

Che cos'è un interfaccia? E i suoi tipi

È un meccanismo che permette il dialogo tra due entità.

Interfacce CODE-BASED

Si ha un'interazione attraverso dei codici; ottimo per moli di lavoro elevate, tuttavia:

- occorre mantenere basso il numero di codici utilizzabili
- nessuna riusabilità

Mole di lavoro da svolgere	Qualità	Facilità di apprendimento	Riutilizzo conoscenza	Soddisfazione
				

```
> copy utenti.txt D:
> print utenti.txt
> delete utenti.txt
```

Interfaccia 3270

Ottimo per data-entry (da documenti cartacei, ad esempio); ho un workflow predefinito.

Mole di lavoro da svolgere	Qualità	Facilità di apprendimento	Riutilizzo conoscenza	Soddisfazione
				

NOME: _____
 COGNOME: _____
 SESSO: _____
 RESIDENZA: _____

PSEUDO GUI

Sono interfacce grafiche che richiamano la struttura di un'interfaccia a caratteri.

È ottima per applicazioni che devono gestire dati fortemente strutturati.

Anagrafica Clienti

Codice:

Cognome:

Nome:

Ragione:

Società:

Indirizzo:

C.A.P.: Provincia:

Città:

Tipo:

☐ Potenziale

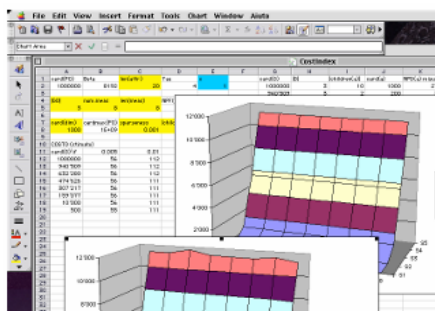
☐ Acquisito

Settore Mercatologico:

OK Cancel

STANDARD GUI

Progettata e sviluppata per un ambiente grafico; esalta la manipolazione (copia e incolla, taglia ecc..) e ottima per applicazioni user-driven



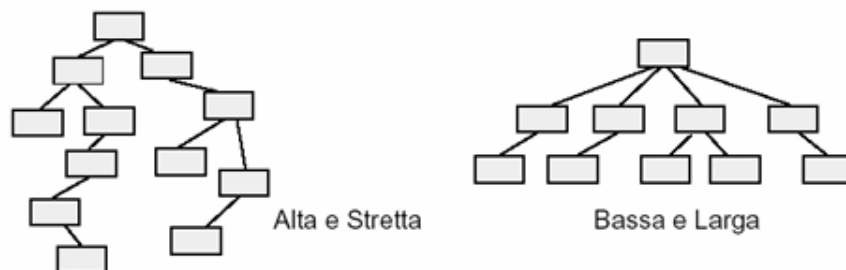
SPECIAL GUI

Interfacce pensate per ambiti applicativi particolari; obiettivo è l'auto esplicazione, ovvero senza manuale devo essere capace in poco tempo di utilizzare l'interfaccia. Esempio: totem per acquisto di biglietti.



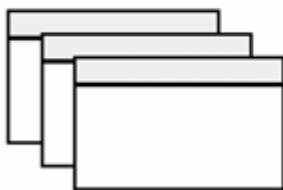
STRUTTURAZIONE

- Una struttura “bassa e larga” fornisce all’utente una visione migliore delle possibilità offerte e facilita la navigazione

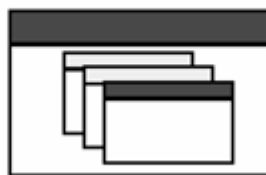


- alta e stretta: poche scelte (vantaggio) ma ho tanti click da fare
- bassa e larga: faccio in fretta ad arrivare in fondo (vantaggio) ma potrei perdere tempo a fare una scelta (contro)

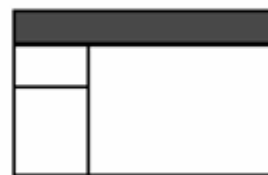
Multi-Window



Multi-Document (MDI)



Multi-Paned



- multi-windows: ogni finestra al suo interno mantiene un documento
- multi-document: una sola finestra con menù
- multi-paned: una sola window alla volta con o senza menù; quindi ho un finestra divisa in pannelli

PROJECT STANDARD

La definizione degli standard riguarda la terminologia, l'uso di metafore e icone e le caratteristiche delle finestre, come menu, bottoni, dimensioni e posizionamento.

L'obiettivo prioritario è agevolare l'utilizzo da parte dell'utente, garantendo innanzitutto la **consistenza esterna** attraverso l'allineamento ai tool già utilizzati in azienda, che rappresentano standard di fatto. La consistenza interna rimane importante, ma è subordinata all'usabilità complessiva del sistema.

Comunicazione visiva nelle GUI

- **L'affordance** mette in evidenza le caratteristiche di un oggetto che suggeriscono come utilizzarlo; disegnare qualcosa che faccia capire all'utente meglio il significato;
- La **metafora**, attraverso parole, frasi o immagini, rappresenta un oggetto o un concetto richiamando uno del mondo reale per facilitare la comprensione.
- Il **layout** definisce la disposizione di testo, immagini e controlli all'interno di uno spazio
- I **colori** aiutano a guidare l'attenzione e a creare associazioni significative.
- Le **icone** sono elementi grafici piccoli, semplici e di tipo metaforico. Le icone possono essere desktop, menù oppure button (come nel caso di excel).
- I **font** contribuiscono alla leggibilità in base alla tipologia e alle caratteristiche del carattere scelto.

Usabilità: è il grado di efficacia, efficienza e soddisfazione con cui specifici utenti svolgono determinati compiti in particolari contesti. L'efficacia indica quanto i compiti previsti vengano portati a termine correttamente, l'efficienza riguarda le risorse necessarie per eseguirli, mentre la soddisfazione misura il livello di accettabilità e gradimento del sistema da parte dell'utente.

Apprendibilità: ha come obiettivo fare in modo che l'80% dei nuovi utenti sia in grado di svolgere correttamente una singola attività dell'applicazione entro 30 minuti. Questo aspetto diventa particolarmente rilevante in contesti caratterizzati da alto turn-over, presenza di utenti saltuari e necessità di ridurre il training. È fondamentale anche quando i sistemi risultano spesso sottoutilizzati per mancanza di formazione o quando i prodotti hanno un ciclo di vita breve.

Velocità: ha come obiettivo consentire 10 inserimenti ogni due minuti. Questo requisito diventa centrale nei casi di utilizzo quotidiano e intensivo del sistema, soprattutto quando le attività da svolgere sono ripetitive.

Soddisfazione: ha come obiettivo che 9 utenti su 10 dichiarano che il sistema è "bello da usare". Questo aspetto è particolarmente importante nei sistemi self-service e nei progetti di Business Process Re-engineering fortemente incentrati sul nuovo sistema.

Facilità di navigazione: ha come obiettivo consentire l'avvio di fino a sei diverse attività a partire da un singolo oggetto, senza dover tornare al menu principale. Questo requisito è particolarmente rilevante quando il cliente "guida il gioco", quando esistono forti collegamenti tra le attività o quando si è in attesa di una decisione, come nel caso di un ristorante. È inoltre importante in contesti con elevato turn-over o quando l'obiettivo principale non è prendere decisioni, ma seguire una procedura ben definita, come in una mensa.

Memorabilità: ha come obiettivo quello di riutilizzare senza difficoltà un'applicazione inattiva da 12 mesi.

Metodo di progetto

La metodologia di progetto dell'interfaccia prevede che, già prima della conclusione dello studio di fattibilità, vengano definite le attività legate alla sua realizzazione, i parametri di riferimento e i criteri di usabilità, pianificate le attività di valutazione e costruito il modello concettuale dell'interfaccia.

Precocemente nella fase di analisi e progettazione si procede poi alla definizione delle strutture di base, come dialogo e look & feel, alla definizione degli standard di progetto, alla prototipazione delle parti critiche e alla verifica della coerenza con il modello concettuale e gli standard stabiliti.

Infine, nella fase di sviluppo, l'interfaccia viene completata nel dettaglio e integrata con la logica applicativa.