

Quindi il modello ha **quattro fasi**:

1. **articolazione**: l'utente articola il compito da svolgere nel linguaggio di input;
2. **prestazione**: l'azione viene tradotta dal sistema dal linguaggio di input nel linguaggio macchina;
3. **presentazione**: il sistema passa ad un nuovo stato dopo l'esecuzione, tale nuovo stato viene comunicato tramite il linguaggio di output;
4. **osservazione**: l'utente interpreta il risultato con l'osservazione;

Pregi: considera il punto di vista del progettista;

Difetti:

- la divisione tra input e output risulta artificiosa;
- non tiene conto dei vincoli HW e SW;
- è statico

Ergonomia: studio delle caratteristiche fisiche (fattori umani) dell'interazione. Serve a definire e progettare strumenti che:

- si adattino alle caratteristiche fisiche dell'utente;
- si integrino nell'ambiente in modo da massimizzare: comodità, efficienza e soddisfazione dell'utente;

Stili di Interazione: modalità con cui l'utente può interagire:

- **interfacce a riga di comando**
 - vantaggi: potente, flessibile, veloce, efficiente, occupa poco spazio a video, sotto il controllo dell'utente;
 - svantaggi: difficile da ricordare e da imparare, facilità di errore, velocità di battitura, aggiornamento non visibile, training;
- **menù**
 - vantaggi: facile da imparare e da ricordare, poca competenza informatica, mostra tutte le funzionalità, poco training, non richiede abilità di battitura;
 - svantaggi: occupa molto spazio video, sotto il controllo del sistema, poco flessibile, non molto potente, il compito deve essere strutturato;
- **linguaggio naturale**: problemi nell'attribuzione di un significato che non sempre è univoco
 - vantaggi: semplice da capire e ricordare, poca competenza informatica, flessibile, non intimidatorio, nessun training;
 - svantaggi: dipende dalla tecnologia, vi sono difficoltà nel generare dialogo e nel comprenderlo;
- **domanda/risposta**
 - vantaggi: autoesplicativo, facile da imparare, ricordare e utilizzare, non intimidisce l'utente, nessun training;
 - svantaggi: inefficiente, sotto il controllo del sistema, facilità di errore, abilità di battitura, poca flessibilità, difficoltà di formulazione domanda;
- **query di dialogo**
 - linguaggio SQL e quindi la complessità dipende dalla base di dati;
- **riempimento di moduli**
 - vantaggi: fortemente autoesplicativo, poco carico cognitivo, ottimizza l'uso di spazio video, specifica un contesto, l'aggiornamento è visibile, è facile da capire;
 - svantaggi: facilità di errore, abilità di battitura, poco flessibile, funzionalità speciali non evidenti, può intimorire l'utente;

- **fogli di calcolo**
 - vantaggi: autoesplicativo, semplice, non intimorisce l'utente, funzionale e flessibile, programmabile;
 - svantaggi: facilità di errore, training, abilità di battitura, tempi specifici;
- **tastiere di funzioni**: alla pressione di un certo tasto corrisponde l'esecuzione di un certo algoritmo associato ad un compito;
 - vantaggi: poco spazio video, funzionale, poco training, facile da imparare e ricordare, specifica un contesto;
 - svantaggi: funzionalità limitate, aggiornamento dipendente dall'HW, sotto il controllo del sistema, poco flessibile;
- **WIMP** (Windows Icons, Menus, Pointers): vedi dopo

La Manipolazione Diretta:

Una buona interfaccia utente porta ai seguenti benefici:

- padronanza dell'interfaccia;
- competenza nell'eseguire compiti;
- facilità nell'apprendere il sistema;
- mantenimento della padronanza nel tempo;
- soddisfazione nell'uso del sistema;
- desiderio di provare gli aspetti più avanzati del sistema;

Problemi con la manipolazione diretta:

- rappresentazioni spaziali o visuali possono essere troppo grandi;
- flow-chart o schemi di database possono risultare confusi;
- alcune informazioni importanti possono rimanere fuori dallo schermo;
- necessità per gli utenti di imparare le rappresentazioni grafiche;
- le rappresentazioni grafiche possono essere equivoche;
- digitare un comando su tastiera può essere più veloce;

Manipolazione diretta e interfacce WIMP:

- **Finestre** (Windows): sono aree dello schermo che si comportano come aree virtuali indipendenti dal terminale;
 - può contenere testo o oggetti grafici;
 - può essere spostata o ridimensionata;
 - più finestre possono affiancarsi, sovrapporsi, oscurarsi a vicenda;
 - le scrollbar consentono di spostare il contenuto della finestra;
 - la barra del titolo fornisce una descrizione della finestra;
 - l'utente deve poter lavorare nelle diverse aree ed avere l'impressione di svolgere lavori diversi simultaneamente, senza curarsi dei segnali dell'interazione;
- **Icone**: sono piccole immagini stilizzate o realistiche usate per rappresentare alcuni oggetti dell'interfaccia, le finestre possono essere iconificate, le icone possono essere hot-spot di collegamento a spazi di schermo più grandi.

Linee guida:

- rappresentare oggetti/azioni in modo familiare;
- limitare il numero di icone diverse;
- far emergere le icone dallo sfondo;
- considerare l'impiego di icone tridimensionali;
- l'icona selezionata deve apparire diversamente dalle altre;
- se vi sono animazioni curarle;

- aggiungere dettagli;
- valutare l'uso di combinazioni di icone per creare nuovi oggetti/azioni;
- **Puntatori**: è un'icona, tipicamente una piccola freccia che indica dove si stà puntando con il mouse.
- **Menù**: scelta di operazioni o servizi che possono essere eseguiti e offerti attraverso lo schermo, richiede la selezione mediante il puntatore. Possono essere fissi o a scomparsa. Nelle finestre vi possono inoltre essere i **Widgets**: sono componenti dell'interfaccia messi a disposizione dell'progettista per migliorare la fruibilità:
 - **Bottoni**: aree di schermo che se selezionate invocano un'azione;
 - **Palette**: insieme di possibili modalità e la modalità corrente (es colori o forme);
 - **Finestre di dialogo** (dialog box): finestre usate per segnalare eventi importanti o richiedere dati;
 - **Tavolozze, cursori, barre di scorrimento**: usati per esprimere chiaramente il modo con cui si può compiere un'azione.

Vantaggi:

- pochi errori, facile da imparare, poco training, le operazioni sono spesso reversibili, flessibile, feedback istantaneo, intuitivo;

Svantaggi:

- può essere inefficiente, è difficile produrre icone riconoscibili, occupa spazio video.