

[Prefazione](#)

[01 - Introduzione al corso](#)

[02 - Introduzione alla UUX](#)

[03 - Design - System-centered design e User-centered design](#)

[04 - ISO 9241-210 - Two user-oriented process models](#)

[05 - Psicologia cognitiva](#)

[06 - User Experience Design – Parte I](#)

[07 - User Experience Design – Parte II](#)

[08 - User Experience Design – Parte III](#)

[09 - Architettura Dell'informazione \(Information Architecture\)](#)

[10 - La Valutazione Dell'usabilità: L'ispezione](#)

[11 - La valutazione dell'usabilità: Il Testing](#)

[12 - Prestazioni Umane \(Human Performances\)](#)

[L'Occhio Umano e la Percezione Visiva](#)

[I Fotorecettori](#)

[13 - The Evaluation - Inspection](#)

[14 - The Evaluation - Testing](#)

Prefazione

Il documento che segue rappresenta una risorsa completa e dettagliata per la preparazione all'esame di Usability & User Experience Design (UUX), basata sulle lezioni tenute dal Prof. Fabio Vitali per l'anno accademico 2024/2025.

Questo lavoro di trascrizione, sintesi e organizzazione è stato curato con l'obiettivo di fornire agli studenti uno strumento di studio organico e di facile consultazione. La raccolta non si limita a una semplice trascrizione delle slide, ma le arricchisce con appunti e approfondimenti che ne chiariscono i passaggi chiave e i concetti fondamentali.

All'interno, il lettore troverà un percorso strutturato che copre l'intero programma del corso:

- Una trascrizione ragionata dei capitoli, che guida passo dopo passo attraverso i temi cardine della disciplina: dai principi di psicologia cognitiva di Donald Norman, agli approcci di design User-Centered, fino ai modelli di processo come l'ISO 9241-210 e il framework dei cinque piani di Jesse James Garrett.
- Una mappa concettuale che offre una visione d'insieme, aiutando a collegare i diversi argomenti e a comprendere la struttura logica del corso.
- Una raccolta di possibili domande d'esame con risposte articolate, pensate per offrire un banco di prova concreto e per consolidare la preparazione in vista della prova finale.

Questi appunti sono pensati per essere un alleato per lo studio individuale e di gruppo: un punto di partenza per approfondire, un riferimento per ripassare e una guida per affrontare con maggiore sicurezza e consapevolezza l'esame.

01 - Introduzione al corso

Questo capitolo funge da introduzione generale al corso "Introduction to the UUX course" per l'anno accademico 2024-25, tenuto dal Prof. Fabio Vitali.

Presentazione del Corso: Introduzione al tema generale della User and Usability eXperience (UUX).

Lingua del Corso: Il corso è tenuto in inglese, con possibili spiegazioni iniziali in italiano.

02 - Introduzione alla UUX

Introduzione alla Lezione

Questo capitolo getta le basi per la comprensione della User and Usability eXperience (UUX). Inizia illustrando la necessità di un buon design attraverso esempi di "cattiva progettazione" in vari contesti (oggetti, etichette, percorsi, pulsanti, messaggi, task, interfacce). Successivamente, si focalizza sulle giustificazioni per l'usabilità e l'UX, sull'evoluzione della terminologia chiave, sulla distinzione tra diversi livelli di progettazione (dalla funzionalità ai servizi), e sui principi fondamentali che guidano un design centrato sull'utente.

1. Giustificazioni per l'Usabilità e l'UX

La necessità di progettare software e sistemi usabili non è solo una questione di estetica o preferenza, ma trova fondamento in diverse aree:

- **Autorità:** Direttive europee (es. 90/270/EEC) richiedono che il software sia adatto al compito, facile da usare, adattabile, fornisca feedback e segua i principi dell'ergonomia software.
- **Business:** I costi umani sono superiori a quelli di hardware/software. Un buon software valorizza il capitale umano, mentre errori umani dovuti a cattiva progettazione generano costi (tempo, denaro, soddisfazione, morale, vite).
- **Mercato:** Gli utenti si aspettano software facili da usare e sono meno tolleranti verso carenze di design.

- **Individui:** I computer sono visti come elettrodomestici da cui ci si aspetta affidabilità e usabilità.
- **Etica e Società:** I computer sono parte critica della società, usati in ambiti rilevanti (educazione, dati personali, operazioni critiche).
- **Sfide di Design:** L'interazione tra la complessità umana e quella dei sistemi rappresenta una sfida progettuale interessante.

2. Evoluzione della Terminologia Chiave

La disciplina si è evoluta, e con essa la sua terminologia:

- **Human Performance (Inizio XX sec.):** Applicazione del taylorismo, l'uomo-lavoratore visto come una macchina da ottimizzare.
- **Ergonomics (II Guerra Mondiale, UK):** Progettazione di macchine (armi) che sfruttano al meglio le caratteristiche fisiche umane.
- **Human Factors (Anni '60, USA):** Simile all'ergonomia, ma con un'enfasi aggiuntiva sugli aspetti cognitivi.
- **Man-Machine Interaction / Human-Computer Interaction (HCI) (Anni '70-'80):** Focus sull'usabilità dei dispositivi computazionali, evolvendo da "Uomo-Macchina" a "Persona-Elaboratore" per maggiore correttezza politica e riconoscimento del ruolo centrale dei computer.
- **User Interface (UI):** Il punto di contatto e interazione tra utente e applicazione.
- **Web Design / Web Usability:** Nascita di discipline specifiche per i siti web, con focus su grafica, contenuto e, grazie ad autori come Jakob Nielsen, sull'applicazione delle teorie di usabilità al web.
- **Interaction Design (IXD):** Interesse multidisciplinare alla progettazione dell'interazione tra persone e computer.
- **User Experience Design (UXD):** Enfasi sulla soddisfazione complessiva dell'utente, oltre la semplice usabilità, con forti influenze dal marketing.
- **Design Thinking (DT):** Termine generico (non solo software) per analizzare e strutturare il processo di design e la mentalità di un designer di successo. Spesso, oggi, i termini IxD, UXD e DT sono usati quasi come sinonimi, con una crescente centratura sull'utente.

3. Dalla Funzionalità ai Servizi: Livelli di Progettazione

L'esempio di un'automobile illustra bene i diversi livelli:

- **Design della Funzionalità:** Lo chassis, il motore, le componenti meccaniche che permettono all'auto di funzionare.

- **Design dell'User Interface (UI):** Il cruscotto, il volante, i pedali, i comandi – ciò con cui l'utente interagisce direttamente per controllare le funzionalità.
- **Design dell'User Experience (UX):** L'esperienza complessiva di guida e permanenza nell'auto: comfort dei sedili, spazio, estetica interna ed esterna, sensazioni durante l'uso.
- **Service Design:** L'intera esperienza connessa al possesso e all'uso dell'auto: il processo di acquisto in concessionaria, l'assistenza post-vendita, la pubblicità, le community online. È l'orchestrazione di molti fattori, anche non computazionali, focalizzata sui bisogni e le aspettative a lungo termine degli utenti.

4. Definizioni Fondamentali

- **User Experience (UX) (ISO 9241-210):** "Le percezioni e le risposte di una persona che risultano dall'uso o dall'uso anticipato di un prodotto, sistema o servizio." (Nielsen la estende a tutti gli aspetti dell'interazione con l'azienda). L'enfasi è sull'impressione soggettiva, dove l'usabilità è solo uno dei parametri.
- **Usability (ISO 9241-110):** "La misura in cui un prodotto può essere usato da utenti specificati per raggiungere obiettivi specificati con efficacia, efficienza e soddisfazione in un contesto d'uso specificato."
 - **6 Aspetti Chiave (ISO 9241-110):** Design basato su analisi esplicita di utenti/task/contesto; coinvolgimento utenti; design guidato da valutazione user-centred; processo iterativo; considerazione dell'intera esperienza; team multidisciplinare.
- **Componenti dell'Usabilità secondo Jakob Nielsen:**
 - **Learnability (Apprendibilità):** Facilità con cui i nuovi utenti compiono task basilari al primo incontro.
 - **Efficiency (Efficienza):** Rapidità con cui gli utenti esperti eseguono i task.
 - **Memorability (Memorizzabilità):** Facilità con cui gli utenti occasionali ristabiliscono la competenza dopo un periodo di non utilizzo.
 - **Errors (Errori):** Numero, gravità e facilità di recupero dagli errori.
 - **Satisfaction (Soddisfazione):** Piacevolezza d'uso del design.

5. Principi e Approcci

- **I 4 Golden Rules della Progettazione per l'Usabilità:**
 1. **Pensa agli utenti:** Il designer NON è l'utente tipo.
 2. **Testa sul campo il sistema:** Il laboratorio non riflette l'uso reale.

3. **Coinvolgi gli utenti:** Hanno competenze non formalizzate preziose.
 4. **Itera:** Nessun design è perfetto al primo colpo; prototipi piccoli, economici ed expendable sono migliori di un unico sforzo costoso.
- **Design Partecipativo:** L'esempio della "siringa automatica" illustra come il coinvolgimento degli utenti (infermieri) nel processo di design porti a soluzioni radicalmente diverse e più adatte al contesto reale rispetto a un design concepito solo dai progettisti.
 - **Natura della UUX:** Non è una scienza esatta con teorie unificanti. È un parallelo con l'architettura: c'è una base scientifica (evitare il collasso), una professione (strutture, tecniche costruttive, costi) e un'arte (grazia, ispirazione, genio). Nel UUX, le teorie e i modelli evolvono; importante è distinguere tra *teorie* (regole permanenti), *paradigmi* (framework filosofici legati a rivoluzioni scientifiche/tecnologiche) e *regole* (norme specifiche, spesso dipendenti da trend).

03 - Design - System-centered design e User-centered design

Introduzione alla Lezione

Questo capitolo esplora i diversi approcci o "modelli" mentali che possono guidare la progettazione di un sistema interattivo. La distinzione fondamentale è tra un design centrato sul sistema (o sul progettista) e un design centrato sull'utente, quest'ultimo ulteriormente articolato in base al focus sui compiti o sugli obiettivi reali dell'utente.

1. System-Centered Design (Design Centrato sul Sistema)

Questo approccio basa le decisioni di progettazione su ciò che è conveniente per il progettista o per il sistema stesso.

- Caratteristiche:
 - Cosa è facile da progettare con la piattaforma disponibile?
 - Cosa è facile da creare con gli strumenti a disposizione?
 - Cosa il progettista trova interessante o appagante realizzare?
- Livello 0: Feature Oriented System (Sistema Orientato alle Funzionalità):

- L'interfaccia mostra tutto ciò che il sistema può fare, come un menu "à la carte" molto esteso.
- Spesso si traduce in sistemi con molte scelte, ma superficiali.
- Non richiede un'analisi degli utenti, ma solo delle caratteristiche e funzioni del sistema.
- **Metafora:** Un ristorante con una lista lunghissima di piatti, senza suggerimenti di abbinamento, dove puoi scegliere qualsiasi cosa in qualsiasi ordine.

2. User-Centered Design (UCD - Design Centrato sull'Utente)

Questo approccio pone al centro le necessità, le caratteristiche e il contesto delle persone che utilizzeranno il sistema.

- **Definizione:** Il design si basa sull'importanza dei bisogni degli utenti. Le caratteristiche intellettuali, culturali, motivazionali e fisiche dell'utente sono parametri fondamentali e non variabili da manipolare per ottimizzare costi o efficienza del sistema.
- Domande Chiave:
 - Cosa può fare l'utente?
 - Di cosa ha bisogno l'utente?
 - In quale contesto l'utente utilizza il sistema?
- **Interaction Design (IXD):** È la disciplina che applica i principi UCD alla progettazione di qualsiasi sistema (non solo IT).

Si divide in due modelli leggermente diversi:

- a) Task-Oriented Design (Design Orientato ai Compiti):
 - L'interfaccia è strutturata attorno ai compiti specifici che l'utente deve svolgere, piuttosto che attorno alle singole funzioni del sistema.
 - Il sistema non è solitamente molto esteso, ma può comunque presentare un numero considerevole di scelte.
 - Richiede una certa curva di apprendimento e la comprensione degli utenti e dei loro compiti.

- **Metafora:** Un ristorante tematico, con uno o più menu fissi dove i piatti sono abbinati tra loro e associati a una stagione, un ingrediente o un "mood".
- b) Goal-Oriented Design (Design Orientato agli Obiettivi):
 - Si concentra sull'obiettivo fondamentale e sulle reali motivazioni dell'utente, andando oltre i semplici compiti.
 - Elimina i task non rilevanti per l'obiettivo primario e si focalizza sulle ragioni per cui l'utente usa il sistema.
 - L'utente non ha bisogno di "imparare il sistema", ma solo di capire se attraverso di esso può raggiungere i propri scopi.
 - **Metafora:** Un ristorante che offre un'esperienza memorabile. La qualità del cibo è solo un elemento, insieme alla composizione dei piatti, all'ambientazione, all'intrattenimento, alla location, ecc. L'obiettivo non è solo "mangiare", ma "passare una bella serata".

3. Fasi del User-Centered Design

Gli approcci UCD sono quasi sempre organizzati nelle seguenti fasi iterative:

1. **Task and User Analysis (Analisi dei Compiti e degli Utenti):** Comprendere chi sono gli utenti, cosa fanno e perché.
2. **Design (Progettazione):** Creare soluzioni basate sull'analisi.
3. **Validation (Validazione):** Verificare se il design soddisfa i requisiti.
4. **Testing (Test):** Valutare il design con utenti reali.
5. **Iteration (Iterazione):** Rifinire il design basandosi sui feedback.

4. Due Modelli di Processo User-Oriented

Esistono modelli formalizzati che incarnano questi principi:

- Un Modello Task-Oriented

- **ISO 9241-210 (2010):** Standard internazionale (originariamente UK) focalizzato sulla progettazione dell'usabilità. Prevede cinque fasi: Studio di Fattibilità, Requisiti Utente, Implementazione, Valutazione, Distribuzione.
- Un Modello Goal-Oriented – *User Experience Design*:
 - **Jesse James Garrett (2011):** Modello di un noto professionista USA, diffuso tramite passaparola, mirato alla User Experience Design complessiva. Si articola in cinque "piani": Strategia, Scopo, Struttura, Scheletro, Superficie.

04 - ISO 9241-210 - Two user-oriented process models

Introduzione alla Lezione

Questo capitolo approfondisce uno dei principali modelli di processo orientati all'utente: lo standard ISO 9241-210, "Ergonomics of human-system interaction – Part 210: Human-centred design for interactive systems". Verrà esplorato il contesto di questo standard all'interno della famiglia ISO, i suoi principi chiave, il processo di design che delinea e le metriche di valutazione associate.

1. Contesto degli Standard ISO sull'Human-Centred Design (HCD)

Lo standard ISO 9241-210 si inserisce in un framework più ampio di normative ISO che coprono diversi aspetti dell'HCD:

- **Organisational Capability (Capacità Organizzativa):** La maturità di un'organizzazione nel portare avanti processi di design user-centred (es. ISO TR 18529).
- **Process Quality (Qualità del Processo):** Guide per l'interazione uomo-sistema durante il ciclo di vita dei sistemi interattivi (qui si colloca ISO 9241-210).
- **Product Quality (Qualità del Prodotto):** Specifiche per l'aspetto e il comportamento delle interfacce software (ISO 9241-100 series) e per l'hardware (ISO 9241-300 series).
- **Quality in Use (Qualità in Uso):** Valutazione dell'efficacia, produttività, sicurezza e soddisfazione quando utenti specifici usano il prodotto in contesti specifici (legato a ISO 9241-110 e ISO/IEC 9126). ISO/IEC 9126, ad esempio, definisce 6 categorie di qualità software: funzionalità, affidabilità, usabilità, efficienza, manutenibilità, portabilità.

2. Principi Chiave di ISO 9241-210

Questo standard descrive 6 principi fondamentali per garantire un design user-centred:

1. Il design è basato su una **comprensione esplicita degli utenti, dei loro compiti e degli ambienti d'uso**.
2. Gli **utenti sono coinvolti** durante tutto il processo di design e sviluppo (design partecipativo).
3. Il design è **guidato e rifinito da valutazioni user-centred**.
4. Il processo è **iterativo**.
5. Il design considera l'**intera user experience**.
6. Il team di design include **competenze multidisciplinari**.

3. I Principi di Dialogo (ISO 9241-110)

Strettamente correlata è la norma ISO 9241-110, che definisce 7 "principi di dialogo" per la progettazione dell'interazione uomo-macchina:

1. **Adeguatezza al compito (Suitability for the task)**: Il dialogo deve essere adatto al compito e al livello di abilità dell'utente.
2. **Autodescrittività (Self-descriptiveness)**: Il dialogo deve chiarire cosa l'utente deve fare dopo.
3. **Controllabilità (Controllability)**: L'utente deve poter controllare il ritmo e la sequenza dell'interazione.
4. **Conformità alle aspettative dell'utente (Conformity with user expectations)**: Il dialogo deve essere consistente.
5. **Tolleranza agli errori (Error tolerance)**: Il dialogo deve "perdonare" gli errori.
6. **Adeguatezza all'individualizzazione (Suitability for individualisation)**: Il dialogo deve poter essere personalizzato.
7. **Adeguatezza all'apprendimento (Suitability for learning)**: Il dialogo deve supportare l'apprendimento.

4. Il Processo di Design Human-Centred (ISO 9241-210)

Lo standard delinea un processo iterativo composto dalle seguenti attività principali:

1. **Pianificare il processo human-centred (Plan the human-centred process)**: Definire l'ambito e le risorse.
2. **Comprendere e specificare il contesto d'uso (Understand and specify the context of use)**: Identificare utenti, compiti, ambiente tecnico e fisico.

3. **Specificare i requisiti utente e organizzativi (Specify the user and organizational requirements):** Derivare i bisogni e tradurli in requisiti misurabili.

4. **Produrre soluzioni di design (Produce design solutions):** Creare prototipi e specifiche di design.

5. **Valutare il design rispetto ai requisiti (Evaluate designs against requirements):** Verificare se le soluzioni soddisfano i bisogni identificati.

Questo ciclo si ripete finché i requisiti non sono soddisfatti.

5. Dettaglio delle Fasi e Attività del Processo HCD

Il processo può essere mappato sul ciclo di vita di un sistema e dettagliato ulteriormente:

- **Fase di Fattibilità (Feasibility Study):**

- **Stakeholder Meetings:** Identificare obiettivi, stakeholder, criteri di successo, ruolo dell'usabilità.
- **Context of Use Analysis:** Definire utenti, compiti, vincoli tecnici/ambientali.
- **Scenarios:** Documentare come gli utenti dovrebbero svolgere i compiti in contesti specifici.

- **Fase dei Requisiti Utente (User Requirements):**

- **Evaluation of Existing Systems:** Analizzare l'usabilità di versioni precedenti o prodotti concorrenti.
- **Analysis of Logs:** Esaminare i pattern d'uso di sistemi esistenti.
- **Interviews:** Raccogliere informazioni dirette dagli utenti su bisogni, priorità, problemi.
- **Direct Observation (Ethnography):** Osservare gli utenti nel loro ambiente di lavoro reale.
- **Context of Use (dettaglio):** Produrre descrizioni documentate dei vincoli (utenti, task, tecnici, culturali, ambientali).
- **Scenarios (dettaglio):** Creare narrazioni delle azioni utente, decomponendo i task.
- **User Requirements Document:** Formalizzare i requisiti, inclusi criteri di verifica quantitativi.

- **Fase di Progettazione (Design):**

- **Identifying and Using Style Guides:** Adottare linee guida per la coerenza di screenful e dialoghi.
- **Prototyping:** Utilizzare prototipi (spesso lo-fi) per chiarire requisiti e iterare velocemente (concept design, interaction design, screen design, screen testing).
- **Fase di Valutazione (Evaluation):**
 - **Internal Prototype Inspection:** Metodi low-cost e low-reliability per identificare problemi potenziali senza utenti reali (es. heuristic evaluation, cognitive walkthrough).
 - **Usability Testing:** Test con utenti reali per misurare i risultati rispetto alle aspettative. Si distingue tra:
 - **Test Formativi:** Per identificare e risolvere problemi *durante* la fase di design.
 - **Test Sommativi:** Per valutare il sistema *nello stato finale* rispetto ai requisiti.
- **Fase di Rilascio/Distribuzione (Deploy):**
 - **Collection of Feedback:** Raccogliere informazioni da utenti non supervisionati sul sistema rilasciato (survey, helpline, forum, ecc.) per informare future release.

6. Metriche di Valutazione dell'Usabilità (Le Tre "E")

ISO 9241-210 e standard correlati (come ISO 9241-110) enfatizzano tre fattori principali per misurare la qualità in uso:

- **Effectiveness (Efficacia):** L'accuratezza e la completezza con cui l'utente conclude i compiti.
 - *Unità primaria:* Errori (numero e gravità).
 - *Esempi di metriche:* % task completati, % funzionalità usate, % task completati al primo tentativo, numero di errori persistenti/per unità di tempo/per task.
- **Efficiency (Efficienza):** Il rapporto tra accuratezza/completezza del compito e le risorse impiegate.
 - *Unità primaria:* Tempo.

- *Esempi di metriche:* Tempo per un task, tempo al primo tentativo, tempo per re-imparare funzioni, tempo per correggere errori, numero di tasti premuti.
- **Emotions (o Satisfaction - Soddisfazione):** La presenza (o assenza) di sentimenti di (in)soddisfazione e l'attitudine (positiva/negativa) percepibile nell'uso continuativo.
 - *Unità primaria:* Voto soggettivo (es. scala 1-10).
 - *Esempi di metriche:* Aggettivi positivi/negativi usati per descrivere il prodotto, % utenti che si sentono "in controllo", % utenti che lo ritengono più facile/soddisfacente/produttivo della concorrenza, % utenti che lo raccomanderebbero.

7. Checklist Finale (Esempio ISO 9241-210)

Una checklist può aiutare a verificare la completezza dell'analisi:

- **Analisi senza utenti:** Tutti i task rilevanti sono stati considerati e decomposti correttamente? Sono state considerate situazioni speciali ed errori? I task si riferiscono a problemi e non a soluzioni? Sono indipendenti e hanno una chiusura?
- **Analisi con utenti:** I task sono appropriati e completi? Le persone descritte negli scenari sono credibili e rappresentative? Le peggiori situazioni sono state considerate e gestite?

Conclusioni

Lo standard ISO 9241-210 fornisce un framework robusto e iterativo per l'Human-Centred Design. Enfatizza la comprensione profonda degli utenti e del loro contesto, il coinvolgimento attivo degli utenti nel processo, e la valutazione continua per guidare il design verso soluzioni efficaci, efficienti e soddisfacenti.

05 - Psicologia cognitiva

Introduzione alla Lezione

Benvenuti a questa lezione sulla psicologia cognitiva applicata agli oggetti di uso quotidiano. Oggi esploreremo come la nostra mente interagisce con gli oggetti che ci circondano, perché alcuni sono intuitivi e altri frustranti, e come una buona progettazione possa fare la differenza. Ci baseremo principalmente sugli studi e sui principi esposti da Donald Norman nei suoi celebri testi, "La psicologia degli oggetti quotidiani" (noto anche come "La caffettiera del masochista") e la sua più recente edizione "The Design of Everyday Things".

Gli argomenti principali che tratteremo sono:

- L'**uso degli oggetti** e i modelli mentali che ci formiamo.
- Gli **errori umani**: perché sbagliamo e come il design può influenzarli.
- La **conoscenza**: quella che abbiamo "nella testa" e quella presente "nel mondo".
- Principi di **progettazione degli oggetti quotidiani** per migliorarne l'usabilità.

1. Gli Oggetti Quotidiani e la Loro Complessità

Pensateci: un adulto medio usa e riconosce a prima vista oltre 20.000 oggetti. Alcuni, come le posate o la vostra tazza preferita, li usiamo ogni giorno e abbiamo sviluppato un **modello mentale** chiaro del loro funzionamento. Altri oggetti risultano semplici da usare perché offrono **indizi chiari** su come interagire con essi. Altri ancora, invece, appaiono irrimediabilmente complessi e oscuri. Perché questa differenza? La risposta risiede spesso nel loro design.

2. Progettare per Semplificare: Scoperta e Comprensione

Un buon design deve aiutare l'utente in due aspetti fondamentali:

- **Scoperta**: L'oggetto mi fa capire quali azioni sono possibili? Quando e dove devo eseguirle?
- **Comprensione**: Cosa fa questo oggetto? Come si aspetta che io lo usi? A cosa servono tutti questi comandi e impostazioni?

Il design definisce come gli oggetti funzionano e la natura dell'interazione tra persone e tecnologia. Le macchine, per loro natura, hanno un numero limitato di comportamenti previsti dal progettista. Se l'utente non segue questi comportamenti, la macchina può "fallire" e, troppo spesso, l'operatore viene incolpato. Questo può portare a semplice **frustrazione** con oggetti di uso quotidiano, ma con macchinari complessi o industriali, le conseguenze possono essere perdite economiche, incidenti, o persino fatalità.

3. L'Utente "Logico" e la Realtà del Progettista

I designer tendono ad avere menti logiche, attenzione ai dettagli e un forte desiderio di apprendere. Questo può portarli a credere, erroneamente, che tutti gli utenti siano altrettanto logici e attenti. Osservando un utente "comune" interagire con un oggetto da loro progettato, potrebbero chiedersi: "Ma cosa sta facendo? Perché lo usa in questo modo?".

È cruciale capire che dobbiamo progettare gli artefatti per **come sono realmente gli utenti**, non per come vorremmo che fossero. Dobbiamo partire dal presupposto che le persone commetteranno errori e progettare sistemi che tengano conto di questa fallibilità.

4. Principi Fondamentali dell'Interazione

L'interazione con un oggetto genera **esperienze**, e queste esperienze dovrebbero essere positive e piacevoli. Le esperienze sono fondamentali perché determinano:

- Come ricorderemo quell'interazione.
- Come approcceremo interazioni future con oggetti simili.
- Cosa ci aspetteremo da interazioni future in generale.

Una **mancanza di comprensione** genera confusione, frustrazione, rabbia o disinteresse. Al contrario, la **comprensione** genera un senso di controllo, padronanza, soddisfazione e persino orgoglio. È importante ricordare che **cognizione ed emozioni** sono strettamente connesse: non possiamo avere l'una senza l'altra. Un oggetto ben progettato non è solo funzionale, ma evoca anche emozioni positive.

5. Concetti Base del Design Cognitivo

Per comprendere meglio come progettare oggetti efficaci, introduciamo alcuni concetti base:

- **Affordance e Significanti:**

- L'**affordance** (o "invito all'uso") si riferisce alle proprietà percepite di un oggetto che suggeriscono come un utente possa interagire con esso. Una superficie piana (lastra) "invita" ad essere spinta, una manopola a essere girata, una fessura a inserirvi qualcosa. Ad esempio, la forma asimmetrica di un floppy disk da 3.5" (un classico!) permetteva un solo modo corretto di inserimento.
- A volte, però, l'affordance non è chiara o è nascosta (es. il vetro è trasparente ma blocca il passaggio). In questi casi, il design deve **segnalare** l'affordance: questo segnale è il **significante**. Un significante è un indizio esplicito (una freccia, un'etichetta, un suono) che comunica la presenza e la localizzazione di un'affordance. Le **affordance** determinano quali azioni sono *possibili*, i **significanti** comunicano *dove* queste azioni dovrebbero avvenire e come.

- **Modelli Concettuali:**

- Noi creiamo naturalmente un **modello concettuale** del funzionamento di un oggetto, basandoci su affordance, vincoli e correlazioni spaziali. Ad esempio, in una bicicletta, la sella, il manubrio e i pedali suggeriscono un'unica posizione corretta. Le forbici,

con il filo tagliente e gli anelli di diverse dimensioni, suggeriscono un solo modo d'uso.

- Un **controesempio** comune è il termostato: molti credono che alzare la temperatura al massimo faccia riscaldare la casa più velocemente, ma il termostato controlla solo la temperatura finale, non la velocità di riscaldamento. Questo è un modello concettuale errato.
- Il designer ha il compito di comunicare il modello corretto. Esiste un **Modello del Progettista**, che viene tradotto nell'**Immagine del Sistema** (l'oggetto fisico, la sua interfaccia, la documentazione). L'utente, interagendo con l'Immagine del Sistema, si costruisce il proprio **Modello dell'Utente**. Se l'Immagine del Sistema è chiara, il Modello dell'Utente si avvicinerà a quello del Progettista.

- **Mapping (Corrispondenza):**

- Il **mapping** è la relazione tra i comandi (controlli) e i loro effetti. Un buon mapping è naturale e intuitivo.
- Esempi di buon mapping: girare il volante a destra fa svoltare l'auto a destra; la disposizione dei comandi dei fornelli che rispecchia quella dei fornelli stessi.
- Esempi di cattivo mapping: una fila di interruttori identici per luci diverse senza un chiaro collegamento spaziale. I comandi del sedile dell'auto Mercedes, che mimano la forma del sedile, hanno un ottimo mapping. Altri sistemi usano pulsanti più astratti.

- **Feedback (Retroazione):**

- Il **feedback** è l'informazione di ritorno che permette all'utente di sapere che un'azione è stata eseguita e quale ne è stato il risultato. Deve essere **immediato, chiaro, non ambiguo e non opprimente**.
- Un buon feedback risponde a domande come: "La macchina è accesa?", "L'operazione è completata?".
- Un feedback scarso o assente genera ansia e incertezza.
- Il feedback può essere **eccessivo** (effetto "al lupo al lupo", dove troppi avvisi portano a ignorarli tutti) o **incomprensibile** (luci lampeggianti o suoni di cui non si capisce il significato).

- **Conoscenza nella Testa e Conoscenza nel Mondo:**

- La **conoscenza dichiarativa** è esplicita, conscia (fatti, regole). È facile da spiegare, ma spesso difficile da imparare e usare.

- La **conoscenza procedurale** è implicita, automatica, difficile da verbalizzare (andare in bicicletta, suonare uno strumento). È difficile da spiegare, ma facile da usare una volta appresa.
- Possiamo immagazzinare conoscenza **nella nostra testa** (es. password, PIN) o affidarci alla **conoscenza nel mondo** (es. etichette, istruzioni, design dell'oggetto stesso).
- La conoscenza nel mondo non richiede apprendimento, è recuperabile se visibile/udibile, ma può rallentare l'efficienza per la continua interpretazione e può portare a interfacce sovraccaricate.
- La conoscenza nella testa richiede apprendimento, il recupero può essere difficile, ma una volta appresa permette alta efficienza e design più puliti.
- Un buon design bilancia questi due tipi di conoscenza.

- **Vincoli (Constraints):**

- I vincoli limitano le possibili azioni, guidando l'utente verso l'interazione corretta. Esistono vari tipi:
 - **Fisici:** La forma fisica impedisce azioni errate (es. una chiave che entra solo nella sua serratura).
 - **Semantici:** Basati sul significato della situazione (es. sappiamo che il parabrezza va davanti in un modellino d'auto).
 - **Culturali:** Convenzioni sociali (es. il rosso significa stop o pericolo).
 - **Logici:** Relazioni logiche tra le parti (es. se avanza un pezzo nel montaggio di un mobile IKEA, qualcosa è andato storto).

6. Le Azioni: Struttura e Sette Stadi

I compiti quotidiani possono essere visti come alberi di sequenze di azioni con scelte a ogni bivio.

- **Compiti difficili** (es. scacchi) hanno strutture ampie e profonde.
- **Compiti semplici** possono avere strutture ampie e piatte (es. menu di un ristorante) o strette e profonde (es. seguire una ricetta).

Donald Norman descrive l'azione attraverso **sette stadi**:

1. Formulare l'**Obiettivo** (Goal).
2. Formulare l'**Intenzione** di agire (Plan).

3. Specificare la **Sequenza di Azioni** (Specify).
4. **Eseguire** la Sequenza di Azioni (Perform).
--- (Interazione con il Mondo) ---
5. **Percepire** lo Stato del Mondo.
6. **Interpretare** lo Stato del Mondo.
7. **Valutare** il risultato rispetto all'Obiettivo.

Questo è un modello idealizzato; nella realtà, le nostre azioni sono spesso **opportunistiche** e guidate dai dati, piuttosto che rigidamente pianificate.

Azioni Consce e Subconsce e i Tre Livelli Emotivi

Molte nostre azioni sono **subliminali**, automatiche. L'**overlearning** (sovra-apprendimento) rende le competenze automatiche e inconsapevoli.

Norman propone tre livelli di elaborazione cognitiva ed emotiva:

1. **Livello Viscerale**: Reazioni immediate, istintive, legate all'amigdala ("cervello rettiliano"). Riguarda l'apparenza, il feeling immediato.
2. **Livello Comportamentale**: Sede delle competenze apprese, azioni subconsce. Riguarda il piacere e l'efficacia dell'uso.
3. **Livello Riflessivo**: Pensiero conscio, riflessione, auto-immagine, ricordi. Riguarda il significato, la cultura, l'esperienza complessiva.

Un buon design considera tutti e tre i livelli. Ad esempio, il livello riflessivo influenza l'obiettivo, quello comportamentale l'intenzione e la specificazione, e quello viscerale l'esecuzione e la percezione.

I Golfi dell'Esecuzione e della Valutazione

Quando c'è una discrepanza tra il modello mentale dell'utente e il funzionamento reale dell'oggetto, si creano dei "golfi":

- Il **Golfo dell'Esecuzione**: L'utente sa cosa vuole fare, ma non sa come farlo con l'oggetto. *L'oggetto mi fa capire cosa posso fare?* È la distanza tra le intenzioni dell'utente e le azioni permesse dal sistema. Si supera con buoni significanti, affordance, mapping e vincoli.
- Il **Golfo della Valutazione**: L'utente ha agito, ma non capisce se l'azione ha avuto l'effetto desiderato. *L'oggetto mi fa capire cosa è successo?* È la distanza tra lo stato del sistema e la sua rappresentazione percepibile dall'utente. Si supera con un feedback chiaro e immediato.

7. Errori Umani: Mistakes e Slips

È fondamentale distinguere tra:

- **Mistake (Errore Concettuale o Sbaglio):** L'intenzione è sbagliata perché si basa su una comprensione errata del sistema (modello mentale fallace). L'utente formula un obiettivo o un piano d'azione errato.
- **Slip (Lapsus o Svista):** L'intenzione è corretta, ma l'esecuzione è errata a causa di disattenzione, automatismi, ecc. L'utente sa cosa fare e come, ma fallisce nell'esecuzione.

I **Mistakes** avvengono principalmente negli stadi superiori del modello a sette stadi (formulazione dell'obiettivo, intenzione, interpretazione, valutazione). Gli **Slips** avvengono negli stadi inferiori (specificazione dell'azione, esecuzione, percezione).

Cause di Mistakes:

- **Spiegazioni fallaci:** L'essere umano è una "macchina che spiega", spesso basandosi su analisi incomplete o concezioni antropomorfe ("il computer mi odia"). La "fisica aristotelica ingenua" (credenze di senso comune smentite dalla fisica) è un esempio.
- **Impotenza appresa (learned helplessness):** Tendenza a incolpare sé stessi per fallimenti, portando a una percezione di incapacità globale e a smettere di provare.
- **Impotenza insegnata (taught helplessness):** Cattive spiegazioni o sistemi mal progettati convincono l'utente di non essere "portato" per un certo compito (es. la matematica per alcuni).
- Questo può portare a **profezie auto-avverantesi**.

Tipi Comuni di Slips:

- **Lapsus da Cattura (Capture slips):** Un'azione frequente e automatizzata "cattura" una sequenza di azioni simile ma meno frequente (es. guidare verso l'ufficio nel weekend per abitudine).
- **Lapsus da Descrizione Errata (Description slips):** L'azione corretta viene eseguita sull'oggetto sbagliato perché la descrizione mentale dell'azione è simile per entrambi (es. mettere il latte nel cestino della spazzatura invece che nel frigo).
- **Lapsus di Memoria (Memory-lapse slips):** Si dimentica parte di un'azione, spesso a causa di interruzioni (es. dimenticare la carta nel bancomat).
- **Lapsus da Errore di Modo (Mode-error slips):** Si esegue l'azione corretta per una modalità diversa da quella attiva (es. premere un tasto che ha funzioni diverse in diverse modalità di un software).

Altre Cause di Errori: minimizzazione dello sforzo, razionalizzazione, attenzione selettiva, pressione sociale/economica.

8. Progettare Tenendo Conto degli Errori

Un buon design non ignora gli errori, ma li gestisce:

- **Visibilità e Feedback Chiari:** Rendere visibili le parti rilevanti (riduce il golfo dell'esecuzione) e fornire feedback immediato ed evidente (riduce il golfo della valutazione).
- **Gestione dell'Errore:**
 - Comprendere le cause degli errori e progettare per minimizzarli.
 - Rendere le azioni **reversibili** (es. comando "Annulla").
 - Rendere le azioni **irreversibili** molto difficili da compiere (es. chiedere conferme).
 - Aiutare la scoperta degli errori e la loro correzione.
 - Accettare che l'utente usi approcci per **tentativi ed errori**.
-
- **Funzioni Forzanti (Forcing Functions):** Vincoli fisici che prevengono errori.
 - **Interlock (Interblocco):** Una parte blocca l'operazione di un'altra (es. lo sportello del microonde che deve essere chiuso per avviare la cottura).
 - **Lock-in (Mantenimento):** Garantisce la funzionalità anche in caso di interruzione (es. salvataggio automatico, richiesta di conferma prima di chiudere un file non salvato).
 - **Lock-out (Esclusione/Blocco di sicurezza):** Rende un'operazione deliberatamente difficile per assicurarsi che l'utente voglia davvero attivarla (es. la sicura di un'arma, la doppia pressione di pulsanti per avviare macchinari pericolosi).

9. Estetica, Usabilità e Oggetti Deliberatamente Difficili

- Il **designer non è l'utente tipico**: ha conoscenze specializzate e un modello del sistema preciso. Il **cliente (committente)** potrebbe essere più interessato ai costi che all'usabilità.
- Attenzione alla **"featuritis"** (eccesso di funzionalità) e ai falsi ideali (privilegiare l'estetica o l'efficienza a scapito dell'usabilità). L'estetica è importante, ma oggetti belli e inutilizzabili sono frustranti. Oggetti usabili, invece, tendono ad essere percepiti come più belli.
- A volte, la **difficoltà è un requisito**: per azioni speciali o pericolose, è necessaria l'attenzione esplicita dell'utente (es. interruttori di sicurezza, comandi a due mani su macchinari industriali, alcuni videogiochi che basano la sfida sulla difficoltà).

Conclusioni

Progettare oggetti quotidiani efficaci richiede una profonda comprensione della psicologia umana. Dobbiamo:

- Capire **come funziona l'utente**: i suoi processi cognitivi, i suoi modelli mentali.
- Capire **quanto e come l'utente può sbagliare**: anticipare gli errori e progettare per prevenirli o mitigarli.
- Sapere **come l'utente impara e ricorda**: bilanciare la conoscenza nella testa e quella nel mondo.

Adottando questi principi, possiamo creare oggetti che non siano solo funzionali, ma anche piacevoli e capaci di arricchire la nostra esperienza quotidiana.

Riferimenti Bibliografici Chiave:

- Norman, D. (1988). *The Psychology of Everyday Things*. Basic Books. (Ed. it. *La caffettiera del masochista*, Giunti Editore).
- Norman, D. (2013). *The Design of Everyday Things: Revised and Expanded Edition*. Basic Books. (Ed. it. *La caffettiera del Masochista*, Giunti Editore, 2015 – fa riferimento alla nuova edizione).

06 - User Experience Design – Parte I

Introduzione alla Lezione

Benvenuti alla prima parte della nostra esplorazione dello User Experience Design (UX Design). Oggi ci concentreremo sui modelli di processo orientati all'utente e inizieremo ad approfondire uno dei framework più influenti in questo campo: "Gli Elementi della User Experience" di Jesse James Garrett. L'obiettivo è capire come strutturare il processo di design per creare prodotti che non siano solo funzionali, ma che offrano un'esperienza utente positiva e significativa.

1. Due Modelli di Processo Orientati all'Utente

Quando parliamo di progettazione centrata sull'utente, esistono diversi modelli che possono guidarci. Ne esamineremo brevemente due:

- **Un modello orientato ai task (compiti): ISO 9241-210 (2010)**
 - Questo è uno **standard internazionale ufficiale**, originariamente sviluppato nel Regno Unito.
 - L'obiettivo principale è il **design dell'usabilità**.

- Prevede cinque fasi principali: Studio di fattibilità, Definizione dei Requisiti Utente, Implementazione, Valutazione e Rilascio (Deploy).
- È un approccio strutturato e formalizzato, spesso utilizzato in contesti che richiedono certificazioni o aderenza a standard rigorosi.
-
- **Un modello orientato agli obiettivi (goal-oriented): Jesse James Garrett (2011)**
 - Proposto da un noto professionista statunitense, il cui modello si è diffuso ampiamente tramite il **passaparola** tra i designer.
 - L'obiettivo è lo **User Experience Design** nella sua accezione più ampia.
 - Si articola in **cinque piani** (Five Planes): Strategia (Strategy), Scopo/Obiettivi (Scope), Struttura (Structure), Scheletro (Skeleton) e Superficie (Surface). Questo sarà il modello su cui ci concentreremo maggiormente.
-

2. Gli Elementi della User Experience secondo Jesse James Garrett

Il modello di Jesse James Garrett, "The Elements of User Experience", è nato inizialmente come uno schema disegnato in modo semplice e diffuso tra i web designer. Successivamente, si è evoluto in una pagina web e infine in un libro (pubblicato nel 2006 e poi nel 2010), fornendo un modello concettuale e una serie di passaggi implementabili per gestire il processo di User Experience Design. È interessante notare che Jesse James Garrett è anche l'inventore del termine **AJAX** nel 2005, una tecnologia che ha rivoluzionato il modo in cui interagiamo con le applicazioni web. In questa lezione, estenderemo il modello di Garrett includendo altre idee e approcci compatibili.

Lo Schema di Garrett: I Cinque Piani

Lo schema di Garrett visualizza il processo di UX design come una serie di **cinque piani sovrapposti**, che si muovono **dall'astratto al concreto**:

1. **Piano della Strategia (Strategy):** Il livello più astratto. Qui si definiscono gli **obiettivi del prodotto** (Product Objectives) e i **bisogni degli utenti** (User Needs). Cosa vogliamo ottenere noi dal prodotto? E cosa vogliono ottenere gli utenti?
2. **Piano dello Scopo/Obiettivi (Scope):** Definisce le **specifiche funzionali** (per prodotti intesi come funzionalità, es. software) e i **requisiti di contenuto** (per prodotti intesi come informazione, es. siti web editoriali). Cosa farà il prodotto? Quali contenuti offrirà?
3. **Piano della Struttura (Structure):** Riguarda il **design dell'interazione** (Interaction Design) e l'**architettura dell'informazione** (Information Architecture). Come si organizzeranno le funzionalità e le informazioni? Come navigherà l'utente?

4. **Piano dello Scheletro (Skeleton):** Si concentra sul **design dell'interfaccia** (Interface Design), il **design della navigazione** (Navigation Design) e il **design dell'informazione** (Information Design, inteso come presentazione visuale). Come verranno disposti gli elementi? Come appariranno i percorsi di navigazione?
5. **Piano della Superficie (Surface):** Il livello più concreto, quello che l'utente vede. Include il **design sensoriale** (Sensory Design): l'aspetto grafico, il testo, le immagini, i link cliccabili, i form.

Caratteristiche del Modello di Garrett:

- È un **processo tendenzialmente lineare** che va dall'astratto al concreto.
- È nato principalmente per il **web**, ma i suoi principi sono applicabili a molti prodotti digitali.
- Mostra un **parallelismo** tra la progettazione di siti intesi come applicazioni (product as functionality) e siti intesi come contenitori di informazioni (product as information).
- Coinvolge **diversi ruoli professionali**: management, architetti dell'informazione, sviluppatori, grafici, commerciali.

Sovrapposizione dei Piani (Overlapping)

Sebbene i piani siano presentati in sequenza, nella pratica c'è una significativa **sovrapposizione**. Le decisioni prese nei piani inferiori (più astratti) influenzano quelle dei piani superiori (più concreti). Tuttavia, l'influenza può anche andare verso il basso: nuove tecnologie, nuove offerte della concorrenza, o una semplice richiesta del management di cambiare lo schema colori possono richiedere una riprogettazione che parte da un piano più concreto e si ripercuote su quelli più astratti.

In questi casi, è previsto che le fasi "precedenti" non siano completamente chiuse prima che le fasi "successive" siano già iniziate.

La Dualità del Web

Il web ha sempre avuto una **dualità fondamentale**:

- **Informazione:** È un medium per diffondere contenuto (testo, immagini, multimedia).
- **Applicazione:** È un'interfaccia distribuita per accedere a servizi remoti (su Internet o Intranet).

Questa dualità esiste a vari livelli: da sito a sito, da sezione a sezione di uno stesso sito, e da un'area all'altra di una singola pagina web. Il modello di Garrett tiene conto di questa dualità distinguendo tra "prodotto come funzionalità" e "prodotto come informazione".

Lo Schema Completo e la sua Relazione con l'ISO

Nello schema completo di Garrett, ogni livello corrisponde a una, due o tre attività specifiche da svolgere per completare la fase corrispondente. Molte di queste fasi trovano una corrispondenza diretta nel processo tradizionale dell'ISO 9241-210. Tuttavia, nel modello di Garrett, gli **obiettivi** (goals) assumono un'importanza molto più ampia e centrale.

3. Il Piano della Strategia (The Strategy Plane)

Il piano della strategia è la base di tutto il processo di UX design. Si articola in due componenti fondamentali:

- **Obiettivi del Prodotto (Product Objectives):** Cosa noi (l'azienda, il team di progetto) vogliamo ottenere da questo prodotto?
 - **Obiettivi di Business (Business Goals):** Come questo specifico prodotto si integra con gli obiettivi di business complessivi?
 - **Identità di Marca (Brand Identity):** Come questo prodotto promuove l'identità del brand e la rende più apprezzata?
 - **Metriche di Successo (Success Metrics):** Quando e come sapremo che il progetto è stato completato e ha avuto successo?
-
- **Bisogni e Obiettivi dell'Utente (User Needs and Goals):** Cosa vogliamo che gli utenti ottengano da questo prodotto?
 - Per comprendere i bisogni degli utenti, possiamo avvalerci di diversi framework e tecniche:
 - **Gerarchia dei Bisogni di Maslow:** Applicata ai bisogni dell'utente.
 - **Livelli di Elaborazione Cognitiva di Norman:** Per capire come gli utenti percepiscono e interagiscono.
 - **Tecniche di Goal-Oriented Design:** Tra cui segmentazione degli utenti, ricerca utente e creazione di personas.
 -
-

La Gerarchia dei Bisogni di Maslow e la User Experience

Abraham Maslow, nel 1954, propose una teoria delle motivazioni umane. Alla base di tutto c'è la sopravvivenza fisica, poi la sicurezza economica e sociale, il senso di appartenenza, la stima e infine l'autorealizzazione.

Ogni prodotto interessante e positivo che scegliamo si rivolge ad almeno uno di questi bisogni. Quindi, per il nostro prodotto, dobbiamo chiederci:

- Risponde ai bisogni fondamentali di sopravvivenza e sicurezza?
 - Protegge l'utente da minacce o altri problemi?
 - Fornisce supporto ai bisogni di un individuo o di un'unità sociale più ampia (famiglia, gruppo)?
 - Promuove un senso di appartenenza?
 - Promuove l'autostima o un senso di realizzazione personale?
- E se il prodotto non lo fa fisicamente, come lo fanno le sue funzioni o il modo in cui le fornisce?

I Livelli di Elaborazione Cognitiva di Norman

Donald Norman (già incontrato nella lezione precedente) elenca tre livelli di elaborazione cognitiva, ognuno con precisi impatti sul design:

- **Livello Viscerale:** Reazioni istintive agli aspetti visivi e sensoriali. Decisioni rapide su "buono", "cattivo", "pericoloso". Si progetta per la bellezza, la desiderabilità, indipendentemente da una valutazione conscia.
- **Livello Comportamentale:** Come eseguiamo compiti semplici e quotidiani. Si progettano comportamenti del prodotto che si adattano ai modelli mentali, alle aspettative e ai comportamenti dell'utente.
- **Livello Riflessivo:** Riflessioni cosce e memoria di esperienze passate che influenzano il comportamento. Si progettano relazioni a lungo termine in cui il prodotto ci cambia, ci migliora e ci fa imparare, evolvere, migliorare e avvicinare i nostri obiettivi.

Obiettivi (Goals) secondo Alan Cooper

Alan Cooper, un'altra figura di spicco nell'interaction design, sostiene che quelli che spesso chiamiamo "obiettivi utente" (es. archiviare documenti, impaginare un libro, verificare un'ipotesi di business) non sono i **veri obiettivi dell'utente**. Questi sono compiti o obiettivi di business.

I **veri obiettivi dell'utente** sono più personali ed emotivi:

- Non sembrare stupido.
- Non fare grossi errori.
- Svolgere una quantità ragionevole di lavoro.
- Divertirsi (o almeno non annoiarsi troppo).

Anche nelle attività lavorative, gli obiettivi sono personali, non legati al lavoro. Un software

progettato per obiettivi di business fallirà; un software progettato per obiettivi personali avrà successo e, di conseguenza, avrà successo anche per gli obiettivi di business.

Tipi di Obiettivi dell'Utente

Possiamo distinguere diverse categorie di obiettivi dell'utente, spesso pre-consoci e paralleli ai livelli di Norman:

- **Obiettivi di Esperienza (Experience goals):** Semplici, universali, privati, difficili da condividere. Determinati da come il nostro livello di elaborazione viscerale si applica a noi stessi.
 - Sentirsi intelligenti, stimolati.
 - Divertirsi.
 - Sentirsi "cool", alla moda, popolari.
 - Essere in controllo, interessati, non annoiati.
 - In breve: **come mi aspetto di sentirmi quando uso l'artefatto.** (Parallelo al livello viscerale)
-
- **Obiettivi Finali (End goals):** Le motivazioni dell'utente nello svolgere i compiti con l'artefatto. Giustificano il successo/insuccesso della nostra interazione.
 - Completare tutte le attività giornaliere entro l'orario d'ufficio.
 - Essere informato tempestivamente sui problemi.
 - Tenere traccia dei progressi delle mie attività.
 - Anche personali: trovare musica che mi piace, restare in contatto con gli amici, trovare un affare al prezzo più basso.
 - In breve: **come voglio usare il prodotto.** (Parallelo al livello comportamentale)
-
- **Obiettivi di Vita (Life goals):** L'insieme di aspirazioni e ambizioni globali che abbiamo. Indipendenti dagli artefatti, ma possono esserne influenzati. Orizzonte temporale lungo.
 - Vivere una vita piena e divertente.
 - Diventare X (trovare un lavoro, avanzamenti di carriera).
 - Essere un esperto in Y.
 - Avere tempo per attività extra-lavorative.
 - Essere attraente e apprezzato dai pari.
 - In breve: **come voglio pensare a me stesso.** (Parallelo al livello riflessivo)

Obiettivi dei Non-Utenti (Non-user goals)

Esistono anche obiettivi che non appartengono direttamente all'utente finale, ma che influenzano il prodotto:

- **Obiettivi del Cliente/Aquirente (Client/Buyer Goals):**
 - **Organizzativi:** Rendere i processi interni più efficaci, controllarli, automatizzarli, ridurre spese e risorse umane.
 - **Individuali (es. genitori per un prodotto per bambini):** Che sia educativo, migliori il rendimento scolastico, aiuti la socializzazione, favorisca lo sviluppo fisico/mentale del bambino, lo renda felice, con una spesa ragionevole.
- **Obiettivi dell'Organizzazione che Costruisce il Prodotto:**
 - **Commerciali:** Aumentare i profitti, la quota di mercato, battere la concorrenza, acquisire nuovi clienti, fidelizzare i vecchi, ampliare la linea di prodotti.
 - **Amministrazione Pubblica:** Fornire un servizio, educare il pubblico, ridurre l'uso di sportelli fisici e call center, ottimizzare l'uso di risorse finanziarie limitate.
 - **Tecnici:** Funzionare sui browser, garantire integrità dei dati, assicurare performance, mantenere compatibilità con versioni precedenti, garantire funzionalità e coerenza tra piattaforme.

Altre Questioni Connesse agli Obiettivi

- **Paura (Fear):** La paura del nuovo e dell'ignoto; paura basata su esperienze passate.
- **Empatia (Empathy):** L'abilità del designer di mettersi nei panni dell'utente.
- **Frustrazione (Frustration):** "Ciò che rende usabile un software è l'assenza di frustrazione nell'usarlo" (Rubin Chisnell).

4. Tecniche per il Design Orientato agli Obiettivi

Per definire e comprendere i bisogni e gli obiettivi degli utenti, utilizziamo diverse tecniche:

- **Segmentazione degli Utenti (User segmentation):**
 - Spesso il numero e la varietà degli utenti target rende difficile creare una caratterizzazione omogenea. La segmentazione permette di raggruppare gli utenti in sottogruppi omogenei rispetto a qualche caratteristica.
 - **Segmentazione Demografica:** Età, istruzione, stato civile, reddito, residenza, ecc. Può essere molto generica (es. uomini 18-59 anni) o molto specifica (es. donne 25-35 del Sud Italia, laureate e disoccupate, non sposate, reddito <20.000€/anno).

- **Segmentazione Psicologica:** Personalità, valori, attitudini, interessi, stile di vita, ecc. (es. motociclista single colletto blu).
- **Ricerca Utente (User research):**
 - **Metodi di Ricerca di Mercato (Market research methods):** Fonti indirette di dati (report esistenti), sondaggi (survey), focus group.
 - **Indagine Contestuale (Contextual inquiry):** Fonti dirette di dati ottenute interagendo con gli utenti: interviste, osservazioni dirette, presenza passiva.
 - **Analisi dei Compiti (Task analysis):** Identificare il contesto dei compiti in cui l'utente usa, vorrebbe usare o vorremmo che usasse il prodotto. Esaminare la sequenza di passaggi e interazioni.
- **Attività per la Ricerca Utente:**
 - **Prima della ricerca utente:** Questionari per i clienti, piani di progetto, "listening tours" (incontri informali), workshop e brainstorm.
 - **Durante la ricerca utente:** Elevator pitch (per presentare idee di design), artefatti dal futuro (per stimolare l'immaginazione), storyboard, modelli di Kano (per prioritizzare le feature).
 - **Questionari per i Clienti (Client Questionnaires):**
 - Fondamentali per allineare clienti e team di design.
 - Domande chiave: Chi fa parte del team (ruoli e limiti)? Quali sono gli obiettivi del progetto? Chi sono gli utenti target (quanti, come dividerli, motivazioni, bisogni, cosa usano ora)? *Errore comune: "I nostri utenti sono tutti". Non si progetta per tutti.* Perché spendere tempo e denaro? Qual è la **proposta di valore**? Perché dovrebbero scegliere questo prodotto? Quali sono le **misure di successo**? Fasi, date e milestone? Rischi?
 - **Piani di Progetto (Project Plan):** Definiscono attività, date di inizio, output, persone chiave, rischi.
 - **Attività Interattive:**
 - **Listening Tours:** Formato informale per apprendere priorità e obiettivi dei colleghi/clienti/utenti.

- **Workshops e Brainstorms:** Più formali, con più utenti e manager. Si decidono insieme obiettivi, aree problematiche, si discutono fattori di successo e metriche, si converge e si **prioritizza**.
 - L'obiettivo è raccogliere dati sotto forma di aspettative, desideri, punti dolenti, idee di design, "dos and don'ts", per dare forma ai requisiti utente.
- **Durante la Ricerca Utente - Esempi:**
 - **Elevator Pitch:** Descrivere l'idea di design in meno di un minuto. Utile un template "fill-in-the-blank" per brainstorm rapidi.
 - **Artefatti dal Futuro:** Un comunicato stampa fittizio, un post di blog, un articolo di giornale che descrive il prodotto finito. Aiuta a condividere la visione.
 - **Storyboard:** Utili per il brainstorming, visualizzano scenari d'uso.
 - **Modelli di Kano:** Aiutano a organizzare le feature e a prioritarle (Basic, Performance, Delighters).
- **Personas:**
 - Un approccio **drammatico e narrativo** al design. Gestire il design di un prodotto è come raccontare una storia, valutabile per interesse e credibilità.
 - Elementi chiave della narrazione: **Personaggi (personas)**, Ambientazione (contesto), Trama (casi d'uso).
 - **Perché le Personas?** Chiedere direttamente agli utenti non è sempre la soluzione migliore a causa di: differenze tra campione e classe di rappresentazione, relazione tra lo stato di "sofferenza" per un problema e l'abilità di risolverlo, differenze d'uso tra test e vita reale.
 - L'idea è **inventare utenti sintetici** che incarnino le caratteristiche che vogliamo supportare. Le personas sono quindi **archetipi astratti** di intenzioni, scopi e abitudini.
 - **Dati da Fornire per Ogni Persona:** Obiettivi (finali), Motivazioni (di esperienza e di vita), Comportamento (online e offline), Attitudine (approccio agli obiettivi e alla vita), Obiettivi del sistema (come il sistema aiuta a raggiungere gli obiettivi dell'utente).
 - **Cattive Personas (Bad Personas) - Da Evitare:**

- **L'utente elastico (John Smith / Mario Rossi):** Descrizione vaga che permette a tutti nel team di usarlo per supportare le proprie opinioni. Ogni volta che si discute una nuova scelta, l'utente diventa "elastico" per adattarsi alle preconcezioni del designer.
 - **L'utente autoreferenziale (Io):** Un'idealizzazione e astrazione del designer stesso (abitudini, idee, capacità, obiettivi).
 - **L'utente estremo (Il caso problematico):** Un anziano cieco, paralizzato, straniero, senza istruzione, senza competenze tecniche e di fretta. Sebbene esistano situazioni estreme, non si progetta per queste a scapito di una gamma di comportamenti più distintiva.
 - **L'utente medio (The average user):** Nessuno scrive storie su persone medie, che non vivono nulla di interessante. Una storia ha un protagonista interessante: tipico ma non medio, o meglio ancora un personaggio "borderline" le cui peculiarità lo distinguono.
- **Specificità nelle Personas:** Il livello di dettaglio della descrizione è cruciale *prima* della specificazione dei task.
- *Non scriviamo:* "L'utente sa già usare un software di videoscrittura".
 - *Scriviamo:* "Emily ha 53 anni, vive in campagna, ha un marito e due figli (17 e 13), diploma superiore, lavora per Global Airways da 12 anni ricevendo richieste da nuovi clienti. Lavora in una stanza 3x4 mt con due colleghi con cui va d'accordo, e usa MS Word 2011, di cui conosce il 15% delle funzionalità, perfettamente sufficienti per lei".
 - **Importante:** Dare nomi, caratteristiche di vita, storie plausibili, evitare che il designer si identifichi, magari dare un volto (foto).
- **Personas Sintetiche e Verosimili:** Gli utenti reali hanno peculiarità e idiosincrasie che un utente virtuale non ha (es. odiare i trackpad o il colore verde). Una persona può saltare dettagli irrilevanti. È più importante che l'utente sia disegnato con precisione, con dettagli che facilitino la costruzione di storie. Dobbiamo evitare le personas medie e concentrarci su una con caratteristiche specifiche e uniche.
- **Tipi di Personas:**
- **Protagonista (Protagonist):** Quello per cui il progetto è realizzato. C'è un protagonista per ogni ruolo, ed è l'utente la cui soddisfazione copre quella

della maggior parte degli altri. Non un caso estremo, non un caso medio, ma un **caso peculiare**.

- **Personas Secondarie (Secondary personas):** Più o meno soddisfatte dalle scelte fatte per il protagonista, ma con bisogni speciali e aggiuntivi che non richiedono una rilavorazione completa del progetto né riducono la soddisfazione del protagonista.
 - **Personas Aggiuntive (Additional personas):** Tutti gli altri utenti né primari né secondari, soddisfatti dal design realizzato.
 - **Personas Negative o Non-Utenti (Negative or non-user personas):** Utenti per i quali il prodotto NON è progettato. Ciò non significa che sia progettato per danneggiarli, ma le loro peculiarità non hanno impatto sul progetto.
- **Il "Cast":** Ogni progetto ha un cast di personas (tra 3 e 12). Si dovrebbe partire da una collezione ricca (anche 50-60), poi scartate progressivamente. Le personas sono tratte principalmente dai profili di segmentazione. La maggior parte sono utenti, alcuni non-utenti. Ogni cast contiene una persona principale, il protagonista, che deve essere soddisfatto al 100%. La scelta del protagonista è difficile ma fondamentale.

5. Considerare la Diversità e il Contesto dell'Utente

- **Curve di Apprendimento (Learning Curves):**

- Un sistema può focalizzarsi sulla **facilità di apprendimento** (aiutando i novizi, curva verde) o sull'**efficienza d'uso** (aiutando gli esperti, curva rossa).
- Oppure pianificare modalità per entrambi (es. menu ricchi e strumenti di personalizzazione), creando un "sistema usabile" che si posiziona sopra entrambe le curve.

- **Tipi di Utenti:**

- **Inesperto, principiante o novizio:** Enfasi sull'apprendimento. (Linee guida Macintosh anni '80).
- **Esperto:** Enfasi sull'efficienza. (Decalogo di Nielsen e Molich, 1989, sull'impazienza degli esperti).
- **Casuale o intermittente o sporadico:** Enfasi sulla memorizzabilità. (Lista di componenti di usabilità di Nielsen, 1994).

- **Intermedio Perpetuo (Perpetual Intermediate):** Enfasi sul giusto rapporto tra sforzo e competenza. (Alan Cooper, 1995: ognuno raggiunge un livello di competenza proporzionato allo sforzo che intende investire).
- **Competenze (Competence, competencies):**
 - L'utente non è necessariamente un informatico. Ci sono altre competenze rilevanti:
 - **Competenza di Dominio:** Specialmente se tecnico (finanza, medicina).
 - **Competenza Linguistica:** Legata al livello scolastico, importante per stranieri.
 - Anche le **abilità informatiche** sono complesse: saper usare un tablet ma non un PC; saper navigare in Internet ma non usare applicazioni desktop; saper usare un PC ma non gestirlo.
- **Forma Fisica (Physical fitness):**
 - Considerare estremi (persone cieche, sorde, quadriplegiche).
 - Ma anche malattie più frequenti e meno invalidanti: miopia, tempi di reazione elevati, difficoltà fisiche.
 - E difficoltà temporanee in individui normodotati: occhiali dimenticati, ambienti rumorosi, mani occupate.
- **Livelli di Attenzione (Attention levels):**
 - Inizialmente, solo contesti lavorativi: compito primario di un lavoro sedentario (segretaria), compito rilevante affiancato ad altri (capo al telefono), compito secondario mentre si fa qualcosa di più importante (tassista che controlla il tassametro guidando).
 - Con sistemi web e mobile, il lavoro diventa solo UNO dei contesti: attività molto coinvolgente (adolescente che gioca), passatempo (navigare su internet annoiato sul divano), attività esterna sotto la pioggia (cercare un indirizzo sullo smartphone).
- **Motivazioni (Motivations):**

- **Contesto lavorativo:**
 - Se lo strumento è scelto dall'organizzazione, devo usarlo, ma non deve piacermi (investo energia e tempo solo per garantire il mio lavoro).
 - Se ho scelto io lo strumento e la mia carriera dipende da esso (massimo investimento per raggiungere i miei obiettivi, ma niente di più).
- **Fuori dal contesto lavorativo:** Più difficile.
 - **Ipotesi Nulla (Null hypothesis) nella User Experience sul web:** spesso dimenticata.
 - **Legge di Jakob Nielsen sulla User Experience in Internet.**
 - **Il Paradosso delle Applicazioni Web Gratuite.**
- **L'Ipotesi Nulla (H0):** L'affermazione generale o predefinita che non c'è differenza tra due eventi che misuriamo, o che non sta accadendo nulla. Nei test A/B, si confrontano soluzioni A e B, ma non si considera se l'utente sceglierebbe di non avere né A né B. In situazioni di scelta discrezionale, l'ipotesi nulla è la **scelta di non usare alcuno strumento**. (Es. confrontare un editor con il disegno a mano; un'app di prenotazione con il presentarsi in ambulatorio).
- **Legge di J. Nielsen sulla User Experience in Internet:**
 - **"Gli utenti passano la maggior parte del loro tempo su ALTRI siti."**
 - Gli utenti preferiscono che il tuo sito funzioni come tutti gli altri siti che già conoscono.
 - La **coerenza esterna (External consistency)** è il singolo razionale di design più importante in qualsiasi campo maturo (come il web).
 - Consiglio: Non inventare. Ricicla. Se inventi, sii audace e inconfondibile. Non fare piccoli miglioramenti.
- **Il Paradosso delle Applicazioni Web Gratuite:**
 - Molte applicazioni web e mobile sono gratuite. Non richiedono investimento economico e pochissimo investimento emotivo.
 - Essere gratuiti o economici influisce negativamente sulle nostre aspettative: abbiamo poca o nessuna pazienza verso software gratuiti e siamo pronti ad abbandonarli alla prima difficoltà.

- Al contrario, un software costoso "impone" un certo sforzo per recuperare almeno l'investimento economico.
- **Nessun investimento (emotivo, economico) => nessuna pazienza.**

Riferimenti Bibliografici Chiave:

- Garrett, J. (2011). *The Elements of User Experience*. New Riders.
 - Harris, P. (2013). *Data Driven Design*. K&R publications.
 - Cooper, A., Reimann, R., Cronin, D. (2009). *About Face 3*. Wiley.
 - Cooper, A. (2004). *The Inmates Are Running the Asylum*. SAMS.
 - Tullis, T., Albert, B. (2013). *Measuring the User Experience*. Morgan Kaufmann.
 - Weinschenk, S. (2011). *100 Things Every Designer Needs to Know About People*. New Riders.
 - Usability Book of Knowledge: <http://www.usabilitybok.org/>
 - Evans, D. (2020). *Bottlenecks*. Apress.
-

07- User Experience Design – Parte II

Introduzione alla Lezione (Ripasso)

Bentornati alla seconda parte della nostra discussione sullo User Experience Design. Nella lezione precedente, abbiamo introdotto due modelli di processo orientati all'utente:

- Il modello **orientato ai task ISO 9241-210**, focalizzato sull'usabilità.
- Il modello **orientato agli obiettivi di Jesse James Garrett**, che mira a un approccio più olistico alla User Experience attraverso i suoi **cinque piani**: Strategia, Scopo, Struttura, Scheletro e Superficie.

Abbiamo anche iniziato a esplorare il modello di Garrett, sottolineando la sua evoluzione da uno schema informale a un framework concettuale completo per il processo di UX Design. Oggi, continueremo la nostra analisi del modello di Garrett, concentrandoci sul **Piano dello Scopo (Scope Plane)** e introducendo alcuni concetti chiave relativi alla definizione dei requisiti e alla narrazione nel design.

1. Il Piano dello Scopo (The Scope Plane)

Dopo aver definito la strategia (bisogni degli utenti e obiettivi del prodotto), il **Piano dello Scopo** si occupa di tradurre queste idee astratte in requisiti concreti. Le domande fondamentali a cui risponde questo piano sono:

- **Cosa stiamo facendo?** (Quali funzionalità e contenuti includeremo?)
- **Cosa NON stiamo facendo?** (Cosa escluderemo deliberatamente per mantenere il focus e la fattibilità?)

Questo piano si articola principalmente in:

- **Requisiti Funzionali (Functional Requirements) o Specifiche Funzionali (Functional Specifications):**
 - Questi termini a volte indicano due documenti distinti:
 - I **requisiti funzionali**, definiti all'inizio del progetto, descrivono **cosa il sistema** .
 - Le **specifiche funzionali**, definite alla fine del progetto, descrivono **cosa il sistema effettivamente fa**.
 -
 - In sostanza, definiscono le funzionalità che il prodotto offrirà per permettere agli utenti di raggiungere i loro obiettivi.
-
- **Requisiti di Contenuto (Content Requirements):**
 - Definiscono quali informazioni devono essere incluse nel contenuto che si sta sviluppando (testi, immagini, video, dati, ecc.).
-

Scrivere Requisiti e Specifiche Efficaci:

- **Non un progetto a sé:** La stesura delle specifiche non deve diventare un'impresa titanica.
- **Chiarezza e accuratezza, non volume:** Il dettaglio eccessivo non risolve i problemi; la chiarezza sì.
- **Focus sui punti critici:** Le specifiche non devono coprire ogni singolo aspetto, ma concentrarsi su quelli che potrebbero generare confusione nelle fasi successive.
- **Essere Positivi:**
 - *No:* "Il sistema non permetterà agli utenti di comprare un aquilone senza filo."
 - *Meglio:* "Il sistema indirizzerà l'utente alla pagina del filo per aquiloni se tenta di comprare un aquilone senza filo." (Suggerisce una soluzione, non solo un divieto).
-
- **Essere Specifici:**

- *No*: "I video più popolari saranno evidenziati."
- *Meglio*: "I video con il maggior numero di visualizzazioni nell'ultima settimana appariranno in cima alla lista." (Definisce chiaramente il criterio).
-
- **Evitare Linguaggio Soggettivo**:
 - *No*: "Il sito avrà uno stile hip e appariscente."
 - *Meglio*: "L'aspetto del sito sarà conforme al documento delle linee guida di branding aziendale." (Fa riferimento a standard oggettivi).

Gli Scenari (Scenarios)

Gli scenari sono **narrazioni dettagliate** che descrivono come un utente (spesso una delle nostre *personas*) raggiunge un obiettivo personale eseguendo uno o più compiti pianificati sul sistema. Sono uno strumento potente nel piano dello scopo.

Caratteristiche chiave degli scenari:

- **Relazione Obiettivo-Compito**: Collegano gli obiettivi dell'utente ai compiti che il sistema deve supportare.
- **Decomposizione dei Compiti**: Scompongono i compiti dell'utente in azioni (interne ed esterne).
- **Identificazione delle Operazioni**: Chiariscono le operazioni dell'utente e del sistema.
- **Narrazione**: Descrivono le azioni dell'utente, i suoi obiettivi e le sue motivazioni nell'usare il sistema.
- **Sistema come "Black Box"**: In questa fase, **NON specificano quali *feature* del sistema vengono usate**. Ci si concentra sul "cosa" e sul "perché", non ancora sul "come" implementativo.
- **Stime e Criteri di Successo**: Possono stabilire stime di tempo e criteri di successo per lo scenario e per ogni compito al suo interno.
- **Utilità Ampia**: Sono utili sia per le situazioni comuni sia per quelle critiche. Servono non solo a definire le feature tipiche, ma anche a **sottoporre a stress test** le caratteristiche peculiari del sistema.

Design Fiction e Prototipi Diegetici

Un concetto correlato agli scenari, particolarmente utile per esplorare possibilità future, è la **Design Fiction**.

- Bruce Sterling (2012) la definisce come "l'uso deliberato di **prototipi diegetici** per sospendere l'incredulità riguardo al cambiamento."

- Julian Bleecker (2009) la descrive come "un mix di fatti scientifici, design e fantascienza... La conclusione della finzione progettata sono **oggetti con storie**. Queste sono storie che speculano su nuove, diverse, distintive pratiche sociali che si assemblano attorno e attraverso questi oggetti."

Un **prototipo diegetico** è il prodotto che si sta progettando, immerso e funzionale alla narrazione della storia. Non è il *soggetto* della storia, ma uno *strumento* per il suo sviluppo. Viene descritto come se fosse un elemento normale nella vita dei personaggi, non un'innovazione recente e meravigliosa.

- Kirby (2010) sottolinea che "gli aspetti performativi dei prototipi sono particolarmente evidenti nei prototipi diegetici, perché la struttura narrativa di un film contestualizza le tecnologie all'interno della sfera sociale. Le narrazioni nel cinema popolare richiedono certezza dai loro dispositivi tecnologici per far avanzare le loro storie."

Elementi della Design Fiction (Poetics of Design Fiction):

- Scenari "What-if" (E se...).
- Regole base della finzione.
- Strumenti di design.
- Prototipi diegetici.
- **Scenari Pastiche:** Spesso si usano personaggi da film o libri o si creano scenari fortemente basati su personaggi noti (es. i protagonisti di Breaking Bad come base per esplorare scenari d'uso).

Fasi dell'Analisi dei Compiti (Task Analysis)

L'analisi dei compiti è un processo strutturato per comprendere in dettaglio cosa gli utenti fanno o devono fare:

1. **Identificare il compito da eseguire.**
2. **Scomporre i compiti in pochi sotto-compiti** in termini di obiettivi e sotto-obiettivi.
3. **Scrivere i diagrammi dei compiti e verificarne la completezza** (gestione dei sotto-casi, gestione degli errori, ecc.).
4. **Verificare una scomposizione omogenea** di tutti i dettagli di ogni sotto-compito.

Un Esempio Tradizionale di Scenario:

Help line per un sistema di ordinazione online

Una mattinata intensa con una lunga coda di chiamate all'ufficio assistenza clienti. Andrea lavora in questo ufficio solo da una settimana e riceve una telefonata dal Sig. Rossi.

Il Sig. Rossi non ha ricevuto della merce che ha ordinato tre settimane fa. Fornisce nome e indirizzo. Andrea recupera l'ordine e lo controlla.

La merce è stata consegnata al corriere una settimana fa, quindi Andrea fornisce al Sig. Rossi il codice di tracciamento del pacco per ulteriori richieste direttamente al corriere.

Mentre controlla i dati del Sig. Rossi, Andrea nota un errore nel codice postale, lo verifica con il Sig. Rossi e lo corregge immediatamente.

Dettagli dell'Esempio:

- Descrive **compiti**, non comandi dell'interfaccia.
- È piuttosto **specifico**.
- Descrive un **compito completo**.
- Descrive gli **utenti** (Andrea, neoassunta; Sig. Rossi, cliente).

Compiti, non Comandi:

- Nello scenario **non si fanno supposizioni** sui tipi di comandi da attivare o sulla struttura dell'interfaccia.
- Questa genericità permette di **confrontare equamente alternative di design diverse**.
- Se avessimo specificato "Andrea digita il nome nel campo di input", avremmo pre-descritto il modo "corretto" di compiere il task, impedendo l'esplorazione di alternative.

Specificità Estrema:

- Lo scenario non dice solo cosa l'utente deve fare con accuratezza, ma specifica esattamente **quali percorsi dell'interfaccia sono interessati**.
- In pratica, permette (forza) di specificare ogni tipo di dettaglio che potrebbe diventare rilevante nella valutazione delle alternative di design.
- Nell'esempio, emerge l'utilità di presentare informazioni interne (come il codice di tracciamento) all'help center. (Si pensi anche al caso dei nomi di file lunghi e come gestirli).

Compiti Completi:

- La descrizione del compito è molto specifica e descrive una **situazione completa**.
- Questo ci permette di valutare come i diversi aspetti dell'interfaccia lavoreranno insieme.
- La lista tradizionale dei requisiti dell'ingegneria del software è spesso solo un elenco di azioni individuali che il sistema deve eseguire. Non si discute solitamente come queste azioni individuali si compongono per realizzare compiti significativi e completi.

- È fondamentale **partire dalla gestione dell'input e dell'output** del compito complessivo.

Descrizione Esatta degli Utenti:

- Così come dettagliamo esattamente i compiti, allo stesso modo dovremmo **dettagliare gli utenti** (attraverso le personas). Questo è l'unico modo per "immedesimarci" nell'attitudine, nella psicologia e nelle esperienze dell'utente.
- Ad esempio, un sistema esperto medico rivolto a dottori (che non hanno tempo di imparare nuovi strumenti) aveva un'interfaccia identica ai moduli cartacei che i dottori già compilavano a mano. Questa decisione non sarebbe stata presa se l'interfaccia fosse stata per assistenti tecnici medici (abituati ai computer e costringibili a imparare un nuovo strumento).

2. Alcune Riflessioni sui Compiti e sull'Interazione

Andando oltre la semplice definizione dei requisiti, ci sono diversi concetti psicologici e di design che influenzano come gli utenti percepiscono e interagiscono con i compiti:

- **L'Importanza della Prima Impressione (First Impression):**
 - "Non si ha mai una seconda possibilità per fare una buona prima impressione."
 - La nostra prima impressione non è necessariamente ottenuta usando il sistema: possiamo vedere un amico usarlo, o uno sconosciuto sul treno.
 - Che tipo di sensazione mi trasmette? Che impatto ha su di me? È gratificante? Divertente? Interessante? Utile?
-
- **Impegno Proporzionato (Proportionate Commitment):**
 - Nessun utente dedicherà tutti i suoi sforzi per far funzionare un sistema.
 - La sua dedizione sarà **proporzionata all'utilità che si aspetta di ricevere** dal sistema.
 - Pertanto, strumenti software che richiedono un impegno maggiore dell'utilità attesa del risultato non saranno usati, indipendentemente da qualsiasi altra giustificazione.
-
- **Segnale-Routine-Ricompensa (Cue-Routine-Reward):**
 - Questo ciclo riduce i requisiti cognitivi delle persone che cercano di risolvere problemi normali e svolgere attività quotidiane.
 - Esempio: L'addestratore lancia un bastone (**segnale**), il cane corre a prenderlo e lo riporta (**routine**) ricevendo un biscotto o una carezza (**ricompensa**).
 - Il **segnale** è l'evento che innesca la reazione.
 - La **routine** è l'azione frequente connessa al segnale.
 - La **ricompensa** è il beneficio conseguente.

-
- Le attività che diventano routine sono le più semplici da eseguire e non richiedono sforzo. (Es. l'attività mentale dei topi che hanno imparato il labirinto è minore di quella dei topi che lo stanno ancora imparando).
-
- **Ciclo dell'Abitudine (Habit Loop):**
 - Il completamento di un'azione, o il moderato successo di una scelta, rinforzerà nella nostra mente tale scelta o azione più di altre potenzialmente altrettanto o più positive.
 - Dopo alcune ripetizioni, questa azione diventa un'**abitudine**, mai più messa in discussione, e questa scelta diventa automatica e immediata. Questo diventa un **ciclo dell'abitudine**.
 - Il ciclo dell'abitudine è il risultato del ricevere una ricompensa dopo una routine. Genera un'idea di proporzionalità nell'impegno speso nelle nostre attività.
 - Sono utili in generale perché semplificano la nostra vita quotidiana riducendo il numero di decisioni e riflessioni.
 - I cicli dell'abitudine sono **pericolosi** quando portano a comportamenti squilibrati (disordini alimentari, mancanza di attività fisica, cattive abitudini, ecc.).
-
- **Dissonanza Cognitiva e Impotenza Appresa (Cognitive Dissonance and Learned Helplessness):**
 - La dissonanza cognitiva è un conflitto psicologico dovuto a credenze incongrue e simultanee.
 - Se gli esseri umani non riescono a trovare una soluzione a un problema in un tempo ragionevole, si considerano soddisfatti di una soluzione subottimale, anche se sanno che, da qualche parte e in qualche modo, è possibile trovare una soluzione migliore.
 - L'insoddisfazione crea dissonanza cognitiva che abbassa la soglia di aspettativa (portando potenzialmente all'impotenza appresa).
-
- **Il Flusso (The Flow):**
 - Nelle barche a vela, c'è un momento magico in cui la velocità è sufficiente per sollevare la barca sulla sua stessa scia (scia), toccando appena la superficie dell'acqua e raggiungendo alte velocità. Accade all'improvviso ed è una sensazione meravigliosa.
 - Gli esseri umani hanno uno stato psicologico simile chiamato "**flow**", che si attiva improvvisamente quando ci concentriamo su un compito.

- Il "flow" è definito come un "coinvolgimento profondo e quasi meditativo" nel compito da svolgere, e spesso induce una "leggera sensazione di euforia" e una perdita del senso del tempo.
- In stato di flow, le persone sono molto produttive, specialmente in attività creative o di progettazione.
-
- **Flusso e Orchestrazione (Flow and Orchestration):**
 - Come per le barche a vela, anche per gli esseri umani il flow è uno stato magico e fragile. È necessario **evitare che la goffaggine dell'interazione lo interrompa**. Una volta usciti dal flow, è difficile e lento rientrarci.
 - Modi utili per evitare di uscire dal flow:
 - **Seguire i modelli mentali:** Uno strumento che organizza le sue procedure attorno al modello mentale dell'utente non disturba.
 - **Diretto, non discutere:** Un volante non discute; limita la scelta dell'utente a due direzioni, ma non avvia una conversazione sulla scelta ottimale della prossima direzione.
 - **Tenere gli strumenti a portata di mano:** Una barra degli strumenti permette agli utenti di tenere vicini gli strumenti di cui hanno bisogno e di sceglierli rapidamente e facilmente.
 - **Fornire feedback privo di modalità (modeless feedback):** Il modo più semplice per fornire un valore di feedback all'utente è mostrare una finestra modale, ma ciò richiede all'utente di chiuderla esplicitamente. L'informazione non modale è un modo molto migliore, che può essere mostrata all'utente interessato senza infastidire chi non lo è.
 -
 - Per ottenere una migliore interazione, l'**orchestrazione** delle varie parti dell'interfaccia in un unico insieme coerente ed efficace è il meccanismo fondamentale. L'obiettivo finale è l'**invisibilità dell'interfaccia**.
-
- **Gamification:**
 - Un altro approccio importante è lo sfruttamento del nostro interesse per la sfida del gioco.
 - La **gamification** è la disciplina che cerca di migliorare la nostra percezione trasformando attività di routine in giochi.
 - Può stimolare naturalmente un effetto di flow e permetterci di concentrare la nostra attenzione sul compito, ottenendo risultati migliori con poco sforzo. (Es. gli operatori degli scanner a raggi X negli aeroporti).

- **Paidia & Ludus:** Concetti che distinguono due forme di gioco:
 - **Paidia:** Attività non strutturate e spontanee (gioco libero dei bambini).
 - **Ludus:** Attività strutturate basate su regole (giochi da tavolo, sport).
 - La gamification si colloca più verso il Ludus, ma può incorporare elementi di Paidia.
-
- **Quadrante della Gamification:**
 - Un modo per visualizzare diversi approcci al gioco e al design ludico:
 - Asse Orizzontale: Parti (Parts) <-> Intero (Whole)
 - Asse Verticale: Giocare (Playing) <-> Gioco Strutturato (Gaming)
 - *Serious Games*: Intero + Gaming
 - *Gameful Design (Gamification)*: Parti + Gaming (applicare meccaniche di gioco a contesti non ludici)
 - *Toys (Giocattoli)*: Intero + Playing
 - *Playful Design*: Parti + Playing
 -
-
- **Componenti del Gioco (Framework MDA: Mechanics-Dynamics-Aesthetics):**
 - **Meccaniche (Mechanics):** I componenti atomici del gioco (regole, numero di giocatori, ruoli, sequenza di azioni consentite). Sono come gli ingranaggi individuali.
 - **Dinamiche (Dynamics):** Il comportamento del gioco mentre le meccaniche vengono usate correttamente (strategie, obiettivi condivisi/privati, eventi run-time). Esempi: conquistare l'Oceania a Risiko, collezionare paludi invece di foreste in Magic, scegliere bianco o nero negli scacchi.
 - **Estetiche (Aesthetics):** I modi in cui il gioco riesce a intrattenere i giocatori. Meccaniche e dinamiche cooperano per evocare sensazioni piacevoli secondo una o più emozioni:
 - *Sensazione*: Il giocatore sperimenta sentimenti non familiari.
 - *Fantasia*: Il giocatore osserva o crea un mondo immaginario.
 - *Narrativa*: Il giocatore osserva o crea una storia che lo guida.
 - *Sfida*: Il giocatore è spinto a padroneggiare un'abilità fisica o mentale.
 - *Compagnia (Fellowship)*: Il giocatore desidera far parte di una comunità.
 - *Scoperta*: Il giocatore vuole esplorare il mondo di gioco.
 - *Espressione*: Il giocatore esercita la propria creatività.

- **Motivazioni (Motivations):**

-

- Le motivazioni ci spingono a compiere azioni specifiche. Ignorarle significa fallire l'analisi.
- **Motivazioni Estrinseche:** Esterne all'individuo. Agiamo per ricevere una ricompensa o evitare una punizione (es. punteggi, badge, classifiche nella gamification).
- **Motivazioni Intrinseche:** Interne all'individuo. Agiamo perché proviamo uno stimolo e una gratificazione dall'atto stesso. Nella gamification, si considerano quattro categorie principali:
- **Motivazioni Intrinseche: Modello RAMP (Marczewski, basato sulla Self Determination Theory di Deci & Ryan):**
 - **Relatedness (Relazionalità):** Il desiderio di essere connessi ad altri esseri umani, vicini e remoti (classifiche, chat, forum).
 - **Autonomy (Autonomia):** La sensazione di essere liberi e responsabili delle nostre decisioni (organizzare attività in "quest" è meglio che forzare sequenze rigide).
 - **Mastery (Padronanza):** Il processo di diventare abili in qualcosa. Siamo orgogliosi di essere bravi in qualcosa di complesso. Il ritmo nell'aumento della difficoltà è importante.
 - **Purpose (Scopo):** L'identificazione del significato generale delle nostre attività, specialmente in relazione e interazione con gli altri.
-

-

- **Giochi e Giocatori (Games and Players):** Esistono diversi tipi di giocatori con motivazioni differenti (secondo la classificazione di Bartle, adattata):
 - **Players (Giocatori orientati alla ricompensa):** Giocano per sé stessi, per collezionare ricompense. (Motivati da Rewards)
 - **Socializers (Socializzatori):** Vogliono interagire con gli altri e creare connessioni. (Motivati da Relatedness/Social)
 - **Free Spirits (Spiriti Liberi/Esploratori):** Giocano per esplorare e creare. (Motivati da Autonomy)
 - **Achievers (Realizzatori):** Vogliono diventare bravi nelle meccaniche del gioco. (Motivati da Mastery)
 - **Philanthropists (Filantropi):** Vogliono aiutare gli altri e aumentare l'utilità del gioco per gli altri. Non giocano per sé stessi. (Motivati da Purpose)

- **Disruptors (Distruttori/Riformatori):** Giocano per il desiderio di cambiare il gioco e controllarlo. (Motivazione più complessa, spesso legata al potere o al desiderio di sfida delle regole).

08 - User Experience Design – Parte III

Benvenuti alla terza e ultima parte della nostra introduzione allo User Experience Design, focalizzata sul modello dei cinque piani di Jesse James Garrett. Nelle lezioni precedenti abbiamo coperto:

- **Il Piano della Strategia:** definizione dei bisogni utente e degli obiettivi del prodotto.
- **Il Piano dello Scopo:** traduzione della strategia in requisiti funzionali e di contenuto, e l'uso di scenari.

Oggi, saliremo ulteriormente nella piramide di Garrett, esplorando i piani più concreti:

- **Il Piano della Struttura (The Structure Plane)**
- **Il Piano dello Scheletro (The Skeleton Plane)**
- **Il Piano della Superficie (The Surface Plane)**

L'obiettivo è comprendere come le decisioni prese in questi livelli danno forma all'interazione finale e all'aspetto del prodotto.

1. Il Piano della Struttura (The Structure Plane)

Il Piano della Struttura si occupa di dare forma e organizzazione ai requisiti definiti nel piano dello scopo. Le domande chiave qui sono:

- **Come vengono usati i servizi/funzionalità?**
- **Come trovo le informazioni?**

Questo piano si divide in due aree principali:

- **Design dell'Interazione (Interaction Design):** Progettazione dei servizi interattivi.
 - **Modelli Concettuali:** Come l'utente si rappresenta il funzionamento del sistema.
 - **Uso delle Convenzioni:** Sfruttare pattern e comportamenti che gli utenti già conoscono.

- **Gestione degli Errori:** Come il sistema previene gli errori e aiuta l'utente a recuperarli.
 - **Il Dialogo:** L'interazione non è una singola domanda e una singola risposta. È come una danza: non è importante dove andiamo, ma come ci muoviamo. Avere un modello preciso del sito aiuta a dare coerenza.
 - **Importanza delle Convenzioni:** Usare modelli concettuali familiari all'utente. Questa è l'utilità residua delle metafore (es. carrelli della spesa).
 - **Gestione degli Errori:** Primo: evitare la possibilità di fare errori. Secondo: aiutare l'utente a capire che si è verificato un errore e aiutarlo a correggerlo.
- **Architettura dell'Informazione (Information Architecture - IA):** La strutturazione dei dati e dei contenuti.
 - **Approcci:** Top-down (dall'generale al particolare), bottom-up (dal particolare al generale).
 - **Strutture:** Alberi, reticoli (lattices), organiche, sequenziali.
 - **Catalogazione e Organizzazione.**
 - **Principi Organizzativi.**
 - **Metadati e Modelli di Metadati.**
 - (Nota: Dedicheremo una lezione intera all'Architettura dell'Informazione, quindi qui ne diamo solo un assaggio).

Strumenti comuni in questo piano sono **diagrammi e blueprint**.

- **Blueprint (Cianografie/Schemi):**
 - Sono disegni che definiscono:
 - I componenti per l'organizzazione del contenuto.
 - Come questi componenti sono connessi tra loro.
 - È spesso difficile rappresentare un sistema complesso con un solo blueprint, quindi è consigliabile fornire **molteplici prospettive** per l'architettura dell'informazione.
 - È anche consigliabile generare **versioni diverse** a seconda di chi vedrà il blueprint (programmatici, clienti, reparto marketing, ecc.).
 - Un blueprint è un modello dell'architettura informativa, uno schema in cui viene spiegato il **modello concettuale del sito**.

- **Non esiste una sintassi standardizzata per i blueprint.** È necessario fornire una legenda per le icone utilizzate e rimanere coerenti nei disegni.

- **Storyboard:**

- Una tecnica per illustrare attraverso immagini la struttura dei passaggi di esecuzione di un compito, mostrando lo stato dello schermo durante le fasi dell'azione.
- Mentre un wireframe si occupa della singola pagina, lo storyboard (possibilmente con meno dettaglio) mostra la **sequenza di pagine** e l'attivazione di widget interattivi (pulsanti, pop-up, ecc.) necessari per compiere le azioni.
- Può essere realizzato in modo molto convincente da illustratori professionisti, o tramite sequenze di schizzi grezzi da qualsiasi programmatore.

2. Il Piano dello Scheletro (The Skeleton Plane)

Il Piano dello Scheletro inizia a dare una forma più tangibile alla struttura, definendo il layout e la disposizione degli elementi. Si occupa di:

- **Design dell'Interfaccia (Interface Design):**

- **Widget:** Gli elementi interattivi come pulsanti, radio button, checkbox, menu, slider, ecc.

- **Design della Navigazione (Navigation Design):**

- Come l'utente si muove da un punto all'altro del sito.
- **Organizzare la navigazione:** Non fornire solo un elenco piatto di link senza gerarchia o ordine.
- **Comunicare la relazione tra gli elementi:** Non basta fornire una lista di link. Cosa hanno a che fare quei link l'uno con l'altro? Alcuni sono più importanti di altri? Quali sono le differenze rilevanti?
- **Comunicare la relazione tra i contenuti e la pagina corrente:** Cosa c'entra tutto questo con quello che sto guardando adesso?
- **Tipi di Navigazione:**
 - **Navigazione Globale:** Dà accesso alle parti principali del sito. Possibilmente non presente in ogni pagina (ma è una buona idea in generale).

- **Navigazione Locale:** Dà accesso a ciò che è "vicino" nel sito (genitori, fratelli, figli nella gerarchia).
 - **Navigazione Supplementare:** Dà accesso a contenuti disconnessi ma in qualche modo correlati alla pagina corrente (argomenti simili da sezioni diverse, "vedi anche").
 - **Navigazione Contestuale:** Accesso dal corpo della pagina, contenuti e link mescolati insieme.
 - **Navigazione di Cortesia (Courtesy Navigation):** Fornisce accesso a servizi a livello di sito sempre accessibili (FAQ, orari negozio, note legali).
- **Architettura dell'Informazione (Information Architecture) - Aspetti Visivi:**
 - **Wireframes:** Disegni delle parti più fondamentali della pagina. Un disegno per ogni "schermata" del sistema.

Wireframes:

- Sono disegni delle parti più fondamentali della pagina. È utile avere un wireframe per ogni schermata significativa del sistema.
- **Strumenti per Wireframes:**
 - Non sono strumenti molto complessi, paragonabili a un editor grafico, dove gli oggetti base non sono solo rettangoli e cerchi, ma anche finestre, pulsanti, aree di testo, ecc.
 - Esempi: Balsamiq (commerciale, creato a Bologna da un ex studente), Evolus Pencil (open source).
 - I wireframe possono anche **descrivere graficamente comportamenti** (es. come appare la schermata *prima* del login e *dopo* il login).

Prototipi Hi-fi vs. Lo-fi (Alta vs. Bassa Fedeltà)

- Una caratteristica condivisa degli strumenti di wireframing è che creano **mockup a bassa fedeltà (lo-fi)**:
 - Aspetto semplice, come uno schizzo veloce.
 - Non rifiniti, approssimativi, abbozzati.
 - Focalizzati sugli aspetti fondamentali, non sui dettagli.
 - Facili da produrre, modificare, discutere.
 - Lasciano ampio potenziale per la creazione di design ad alta fedeltà.

- I prototipi lo-fi hanno un **importante fattore psicologico**:
 - Clienti e management si sentono **più a loro agio nel suggerire idee, miglioramenti e critiche** grazie alla loro semplicità (un design troppo rifinito può intimidire).
 - Alla fine, la consegna non appare come una fase di "accetta/rifiuta", ma come una **collaborazione** tra designer e clienti per raggiungere un design accettabile.

Wabi Sabi (Estetica Giapponese)

- Un'importante caratteristica dell'estetica giapponese che descrive la bellezza:
 - ...delle cose **imperfette, incomplete, impermanenti**.
 - ...delle cose **umili e modeste**.
 - ...delle cose **non convenzionali**.
- Caratteristiche del Wabi Sabi: Irregolare, Intimo, Che suggerisce un processo naturale, Senza pretese, Terreno, Semplice.
- Questo concetto può ispirare l'approccio ai prototipi lo-fi, apprezzandone la natura "grezza" e collaborativa.

3. Il Piano della Superficie (The Surface Plane)

Questo è il livello più concreto, quello che l'utente finale vede e con cui interagisce direttamente. Riguarda il **Design Sensoriale (Sensory Design)**.

- **Estetica della Percezione**:
 - Sfruttare i principi della **Gestalt** (prossimità, somiglianza, chiusura, ecc.) per organizzare visivamente gli elementi.
- **Contrasto e Uniformità**:
 - Facilitare il **riconoscimento e l'orientamento** attraverso un uso efficace del contrasto e mantenendo un'uniformità visiva.
- **Consistenza Interna ed Esterna**:
 - Per facilitare l'apprendimento. La **consistenza interna** si riferisce all'uniformità all'interno del prodotto stesso. La **consistenza esterna** si riferisce all'uniformità con altri prodotti o convenzioni che l'utente già conosce.
- **Linee Guida (Guidelines)**:

- Provenienti dal sistema operativo (per app native), dall'azienda (brand guidelines) e dal progetto stesso.
- Fondamentali per la **coerenza nel tempo e tra diversi team di sviluppo**.

Acuità Visiva (Visual Acuity):

- L'occhio umano può leggere o percepire i colori effettivi solo in un'area di circa **6° attorno alla fovea** (centro focale della visione), che corrisponde a circa 5-6 caratteri.
- Il contenuto, ad esempio, NON sarà visibile se l'occhio del lettore è fisso a più di pochi centimetri di distanza, a meno che non si muova o cambi forma (attivando i gangli).

Scansione e Pattern di Sguardo a F (Scanning and the F shape gaze pattern):

- La scansione è **modale**: quando cerchiamo la dimensione del font ignoriamo i colori, quando cerchiamo immagini ignoriamo le parole, ecc.
- Gli occhi si muovono da sinistra a destra, dall'alto in basso, in una forma approssimativa di **F**:
 - L'area superiore è scansionata per la maggior parte della sua larghezza.
 - Poi lo sguardo scende con movimenti via via più brevi verso destra, formando una F più o meno chiara.
- L'area in **basso a destra** è nota come il **deserto attenzionale (attentional desert)**.
- Ovviamente, l'opposto accade nei sistemi di scrittura da destra a sinistra (es. Ebraico e Arabo).

Motivazione, Attenzione e Pattern di Sguardo:

- Quando cercano informazioni, i lettori scansioneranno la pagina secondo il pattern a F cercando qualcosa correlato all'informazione che cercano, seguendo una "scala attenzionale":
 1. Intestazioni (Headers)
 2. Icone
 3. Blocchi di parole
 4. Parole individuali
 5. Lettere individuali
- Il problema è che la scala attenzionale richiede sempre più energia, e quindi la **motivazione gioca un ruolo enorme**.

- I lettori smetteranno di percorrere la scala attenzionale non appena l'energia richiesta sarà maggiore della motivazione, e smetteranno di cercare l'informazione lì per passare a Google (o smettere del tutto di cercare).

Risposta di Orientamento e Sovraccarico (Orienting Response and Overwhelming):

- La percezione inaspettata dalla visione periferica è attivata dai **gangli**, che sono più frequenti lì.
- Dato il ruolo potenziale nell'informarci dei pericoli, l'analisi corretta dell'informazione periferica deve essere fatta con grande priorità in quella che è chiamata "**risposta di orientamento**".
- Questa è una risposta automatica molto forte che si **interrompe quando il cervello è sovraccaricato** da segnali da ogni direzione.
- Pertanto, la raccolta e l'integrazione delle **notifiche** è diventata un problema importante nei sistemi operativi recenti. (I "pallini rossi" delle notifiche – *red meatballs* – sono usati per questa integrazione).

Risposta di Orientamento e Abituazione (Orienting Response and Habituation):

- Per essere utile, una notifica dovrebbe essere **inaspettata, transitoria e infrequente**.
- La percezione costante di avvisi di notifica interrompe la risposta di orientamento e attiva una forma di **abituazione**, in cui ignoriamo felicemente percezioni che normalmente attiverrebbero una risposta.
- Più attiviamo una risposta di orientamento con suoni, colori e movimenti, più il collo di bottiglia attenzionale si restringerà e gli utenti la ignoreranno.

Gestalt e "Avarizia" Cognitiva (Gestalt and Avarice):

- Il raggruppamento di oggetti secondo dimensione, posizione, forma, ecc., permesso dai principi della Gestalt, sono meccanismi che il nostro cervello impiega per **ridurre la fatica** nell'analizzare e dare senso alla percezione.
- Il cervello cerca sempre modi per **ridurre lo sforzo mentale** necessario per arrivare a una nuova decisione, in una chiara forma di "avarizia" di energia.
- Questo a volte ha effetti inaspettati:
 - Senza raggruppamento Gestalt, dobbiamo spendere energia per scansionare, comprendere la percezione e persino impegnare la nostra mente conscia.

- Ad esempio, file e file di comandi identici (come nei vecchi telecomandi) sono difficili da scansionare e mappare a funzioni significative.
 - Cambiando forma e raggruppandoli in modo appropriato (come nei telecomandi più recenti), la mente conscia non è più necessaria.
-

09 - Architettura Dell'informazione (Information Architecture)

Introduzione alla Lezione

Benvenuti a questa lezione dedicata all'Architettura dell'Informazione (AI). Oggi esploreremo cosa significa strutturare e organizzare l'informazione in ambienti digitali per renderla accessibile, comprensibile e utile. L'AI è una disciplina fondamentale per la User Experience, poiché un'informazione ben architettata è la base per un'interazione efficace e soddisfacente.

1. Cos'è l'Informazione?

Prima di addentrarci nell'architettura, definiamo cosa intendiamo per "informazione". L'**American Library Association** definisce la **cultura dell'informazione** come:

L'insieme di competenze che ci permettono di riconoscere il bisogno di informazione, di identificare l'informazione appropriata, di trovarla, di valutarla e di sfruttarla in relazione a una data situazione, dalla prospettiva di risolvere un problema.

L'informazione, quindi, non è un dato grezzo, ma qualcosa che acquisisce significato e utilità in un contesto specifico.

2. Cos'è l'Architettura dell'Informazione?

L'Architettura dell'Informazione (AI) può essere definita in diversi modi, tutti convergenti sull'idea di dare forma e struttura all'informazione:

- La **progettazione strutturale di ambienti digitali** per lo scambio di informazioni.
- La **struttura logica, organizzativa e semantica** dell'informazione, del contenuto e delle funzioni di un sistema.
- La **combinazione di sistemi di organizzazione, etichettatura (labeling) e navigazione**.
- L'**arte e la scienza di dare forma all'informazione** per supportare l'usabilità e la reperibilità (discovery).

- Un modo per **connettere utenti e contenuti**, portando i principi del design e dell'architettura negli ambienti digitali.

Scopo dell'Architettura dell'Informazione

L'AI ha tra i suoi obiettivi principali che:

- Il **contenuto sia gestito ragionevolmente, organizzato propriamente, catalogato e filtrato.**
- Sia possibile **recuperare l'informazione efficacemente**: l'utente deve potersi concentrare sui propri compiti e obiettivi, non sulla struttura del sito o del contenuto. L'AI deve essere "invisibile" e intuitiva.

Architettura dell'Informazione e Design dell'Informazione

Entrambe le discipline si occupano della presentazione dell'informazione, ma con focus leggermente diversi:

- **Design dell'Informazione (Information Design)**: Come l'informazione *deve essere progettata* per essere compresa (es. visualizzazione dei dati, chiarezza del linguaggio). Si concentra sull'efficacia comunicativa del singolo elemento informativo.
- **Architettura dell'Informazione (Information Architecture)**: Come gli elementi informativi sono *correlati tra loro* e organizzati in un sistema più ampio. Si concentra sulla struttura e sulla navigabilità dell'intero corpus informativo.

Approfondimento sull'Information Design

L'Information Design è definito come l'arte e la scienza della preparazione dell'informazione affinché possa essere usata dagli esseri umani in modo efficiente ed efficace (Horn 1999).

I suoi obiettivi primari sono:

- Lo sviluppo di documenti che siano **comprensibili, ricercabili rapidamente e accuratamente, e facilmente traducibili in azioni.**
- Progettare **interazioni con strumenti che siano il più facili e piacevoli possibile.**
- Aiutare le persone a **orientarsi**:
 - In spazi tridimensionali, specialmente urbani.
 - In spazi virtuali.
 - In spazi ibridi (interfacce tangibili e naturali).

Il primo passo nella trasformazione dei dati è lavorare con la loro organizzazione. Il modo in cui organizziamo le cose riflette e influenza il modo in cui le percepiamo. Al cuore di tutto c'è la consapevolezza che **il dato in sé è fondamentalmente inutile o neutro**. È nella sua preparazione per la comunicazione che il dato acquisisce significato e valore, diventando **informazione**.

L'informazione non è la conclusione del continuum chiamato comprensione. L'**informazione deve essere trasformata in conoscenza**, che, a sua volta, viene valutata e interpretata insieme al resto della conoscenza e diventa **saggezza**.

Dal Dato all'Informazione, alla Conoscenza, alla Saggezza

- **Dato (Datum):**

- Numeri, simboli, misure, parole.
- Non implica né contiene significato.
- È un fatto quantificabile che risulta da un'osservazione diretta.
- Può esistere in qualsiasi forma, più o meno usabile.
- *Esempi:* stringhe ("UUXD"), date (17 Febbraio 2025), misure (2.48 m, 23 kg).

- **Informazione (Information):**

- Si colloca tra dato e conoscenza.
- Rappresenta un dato a cui può essere associato un significato.
- Esiste all'interno di un contesto.
- Fornisce uno scopo al dato.
- I messaggi tra persone sono comprensibili se inseriti nel contesto in cui sono avvenuti.
- *Esempi:* "UUXD" è il nome di un corso, 17 Febbraio 2025 è l'inizio del semestre, 2.48 m è l'altezza del ponte XY, 23 kg è il peso del pacco con codice XYZ.

- **Conoscenza (Knowledge):**

- Ciò che è dentro la testa delle persone.
- È consapevolezza e comprensione di fatti o informazioni ottenute attraverso l'esperienza o l'apprendimento.
- È l'autoconsapevolezza del possesso di informazioni interconnesse e sfruttabili.
- L'informazione diventa conoscenza quando ha un valore maggiore considerata nel suo insieme rispetto a quanto avrebbe presa separatamente.

- *Esempi:* "Per superare l'esame UUXD è necessario studiare il materiale", "Dovrò iniziare a frequentare il corso dal 17 Febbraio 2025", "I veicoli devono essere più bassi di 2.48 metri per passare sotto il ponte XY", "23 kg è sopra il limite di peso per i pacchi a prezzo normale".
- **Saggezza (Wisdom):**
 - L'insieme delle abilità necessarie per riconoscere il bisogno di informazione e per localizzare, valutare, applicare e creare informazione in un dato contesto culturale e sociale.
 - Il termine "padronanza dell'informazione" (mastery of information) è a volte preferito a "saggezza", che può sembrare più restrittivo.
 - *Esempi:* "Superare l'esame UUXD è necessario per laurearsi", "Devo trovare una stanza a Bologna qualche giorno prima del 17 Febbraio 2025", "Dato che il mio veicolo è più alto di 2.48 metri, devo trovare un percorso che non passi sotto il ponte XY", "Devo capire se è più economico dividere il pacco da 23kg in due più piccoli sotto il limite di peso, o pagare l'aumento di prezzo per i pacchi pesanti".

3. Il Modello Cynefin

Per affrontare i problemi di informazione e conoscenza, è utile il **Modello Cynefin** (pronunciato ku-NEV-in) di Dave Snowden (IBM, 1999-2014). Ogni comprensione di un problema deve essere collocata in uno dei cinque domini del modello:

- **L'Ovvio (The Obvious/Simple):** La relazione causa-effetto è ben compresa. L'approccio giusto è **Percepire (Sense)**, **Categorizzare (Categorize)** e **Rispondere (Respond)**. *Pratiche Migliori (Best Practices)*.
- **Il Complicato (The Complicated):** La relazione causa-effetto non è ovvia, ma esistono tecniche (spesso basate su analisi e applicazioni precedenti) che permettono un approccio **Percepire (Sense)**, **Analizzare (Analyse)** e **Rispondere (Respond)**. *Buone Pratiche (Good Practices)*.
- **Il Complesso (The Complex):** La relazione causa-effetto può essere scoperta solo retrospettivamente. L'approccio corretto è **Sondare (Probe)**, **Percepire (Sense)** e **Rispondere (Respond)**. *Pratiche Emergenti (Emergent Practices)*.
- **Il Caotico (The Chaotic):** La relazione tra causa ed effetto non esiste (o non è percepibile). L'approccio giusto è **Agire (Act)**, **Percepire (Sense)** e **Rispondere (Respond)**. *Pratiche Nuove (Novel Practices)*.

- **Il Disordine (The Disorder):** Non è chiaro se esista una relazione causa-effetto. Questo è lo spazio centrale di incertezza.

L'AI spesso si muove tra i domini Complicato e Complesso.

Le Sette Regole della Conoscenza (Dave Snowden)

1. La conoscenza può essere fornita solo spontaneamente, mai forzata.
2. Conosciamo le cose che sappiamo solo quando ne abbiamo bisogno.
3. Di fronte a un bisogno reale, nessuno "trattiene" la conoscenza.
4. La conoscenza è sempre frammentata.
5. La tolleranza agli errori supporta l'apprendimento meglio dei successi.
6. Il modo in cui conosciamo le cose non è il modo in cui diciamo agli altri che le conosciamo.
7. Sappiamo più di quanto diciamo ad alta voce, e diciamo ad alta voce più di quanto scriviamo.

4. Organizzazione dell'Informazione

Nathan Shedroff suggerisce che esistono solo **7 modi principali per organizzare l'informazione**:

1. **Alfabetico (Alphabets):** Indice di un libro, elenco telefonico, enciclopedia.
2. **Mappe (Maps):** Indicazioni per i bagni o le uscite di emergenza, piante, diagrammi, mappe della metropolitana.
3. **Lineare (Linear):** Storia, orari dei treni, ricette di cucina, sviluppo di un progetto.
4. **Continuum:** Voti scolastici, scale (durezza delle pietre, devastazione dei terremoti, valore del ristorante).
5. **Numerico (Numerical):** ISBN, classificazione Dewey, indirizzi IP. Usa numeri per indicare classi e sottoclassi in modo parzialmente arbitrario.
6. **Categorie (Categories):** La classificazione e la nomenclatura sono attività comuni degli esseri umani e controllano direttamente la percezione dell'informazione.
7. **Casuale (Random - nessuna organizzazione esplicita):** Utile in circostanze in cui l'organizzazione è l'informazione stessa (es. un solitario in cui le carte devono essere ordinate).

Eccellenza Grafica (Edward Tufte)

L'eccellenza grafica consiste nel comunicare idee complesse in modo chiaro, accurato ed efficiente. Le visualizzazioni grafiche di dati statistici dovrebbero quindi:

- Indurre gli osservatori a **concentrarsi sulla sostanza** piuttosto che sulla metodologia, sul design grafico, sulla tecnologia usata, ecc.
- **Mostrare i dati.**
- Evitare di distorcere ciò che i dati devono comunicare.
- Presentare molti numeri in poco spazio.
- Creare insiemi coerenti di numeri molto grandi.
- Incoraggiare l'occhio a confrontare diversi pezzi di dati.
- Mostrare i dati con diversi gradi di profondità.
- Essere al servizio di **uno scopo chiaro**: descrizione, esplorazione o decorazione.
- Essere integrata con la descrizione verbale e statistica di un set di dati.

5. Parole Chiave dell'Architettura dell'Informazione

- **Strutturare (Structuring)**
- **Organizzare (Organizing)**
- **Classificare (Classifying)**
- **Rendere Reperibile (Make findable)**
- **Rendere Gestibile (Make manageable)**

Strutturare, Organizzare, Classificare

- **Strutturare:** Determinare il livello di granularità dei dati presenti nel contenuto e decidere come sono correlati tra loro.
 - *Esempio rivista:* parole in frasi, in sezioni, in articoli, in numeri, in annate.
 - *Esempio dataset temporale:* dati puntuali in somme orarie, conteggi o medie giornaliere, settimanali, ecc.
-
- **Classificare:** Definire le categorie e la serie di link che le connettono.
- **Organizzare:** Raggruppare questi componenti informativi in categorie distinte e specifiche.

Reperibilità e Gestibilità (Findability and Manageability)

- **Reperibilità (Findability):** Permettere all'utente di accedere al contenuto e trovare l'informazione a cui è interessato, sia tramite navigazione (browsing) sia tramite funzionalità di ricerca.
- **Gestibilità (Manageability):** Bilanciare i bisogni dell'utente con gli obiettivi di business. Una gestione efficiente dei contenuti, policy e procedure sono essenziali.

Diverse Architetture Informative: Libreria vs. Negozi di Libri

- **Negozi di Libri (Bookstore):**

- Può organizzare i libri in modo "causale" (in realtà guidato da logiche commerciali e di marketing), suggerendo un'esperienza esplorativa.
- Ma se cerchi qualcosa di specifico, la ricerca può essere difficile.

-

- **Libreria/Biblioteca (Library):**

- Sistemi complessi e professionisti lavorano insieme per selezionare, valutare, classificare, descrivere, strutturare e organizzare il contenuto.
- Gli utenti hanno molti modi per trovare ciò che cercano. Molto più difficile per il suggerimento o la scoperta casuale (**serendipità**).

-

Serendipità (Serendipity)

- Termine coniato dallo storico Walpole nel 1754 per riferirsi ai protagonisti di una favola persiana, i tre principi di Serendip (antico nome dello Sri Lanka), che "trovavano sempre, per caso o saggezza, cose che non stavano cercando".
- Si riferisce a tutti i risultati utili, piacevoli o positivi a cui siamo attratti per caso e senza piani, mentre siamo impegnati a fare o cercare qualcos'altro.
- Incoraggiare la serendipità significa costruire sistemi che forniscano più del risultato specificamente cercato dall'utente, creando un contesto per i risultati, o nel percorso per raggiungerli, per facilitare la scoperta di cose inaspettate, curiose e utili.

6. Ecologia dell'Informazione (Information Ecology)

Il design dell'architettura dell'informazione deve tenere conto della stretta relazione tra **contesto**, **contenuto e utenti**, evidenziata da Davenport e Prusak con la metafora dell'Ecologia dell'Informazione.

- La **tecnologia** dovrebbe essere il quarto elemento? Forse sì, ma spesso l'aspetto tecnologico riceve troppa attenzione (Rosenfeld, Morville 2006).
- **Ecologia dell'Informazione: il Contesto**
 - **Contesti organizzativi e sociali:** Siti di ONG, e-government, social network.
 - **Contesto di business specifico:** Vendere libri è diverso da vendere viaggi, biglietti per spettacoli, giocattoli.

- **Missione, Obiettivi, Processi, Procedure, Cultura aziendale.**
- Di conseguenza, il vocabolario e la struttura di un sito sono condizionati da questi fattori; sono infatti una parte importante del dialogo tra il business del sito e i suoi utenti.
- Come abbiamo visto in precedenza, l'analisi richiede: identificazione degli utenti, dei loro compiti e obiettivi, dei vincoli tecnici e culturali.
- Il motto dell'architettura dell'informazione è che **ogni situazione è unica.**
- **Ecologia dell'Informazione: il Contenuto**
 - Fattori che influenzano il contenuto:
 - **Controllo:** Centralizzato o distribuito nei dipartimenti? Usiamo anche contenuti da fornitori esterni?
 - **Formato:** Testo, immagine, audio, video, ecc.
 - **Struttura:** Il nostro contenuto è completo o in continua crescita? Qual è il livello di granularità? Poche note da 100 parole o un manuale da 1000 pagine?
 - **Metadati:** Qual è lo scopo dei metadati che descrivono il contenuto del sito? Il contenuto è descritto manualmente o automaticamente? Gli utenti possono creare i propri modelli (es. creare e usare tag)?
 - **Volume:** Di quanti documenti stiamo parlando?
 - **Dinamicità:** Come e in che misura ci si aspetta che il sito cambi in futuro?
- **Ecologia dell'Informazione: gli Utenti**
 - Le differenze nelle preferenze e nei comportamenti dei clienti nel mondo fisico si riflettono in diversi bisogni informativi e in comportamenti di ricerca differenti.
 - Quindi, tutto ciò che sai sugli utenti è utile per determinare i loro bisogni/obiettivi.
 - **Quattro tipi di bisogni informativi:**
 - Coloro che cercano qualcosa di specifico (es. quanti abitanti ha Bologna?).

- Coloro che esplorano il sistema per trovare risposte (es. consultare il sito delle Relazioni Internazionali dell'Università di Bologna per decidere dove andare in Erasmus).
- Coloro che cercano di sapere tutto su un argomento (es. fare ricerca per una tesi).
- Coloro che cercano e poi riutilizzano (es. bookmarking, Del.icio.us).

7. Approcci all'Architettura dell'Informazione

- **Design Top-Down:**

- Nel modello top-down, una struttura generale del sistema viene formulata senza entrare nei dettagli di nessuna delle sue parti.
- Ogni parte del sistema viene progressivamente raffinata aggiungendo più dettagli dalle fasi di progettazione precedenti.
- Questo raffinamento continua finché la specificazione completa non è sufficientemente dettagliata per validare il modello.
- Il top-down parte dagli **obiettivi** e da essi indica la strategia appropriata per raggiungerli.

- **Design Bottom-Up:**

- In contrasto con il modello top-down, c'è il design bottom-up in cui le parti individuali del sistema sono caratterizzate in dettaglio.
- Queste parti vengono poi unite per formare componenti più grandi, che vengono a loro volta interconnessi per formare un sistema completo.

8. Componenti dell'Architettura dell'Informazione

L'AI si articola in diversi componenti chiave:

- **Organizzazione:**

- Per Soggetto.
- Cronologica.

- **Etichettatura (Labeling):**

- Modalità di rappresentazione dell'informazione (i nomi dati a link, sezioni, categorie).
- **Navigazione:**
 - Navigazione a livello di pagina (menu principali, barre laterali).
 - Navigazione all'interno del contenuto (link nel testo, ancore).
- **Ricerca (Search):**
 - Come cercare l'informazione.
 - Query vs. navigazione.

Un Vocabolario di Strutture Visive (Morville e Rosenfeld)

Morville e Rosenfeld propongono un sistema di progettazione basato su:

1. **Aiuti alla Navigazione (Browsing aids)**
2. **Aiuti alla Ricerca (Search aids)**
3. **Contenuti e Compiti (Contents and Tasks)**
4. **Componenti Invisibili (Invisible components)**

- **1. Aiuti alla Navigazione (Browsing Aids):**
 - Presentano all'utente un insieme di strumenti che lo aiutano a navigare nel sito. Permettono di trovare informazioni tramite navigazione (menu, link) piuttosto che query.
 - Scopo: incoraggiare l'esplorazione del contenuto, l'orientamento nel sito, la comprensione dello scopo e dell'organizzazione del sito, la serendipità.
 - **Includono:**
 - Sistemi di organizzazione dei contenuti (per storia, soggetto, compito, pubblico).
 - Gerarchie e raggruppamenti di elementi sulla pagina.

- Sistema di navigazione globale del sito.
- Sistema di navigazione locale (dove sono e cosa posso fare con una porzione del sito).
- Sistema di navigazione contestuale (all'interno del testo, per connettere contenuti specifici).
- Sistemi di orientamento (palette di colori, "dove siamo", breadcrumb).
- Mappa del sito (inclusa tabella dei contenuti).
- Indice del sito (in ordine alfabetico).
- Guida del sito (testo libero).
- Tag cloud.
- Wizard (sequenze passo-passo per compiti specifici).

- **2. Aiuti alla Ricerca (Search Aids):**

- Componenti che permettono all'utente di interrogare il sistema e ottenere un insieme di risultati. Una controparte dinamica e automatica degli aiuti alla navigazione.
- **Interfaccia di Ricerca:** Fornisce modi per inserire una query e visualizzare i risultati. Normalmente offre l'opzione di configurare la propria ricerca (ricerca avanzata).
- **Linguaggio di Query:** Operatori booleani (AND, OR, NOT) o modalità per specificare quale campo di interesse cercare (es. AUTORE = "Norman").
- **Costruttori di Query (Query Builders):** Modi per aumentare le prestazioni della ricerca (ortografia esatta, uso di sinonimi per suggerire ricerche alternative, stemming).
- **Presentazione dei Risultati:** Lista, griglia, carosello, ecc. Cliccabili, comparabili. Navigabilità tra risultato e dettaglio.
- **Problemi nelle query:**
 - Come gestire una query sintatticamente scorretta? (Campo specifico o campo di ricerca singolo).

- Come gestisco una query che NON restituisce risultati? (Messaggio di errore, suggerimento di query alternative, migliore approssimazione – es. ricerche di geolocalizzazione).
- Come gestire una query che restituisce UN SOLO risultato? (Mostrare il risultato in una lista altrimenti vuota, inviare l'utente direttamente alla vista del record individuale).
- Come gestire una query che restituisce TROPPI risultati? (Paginazione dei risultati / scroll infinito, messaggio di errore con ulteriore filtraggio, cut-off arbitrario).
- Differenza tra ricerca e filtro.

- **3. Contenuti e Compiti (Content and Tasks):**

- Sono la destinazione finale degli utenti (e non stanno navigando e cercando per l'utente).
- Usiamo elementi come:
 - **Intestazioni (Heading):** Etichette per il contenuto seguente.
 - **I link inclusi nel testo:** le etichette rappresentano il contenuto a cui si riferiscono.
 - **Tag inclusi nel testo** (es. un ingrediente in una ricetta da cui si può iniziare a cercare altre ricette con lo stesso ingrediente).
 - **Blocchi di testo (Text chunk):** Unità di contenuto logiche; possono variare per granularità ed essere annidate.
 - **Liste (Lists):** Blocchi o gruppi di link che puntano a un blocco. Assumono particolare importanza perché sono raggruppati insieme.
 - **Aiuti sulla posizione all'interno di una sequenza** (es. questo è il passo 3 di 8).
 - **Identificatori (Identifiers):** Suggestiscono dove si trova l'utente nel sistema (es. breadcrumb, colori di sezione, loghi).

- **4. Componenti Invisibili (Invisible Components):**

- Sono strutture informative non user-friendly ma che possono essere utili ad altri componenti visibili.
- Esempi:
 - **Thesaurus:** Vocabolari controllati (spesso riferiti a un dominio particolare) usati per fornire link a concetti più ampi, concetti correlati, sinonimi. Offrono un contesto semantico ai termini che l'utente sta cercando.
 - **Algoritmi di ricerca sofisticati.**
 - **Ordinamento "Best Bet":** Ordina per rilevanza dei risultati di ricerca (per similarità e contiguità, per popolarità, per bisogni di business).

9. Il Processo di Progettazione dell'Architettura dell'Informazione

1. Requisiti (Requirements):

- Esaminare il contenuto esistente.
- Incontrarsi con gli stakeholder per discutere obiettivi di alto livello, contesto di business e architettura esistente.
- **Domande Fondamentali sui Requisiti:**
 - *Management:* Obiettivi a lungo/breve termine? Strategie e piani di business? Scadenze e budget?
 - *Analisi Utente:* Chi ci aspettiamo come utenti? Perché dovrebbero venire qui? Perché dovrebbero tornare? Cosa ha funzionato/non funzionato in passato?
 - *Gestione dei Contenuti:* Il contenuto è statico o dinamico? Come sarà creato e da chi? Esiste un sistema di gestione dei contenuti? Aspetti legali (copyright)?
 - *Metadati:* Esistono metadati strutturali (gerarchia, posizione documento), descrittivi (di cosa si tratta) o amministrativi (chi ha prodotto i dati)? Come sono gestiti?
- **Strutture Informative dei Requisiti:**
 - *Ecologia Informativa del Sistema:* contenuto, contesto e utenti e la loro relazione.

- *Elementi dell'Architettura Informativa*: aiuti alla navigazione, aiuti alla ricerca, contenuto e compiti/componenti invisibili.

2. Progettazione (Design):

- Blueprint, wireframe della struttura delle pagine e schema dei metadati associati alle pagine.
- È il nucleo della fase di progettazione per l'architettura dell'informazione.
- **Gestire l'architettura dell'informazione**: Approccio top-down o bottom-up? Sistema di organizzazione ed etichettatura (top-down)? Determinare i campi di metadati appropriati? Progettazione del sistema di navigazione? Come integrare strategie top-down e bottom-up?
- **Card Sorting**: Una tecnica di categorizzazione dei contenuti.
 - Scrivere su foglietti o Post-It alcuni compiti specifici del sito web.
 - Su altri foglietti scrivere categorie (ovviamente una categoria per ogni post-it).
 - Utenti e/o clienti sono invitati a raggruppare e associare i compiti con le categorie tematiche del sito.
 - **Card Sorting Aperto (Open card sorting)**: l'utente può scegliere tra le categorie proposte o inventarne di nuove (per scoprire nuove categorie). Basta aggiungere post-it vuoti e una penna.
 - **Card Sorting Chiuso (Closed card sorting)**: l'utente può scegliere solo tra le categorie proposte (usato per confermare un sistema di etichettatura).
 - **Card Sorting Inverso (Reverse card sorting)**: Invece di organizzare tutte le categorie in un unico mazzo, sono pre-organizzate in categorie fittizie (o quelle attuali del sito, ecc.) e si chiede all'utente di riorganizzarle.
 - Esistono applicazioni (es. WebSort <http://www.websort.net/>) che permettono il card sorting remoto.

3. Implementazione (Implementation):

- I progetti vengono implementati e testati (tester di prototipi e designer stessi in modo iterativo).

4. Gestione (Management):

- Valutazione continua dell'architettura: nuovi documenti vengono aggiunti ed etichettati e la validità dei metadati precedenti viene verificata. Richiede anche feedback dagli utenti, utile per riprogettare le operazioni.

10. Alcuni Tipi di Strutture Informative

- **Struttura Piatta (Flat structure):**

- Utile per siti molto piccoli, come brochure o eventi semplici. (Homepage con poche pagine direttamente collegate).

- **Indice dei Contenuti (Table of content):**

- Probabilmente il tipo di struttura più comune: c'è un unico punto di accesso e un unico percorso dalla pagina principale a ogni contenuto. (Struttura ad albero semplice).

- **Organizzazione Strettamente Gerarchica (Strictly hierarchical organization):**

- Oltre alla homepage e alle pagine foglia (contenuto finale), c'è una struttura rigidamente gerarchica di navigazione e partizionamento del contenuto.
-

- **Organizzazione Co-Gerarchica (Co-hierarchical organization):**

- Le strutture intermedie sono connesse e ci sono più percorsi per arrivare alle pagine foglia.
- Anche in questo caso, tuttavia, c'è una chiara distinzione tra: homepage, pagine intermedie (navigazione) e pagine foglia (contenuto).

Conclusioni

In questa lezione abbiamo discusso:

- L'importanza di una **buona architettura dell'informazione**.
- Come il **contenuto viene organizzato e gestito** per una buona comprensione.
- E soprattutto, sottolineiamo che il **modo in cui organizziamo il contenuto riflette e influenza il modo in cui i nostri utenti lo percepiscono**.

Un'architettura dell'informazione ben pensata è invisibile all'utente, ma cruciale per il successo di qualsiasi prodotto digitale.

Riferimenti Bibliografici Chiave:

- Davenport, T.H., Prusak, L. (1997). *Information Ecology: Mastering the Information and Knowledge Environment*. Oxford University Press.
- Jacobson, R. (Ed.). (1999). *Information Design*. The MIT Press.
- Morville, P., Rosenfeld, L. (2006). *Information Architecture for the World Wide Web, Designing Large-Scale Web Sites (3rd Edition)*. O'Reilly Media.
- Tufte, E. (1992). *The Visual Display of Quantitative Information*. Graphics Press.

Siti Utili:

- <http://www.jjg.net/ia/> (Jesse James Garrett)
- <http://www.trovabile.org/>
- <http://www.informationarchitecture.it/>
- <http://www.nosolousabilidad.com>

10 - La Valutazione Dell'usabilità: L'ispezione

Introduzione alla Lezione

Benvenuti a questa lezione dedicata a una fase cruciale del processo di design: la **valutazione**. In particolare, oggi ci concentreremo sui metodi di **ispezione**, ovvero quelle valutazioni che avvengono *internamente* al team di sviluppo. Comprendere come e quando valutare è fondamentale per creare prodotti usabili ed efficaci.

1. Modalità di Valutazione

La valutazione dell'usabilità di un prodotto può avvenire principalmente in due modi:

- **Internamente al team di sviluppo:** Questa modalità si attua mentre il prodotto è ancora in fase di sviluppo. Viene chiamata **ispezione del design (inspection of the design)**. È il focus della lezione di oggi.
- **Esternamente al team di sviluppo:** Questa modalità prevede la partecipazione di potenziali utenti esterni. Viene chiamata **testing**. Ne parleremo in una lezione successiva.

2. Cosa Valutare e Quando?

La valutazione può essere applicata a diversi stadi di sviluppo del prodotto, dal più grezzo al più rifinito:

- **Schizzi (Sketches):**
 - Disegni rudimentali su carta (es. fogli A4) mostrati all'utente (o al valutatore interno).
 - L'assistente al test (o un membro del team) cambia manualmente e apertamente i disegni in base alle indicazioni dell'utente/valutatore, simulando l'interazione.
-
- **Mock-Up su Computer:**
 - Un wireframe dell'applicazione viene mostrato all'utente/valutatore.
 - Lo strumento di wireframing o l'assistente cambia manualmente le schermate.
-
- **Mago di Oz (Wizard of Oz):**
 - Viene mostrata un'interfaccia di alta qualità dell'applicazione.
 - "Dietro le quinte", una persona (il "mago") fornisce le risposte e simula il comportamento del sistema, cambiando le schermate. L'utente crede di interagire con un sistema funzionante.
-
- **Prototipo (A prototype):**
 - Viene mostrato un sistema parzialmente funzionante con un'interfaccia finalizzata.
 - Vengono testate solo le parti effettivamente funzionanti.
-
- **Il Sistema Funzionante (The working system):**
 - Valutazione del prodotto completo e operativo.
-

Regola Fondamentale delle Valutazioni:

Quanto prima si valuta il sistema, tanto meno precisi saranno i risultati, ma tanto meno costoso sarà risolvere i problemi individuati.

Di conseguenza, la strategia dovrebbe essere:

- **Valutare presto, valutare spesso.**
- **Rimuovere i grossi errori nelle fasi iniziali** (prima che la loro correzione diventi troppo onerosa).

- Lasciare le valutazioni successive per problemi minori (es. estetici) e per valutazioni formali (es. contrattuali).

3. L'Ispezione (The Inspection)

La fase di ispezione, come detto, avviene **all'interno del team di design**. Per questa ragione, è uno strumento **economico** (non richiede il reclutamento di utenti esterni), ma può essere anche **molto impreciso** (il team è intrinsecamente "viziato" dalla propria conoscenza del prodotto).

Durante la fase di ispezione, si possono svolgere tre attività rilevanti:

- **Cognitive Walkthrough (Percorso Cognitivo):** Un'esecuzione fittizia e passo-passo di un compito, con una valutazione empirica della verosimiglianza di tale "finzione" interattiva.
- **Analisi delle Azioni (Action Analysis):** Un'analisi quantitativa di azioni specifiche che devono essere eseguite per compiere un'azione più complessa.
- **Analisi Euristica (Heuristic Analysis o Applicazione di Linee Guida):** La valutazione delle interfacce basata su regole di buon senso derivate dall'esperienza.

Approfondiamo il Cognitive Walkthrough (CW)

- **Definizione (CW 1/2):**
 - Il Cognitive Walkthrough è un modo **formalizzato di immaginare i pensieri e le azioni degli utenti** quando usano un'interfaccia per eseguire un compito per la prima volta.
 - Si prende il sistema, un prototipo o anche solo una serie di disegni. Si seleziona un compito da eseguire con quell'interfaccia e si racconta una **storia credibile** su ogni azione che l'utente deve compiere per completare il compito.
 - La storia è credibile se si può **motivare ogni azione dell'utente** basandosi sulla conoscenza generale dell'utente presunto (la *persona*) e sulle indicazioni e sul feedback forniti dall'interfaccia. Se non si riesce a raccontare una storia credibile, c'è un problema di interfaccia.
 - **Obiettivo:** Determinare la plausibilità dell'usabilità dell'interfaccia per il segmento di utenti scelto.
- **Ingredienti Necessari per un CW (CW 2/2):**
 - Una **descrizione o un prototipo dell'interfaccia**, il più dettagliato possibile.
 - La **descrizione di un compito**, possibilmente uno di quelli rappresentativi identificati nella progettazione basata sui compiti o sugli obiettivi.

- Una **lista completa e scritta delle azioni necessarie** per completare il compito (a volte chiamato "Happy Path", perché rappresenta la sequenza ideale).
- Una **chiara descrizione dell'Utente** (la *persona*), delle sue capacità e aspettative.
- Basandosi su questi ingredienti, compito per compito, è necessario costruire una storia e valutarne la credibilità.
-
- **Esempio di CW (CW 1/3 - Fotocopiatrice):**
 - *Contesto*: Valutiamo l'interfaccia di una fotocopiatrice.
 - *Interfaccia Fornita*: Disegno di un tastierino numerico, un pulsante "copia" e un interruttore sul retro per accenderla/spegnere. Informazione aggiuntiva: si spegne dopo 5 minuti di inattività.
 - *Compito*: Copiare una singola pagina.
 - *Utente (Persona)*: Una segretaria neo-assunta.
 - *Storia Raccontata*: "La segretaria deve fare una copia e sa che la fotocopiatrice deve essere accesa, quindi preme il pulsante di accensione. Inserisce il foglio nella fotocopiatrice e preme il pulsante 'copia'."
-
- **Analisi dell'Esempio (CW 2/3):**
 - Questa storia **non è molto credibile**:
 - Come sa la segretaria che la fotocopiatrice è spenta?
 - Come sa dov'è il pulsante di accensione (se è sul retro)?
 - Come sa come inserire il foglio?
 - Siamo sicuri che sappia che il pulsante sopra il tastierino significa "copia"?
 -
 - **Attenzione**: Non dobbiamo valutare l'interfaccia in sé, ma la **credibilità della storia che**. Questo può portare a revisioni sia dell'interfaccia sia della storia.
 - *Possibili Miglioramenti*: Aggiungiamo un display che indica quando la fotocopiatrice è pronta, un disegno frontale che mostra come inserire i fogli, spostiamo l'interruttore principale in posizione visibile, cambiamo il pulsante in "Copia" (se non lo era già) e ripetiamo l'esperimento narrativo. E così via.
-
- **Problemi Scoperti dal CW (CW 3/3):**
 - **Assunzioni del designer sul ragionamento degli utenti**: ("Perché l'utente dovrebbe pensare che la fotocopiatrice debba essere accesa?")
 - **Comandi ovvi per il designer ma non per l'utente**: ("L'utente sa che vuole accendere la macchina, ma sa dove trovare l'interruttore?")

- **Problemi con etichette e prompt:** ("Come sa come inserire il foglio, e siamo sicuri che capisca l'icona?")
- **Problemi di feedback:** ("Come sa se la fotocopiatrice è accesa o spenta?")
- **Errori Comuni nel CW:**
 - **Confondere la lista delle azioni con il walkthrough stesso:** Lo scopo del CW è raccontare credibilmente come l'utente esegue le azioni ottimali per completare il compito, non descrivere l'utente mentre scopre queste azioni.
 - **Confondere il CW con un test utente reale:** Il CW identifica una classe di problemi che un test con 5-10 utenti potrebbe non identificare, ma il test reale con gli utenti identifica aspetti reali che non possono essere scoperti solo con il CW.
- **CW – Differenze con gli Scenari:**
 - Uno **scenario non include un prototipo dell'interfaccia né la descrizione dettagliata delle azioni** (l'happy path) per svolgere un compito.
 - Uno **scenario mira a costruire l'interfaccia e l'happy path**, non a valutarli.
 - Gli **scenari sono credibili per costruzione**, mentre i **CW diventano accettabili dopo essere diventati credibili** attraverso iterazioni.
- **CW - Auto-Valutazione:** Domande da porsi durante l'auto-valutazione dei risultati del CW:
 - È realistico che il personaggio (persona) cercherà di compiere questa specifica azione?
 - Il controllo che comanda questa azione è disponibile?
 - C'è un legame ovvio tra controllo e azione?
 - Il feedback è appropriato?

4. Analisi delle Azioni (Action Analysis)

L'analisi delle azioni è un processo di valutazione che esamina da vicino la sequenza di azioni da eseguire per completare un compito. Esistono due tipi:

- **Analisi Formale delle Azioni (Keystroke-Level Analysis - KLA):**
 - Caratterizzata da un dettaglio estremo nella descrizione delle azioni.
 - Può predire con un margine del 20% il tempo effettivo di completamento del compito, il tempo medio per imparare un'interfaccia, il rapporto tra azioni ed errori.
 - Sfortunatamente, è molto complicata e lunga da eseguire.
 - **Obiettivo:** Calcolo specifico dei tempi medi di utilizzo di un sistema (o, più spesso, di un widget).

- L'approccio formale all'analisi delle azioni viene utilizzato per fare **previsioni accurate del tempo impiegato da un utente esperto** nell'eseguire un compito. Per fare ciò, è necessario stimare i tempi per eseguire ogni passo (fisico e mentale) del compito e sommarli. Il passo tipico è la pressione di un tasto, da cui il nome "analisi a livello di battuta".
 - La stima del tempo di ogni azione deriva da tabelle ottenute testando centinaia di utenti, migliaia di azioni individuali, in migliaia di situazioni, e poi mediando. Se un controllo non è descritto in queste tabelle, o lo si approssima o si eseguono test simili sul nuovo controllo.
 - Per ottenere il tempo necessario per eseguire un compito, quindi, si deve adottare un approccio top-down nella descrizione del percorso ottimale, e poi associare i tempi appropriati a ogni azione individuale.
 - Un'analisi formale di un'interfaccia complessa è un compito arduo: 10 minuti di un'azione possono richiedere la descrizione di mille azioni e i relativi tempi. Inoltre, la descrizione del compito e delle azioni degli utenti è a discrezione del valutatore, quindi possono esserci discrepanze significative tra diverse analisi. Per questo motivo, l'analisi delle azioni è utile solo in poche circostanze speciali:
 - Per esaminare aspetti molto specifici di un'interfaccia (es. le prestazioni di un nuovo widget di una GUI).
 - Per esaminare compiti molto strutturati e controllati (es. analizzare il carico di lavoro di un operatore telefonico di un helpdesk online).
-
- **Modello GOMS (Goals, Operators, Methods, Selection rules) per l'Analisi delle Prestazioni:**
 - I modelli cognitivi descrivono essenzialmente:
 - **Competenza** (conoscenza di sequenze di comportamenti atomici), o
 - **Prestazione** (velocità di esecuzione, ma solo per compiti di routine).
 - L'analisi delle azioni è un modello di prestazione per descrivere obiettivi e compiti. GOMS è il modello più antico di Analisi delle Azioni.
 - **Si basa su:** Obiettivi (Goals), Operatori (Operators), Metodi (Methods), Regole di Selezione (Selection rules).
 - **Input:** una descrizione dettagliata dell'interfaccia e dei compiti.
 - **Output:** misure quantitative e qualitative.

- **Tempi Medi per Azioni di Interfaccia (esempi da Olson & Olson, 1990):**

- *Percezione Visiva:*

- Rispondere a una breve luce: 0.10 s (varia con intensità, da 0.05 s).
 - Riconoscere una parola di 6 lettere: 0.34 s.
 - Spostare gli occhi su una nuova posizione sullo schermo principale: 0.23 s.

- *Movimenti Fisici:*

- Digitare un tasto sulla tastiera: 0.28 s (0.2 s per dattilografi esperti, più lungo per sequenze casuali).
 - Usare il mouse per puntare un oggetto sullo schermo: 1.50 s (leggermente inferiore per schermi piccoli, aumenta con schermi grandi/oggetti piccoli).
 - Spostare la mano su un dispositivo di puntamento o tasto funzione: 0.30 s.

- *Azioni Mentali:*

- Recuperare un elemento semplice dalla memoria a lungo termine: 1.20 s (es. abbreviazione comando "dir").
 - Imparare un singolo "passo" in una procedura: 25.00 s (minimo 10-15 s, escluso tempo di addestramento).
 - Eseguire un "passo" mentale: 0.07 s.
 - Scegliere tra metodi: 1.20 s (varia da 0.06 a 1.8 s a seconda della complessità).

- **Modello KLM (Keystroke-Level Model):**

- Una delle molte evoluzioni di GOMS ancora ampiamente utilizzata. Prevede o stima quanto tempo impiegherà un **utente esperto** per completare un **compito di routine** con uno strumento software
 - Composto da sei operatori:

- **K:** battuta di tasto o pressione di pulsante (premere Shift + A sono due azioni K).
 - **P:** puntamento con il mouse (movimento separato dal click K).
 - **H:** homing (spostare mani/dita sulla tastiera o altro dispositivo, es. da tastiera a mouse).
 - **D:** disegno manuale (non usato frequentemente).
 - **M:** preparazione mentale (tempo per pensare, pianificare, decidere).
 - **R:** tempo di risposta del sistema, o tempo di attesa (anche: W).
- *Esempio di utilizzo KLM:* Inserire l'indirizzo "Via Irnerio 36" in un campo di testo.
- Valori standard: K (dattilografo medio 40ppm) = 0.28s, B (pressione pulsante mouse) = 0.1s, P = 1.1s, H = 0.4s, M = 1.35s.
 - Sequenza: Iniziare l'azione (M), Trovare il campo di testo corretto (M), Portare il puntatore del mouse sul campo (P), Premere il pulsante del mouse (B), Rilasciare il pulsante del mouse (B), Spostare le mani dal mouse alla tastiera (H), Digitare "Via Irnerio 36" (14 lettere, due numeri, due maiuscole: 16K – ipotizzando che i numeri siano come tasti minuscoli e ricordando che le maiuscole richiedono 2K ciascuna).
 - Tempo totale = $2M + 1P + 2B + 1H + 16K = 8.88s$.
-
- **Pro e Contro dell'Analisi Formale Keystroke-Level:**
- *Pro:* Pratica (non serve verificare i dati con utenti reali), Generale (non serve conoscenza di modelli psicologici), Utilizzo precoce (non servono prototipi funzionanti).
 - *Contro:* Restituisce solo il tempo come metrica fondamentale di prestazione, Progettata per utenti esperti in compiti di routine e senza errori, Non c'è spazio per apprendimento, esplorazione, esitazione, errori e slips, Modello molto semplificato per le operazioni mentali (M), Produrre la lista completa delle azioni può richiedere molto tempo per compiti lunghi, Più lungo è il compito, più impattante è il numero di operazioni mentali (M) rispetto alle azioni fisiche, e quindi più impreciso è il risultato calcolato.
 - Principalmente utile per compiti brevi e ben noti.

- **Analisi Informale delle Azioni (Back-of-the-Envelope Analysis):**
 - Ignora il micro-dettaglio e si concentra sul quadro generale, elencando una serie "naturale" di azioni e valutandole globalmente.
 - Invece di "prendere la mano dalla tastiera e afferrare il mouse", le azioni descritte sono del tipo: "scegliere l'opzione X dal menu Y".
 - **Obiettivo:** Determinazione euristica dei passaggi con il peso maggiore (in termini di tempo e numero di azioni atomiche) nell'esecuzione di un compito, e quindi le potenziali fonti di eccessiva complessità, perdita di tempo, disorientamento.
 - L'enfasi non è sul decimo di secondo per valutare le prestazioni di un widget, ma sulla valutazione di risposte a domande come:
 - Posso eseguire un compito semplice in modo semplice?
 - Posso eseguire un compito frequente rapidamente?
 - Quanti passaggi e fatti devo imparare prima di poter eseguire un compito?
 - Abbiamo descritto ogni passaggio nella documentazione?
 -
 - Senza essere precisa come l'analisi formale, l'analisi informale è più **robusta e meno soggetta a imprecisioni**, e può essere usata per:
 - Verificare che l'esecuzione di un compito non richieda tempi comparabili a farlo a mano, su carta, con un'applicazione diversa, ecc.
 - Decidere se l'aggiunta di una feature complicherà o meno il resto dell'interfaccia.
 - Decidere se aggiungere più modi per compiere un compito.
 - Controllare quali operazioni possono generare un errore e quanto grave può essere questo errore (in termini di tempo necessario per correggerlo).
 -
-

5. Analisi Euristica (Heuristic Analysis)

Sia il cognitive walkthrough sia l'analisi delle azioni sono valutazioni orientate ai compiti. Anche il test con gli utenti, che vedremo, è orientato ai compiti.

Le valutazioni orientate ai compiti hanno punti di forza e debolezza:

- **Appropriatezza:** Valutano le caratteristiche di un sistema all'interno di un flusso di lavoro credibile e guidato da obiettivi indipendenti dall'interfaccia.
- **Copertura:** Ci permettono di valutare solo pochi compiti, ignorandone molti altri.

- **Interazioni tra compiti:** Ci permettono di valutare come si comporta il sistema quando l'utente esegue molte azioni contemporaneamente.

Obiettivo dell'Analisi Euristica: Verificare l'aderenza del sistema alle linee guida identificate per il progetto e giustificare qualsiasi deviazione (o modificare il design).

- **Valutazione Euristica:**

- Uno strumento di ispezione (quindi eseguibile dal team di sviluppo) per valutare l'usabilità di un sistema **indipendentemente dai compiti** per cui è progettato (è quindi una valutazione indipendente dal dominio).
- Confronto dell'applicazione con principi generali e universalmente riconosciuti.
- Si basa sul principio fondamentale della **coerenza esterna**, applicato sia in senso positivo sia negativo:
 - Se in altre applicazioni la scelta X era positiva, è probabilmente positiva anche in questo sistema.
 - Se in altre applicazioni la scelta Y era negativa, è probabilmente negativa anche in questo sistema.
-
- Questa è chiamata valutazione euristica perché fornisce **linee guida per la scoperta (euristica) di problemi di usabilità**.

- **Linee Guida (Guidelines):**

- "Le linee guida sono belle perché sono tante e variegate"
- **Linee Guida Generali (esempi):**
 - Le 10 euristiche di Nielsen e Molich (1994).
 - Le linee guida di UserFocus.co.uk (commerciale, UK, 2014).
- **Linee Guida Governative (esempi):**
 - US Research-based Web Design and Visibility Guidelines (2006).
 - EU Europa Web Guide (Regole e linee guida per i siti della Commissione Europea).
 - EU Usability guidelines for websites and products of statistical organisations (2020).

- **Le 10 Euristiche di Nielsen e Molich (1994):**

- **Visibilità dello stato del sistema:** Il sistema dovrebbe sempre tenere l'utente informato su ciò che accade, tramite feedback appropriato fornito entro un tempo ragionevole.
- **Corrispondenza tra sistema e mondo reale:** Il sistema dovrebbe parlare il linguaggio dell'utente, con parole, frasi e concetti familiari all'utente piuttosto che termini di sistema. Deve seguire le convenzioni del mondo reale e far apparire le informazioni in un ordine naturale e logico.
- **Controllo e libertà dell'utente:** Poiché l'utente sceglie spesso funzioni di sistema per errore, ha bisogno di "uscite di emergenza" chiaramente contrassegnate per lasciare lo stato indesiderato senza dover passare attraverso un dialogo complesso. Supportare annulla e ripeti.
- **Coerenza e standard:** Gli utenti non dovrebbero chiedersi se parole, situazioni o azioni diverse significano la stessa cosa. Seguire le convenzioni della piattaforma.
- **Prevenzione degli errori:** Ancora meglio di buoni messaggi di errore è un design attento che impedisce che un problema si verifichi in primo luogo. Eliminare le condizioni soggette a errori o verificarle e presentare agli utenti un'opzione di conferma prima che si impegnino nell'azione.
- **Riconoscimento piuttosto che ricordo (Recall):** Minimizzare il carico di memoria dell'utente rendendo visibili oggetti, azioni e opzioni. L'utente non dovrebbe dover ricordare informazioni da una parte all'altra del dialogo. Le istruzioni per l'uso del sistema dovrebbero essere visibili o facilmente recuperabili quando appropriate.
- **Flessibilità ed efficienza d'uso:** Acceleratori – non visti dall'utente inesperto – accelerano l'interazione per l'utente esperto, in modo che il sistema possa soddisfare sia utenti inesperti sia esperti. Permettere agli utenti di personalizzare azioni frequenti.
- **Estetica e design minimalista:** I dialoghi non dovrebbero contenere informazioni irrilevanti o raramente necessarie. Ogni unità extra di informazione in un dialogo compete con le unità di informazione rilevanti e ne diminuisce la visibilità relativa.
- **Aiutare gli utenti a riconoscere, diagnosticare e recuperare dagli errori:** I messaggi di errore dovrebbero essere espressi in linguaggio semplice (senza codici), indicare precisamente il problema e suggerire costruttivamente una soluzione.
- **Aiuto e documentazione:** Anche se è meglio se il sistema può essere usato senza documentazione, potrebbe essere necessario fornire aiuto e documentazione. Tali informazioni dovrebbero essere facili da cercare, focalizzate sul compito dell'utente,

elencare passaggi concreti da eseguire e non essere troppo lunghe.

- **Le Linee Guida di [Userfocus.co.uk](http://www.userfocus.co.uk):**

- Un'azienda privata che fornisce consulenza sulla progettazione e valutazione dell'usabilità di applicazioni e siti web.
- Sul loro sito si trovano articoli, libri (scaricabili gratuitamente) e le linee guida per la valutazione dell'usabilità.
- 247 linee guida organizzate in 9 capitoli. C'è un comodo file Excel da compilare con valori appropriati durante la valutazione, che genera automaticamente un grafico dell'usabilità complessiva dell'applicazione o del sito web.
- (Link: <http://www.userfocus.co.uk/resources/guidelines.html>)
- **Capitoli Principali:** Usabilità della homepage (20 gl), Orientamento ai compiti (44 gl), Navigazione e IA (29 gl), Moduli e inserimento dati (23 gl), Fiducia e credibilità (13 gl), Qualità della scrittura e dei contenuti (23 gl), Layout di pagina e design visivo (38 gl), Usabilità della ricerca (20 gl), Aiuto, feedback e tolleranza agli errori (37 gl).

- **USA: Research-Based Web Design & Usability Guidelines (2006):**

- Nate per il Dipartimento della Salute e dei Servizi Umani del governo USA nel 2003, poi adottate e standardizzate nel 2006 per tutti i siti web governativi USA.
- (Link: https://www.usability.gov/sites/default/files/documents/guidelines_book.pdf)
- 18 capitoli (il primo e l'ultimo dettagliano il processo, gli altri enfatizzano aspetti specifici del design). 209 Linee guida complessive.

- **EU: Europa Web Guide:**

- Regole e linee guida che si applicano ai siti web della Commissione Europea, coprendo aspetti editoriali, legali, tecnici, visivi e contrattuali.
- (Link: https://commission.europa.eu/resources-partners/europa-web-guide_en)
- 10 regole e 4 blocchi di linee guida, che includono una varietà di aspetti non tutti correlati all'usabilità.
- **Regole:** Dominio e sottodomini UE, Branding, Struttura URL, Categorie di siti, Identità visiva, Accessibilità, Soluzioni aziendali, Protezione dei dati, Diritti di proprietà intellettuale, Archiviazione.

- **Linee Guida:** Architettura e Navigazione, Raccomandazioni di design, Linee guida sui contenuti, Ottimizzazione per i motori di ricerca (SEO).
- **EU: Usability guidelines for websites and products of statistical organisations (2020):**
 - Migliori pratiche e raccomandazioni per la progettazione di siti web e altri strumenti online utilizzati per la diffusione di statistiche ufficiali.
 - (Link: https://cros-legacy.ec.europa.eu/system/files/usability_guidelines_for_websites_and_products_of_statistical_organisations.pdf)
 - Pensate per gli uffici UE che pubblicano tabelle statistiche di impatto a livello UE, ma contengono molti consigli utili per i siti web informativi della Pubblica Amministrazione in generale.

11 - La valutazione dell'usabilità: Il Testing

Introduzione alla Lezione (Ripasso e Prospettiva)

Bentornati. Nelle lezioni precedenti abbiamo discusso l'importanza della valutazione nel ciclo di vita del design e abbiamo approfondito i metodi di **ispezione**, ovvero quelle valutazioni condotte *internamente* al team di sviluppo (come il Cognitive Walkthrough, l'Analisi delle Azioni e l'Analisi Euristica).

Oggi, spostiamo il nostro focus sulla seconda principale modalità di valutazione: il **testing**. Questa forma di valutazione avviene **esternamente al team di sviluppo**, con la partecipazione cruciale di **potenziali utenti esterni**. Esploreremo perché il testing è indispensabile, le sue diverse categorie, i costi e i benefici, e come strutturare un processo di testing efficace.

Ricordiamo brevemente **cosa possiamo valutare** e la **regola fondamentale delle valutazioni**:

- **Cosa valutare:** Dagli schizzi iniziali, ai mock-up, ai prototipi (anche con la tecnica del Mago di Oz), fino al sistema funzionante.
- **Regola fondamentale:** *Valutare presto, valutare spesso*. Prima si valuta, meno precisi i risultati, ma meno costoso correggere i problemi.

1. Perché Continuare a Fare Testing?

Nonostante l'utilità delle tecniche di ispezione, perché è ancora fondamentale condurre test con utenti reali?

- **Limiti delle Tecniche Analitiche:** Nessuna teoria, nemmeno con le tecniche analitiche più raffinate, troverà tutti i problemi software o di usabilità.
- **Problemi Reali vs. Problemi Rilevati:** I problemi scoperti con le tecniche di ispezione non sono sempre problemi effettivamente riscontrati dagli utenti. Il team di design può avere bias o fare assunzioni errate.
- **Realismo e Credibilità:** L'aiuto di utenti reali è essenziale per avere considerazioni realistiche sul sistema, che siano credibili e dimostrabili. Gli utenti portano prospettive e comportamenti che il team interno potrebbe non prevedere.
- **Differenze Individuali:** Tuttavia, è importante ricordare che gli utenti presentano significative differenze individuali (in termini di preparazione tecnica, attitudine psicologica, forma fisica, ecc.), che possono influenzare i risultati del test.

2. Categorie di Test Utente

I test utente possono essere classificati secondo diversi criteri:

- **Scopo (Purpose):**
 - **Test Formativi (Formative tests):** Hanno lo scopo di **raccogliere problemi** (sospettiamo che ce ne siano e vogliamo prove di dove si trovano). Sono orientati al miglioramento continuo durante lo sviluppo.
 - **Test Sommativi (Summative tests):** Hanno lo scopo di **verificare la correttezza** (crediamo di aver risolto tutti i problemi e vogliamo prove che siano stati risolti). Valutano il prodotto finale rispetto a benchmark o requisiti.
- **Risultato (Outcome):**
 - **Quantitativo (Quantitative):** Producono dati "duri" (numeri, statistiche), difficili da obiettare, ma spesso dimostrano solo l'esistenza di un problema senza spiegarne la causa.
 - **Qualitativo (Qualitative):** Producono dati "morbidi", impressioni soggettive, opinioni, ma possono fornire idee e spunti per la soluzione.
- **Fase (Phase):**

- **Precoce (Early):** Durante le varie fasi di sviluppo, con un sistema consapevolmente incompleto, sperando di catturare i problemi presto e risolverli a basso costo.
- **Tardiva (Late):** Alla fine dello sviluppo, con il sistema completo, sperando di verificare il superamento con successo di tutti i requisiti di usabilità.
- **Metodologia (Methodology):**
 - **Dati Statisticamente Significativi:** Richiede poche variabili controllate contemporaneamente e un numero statisticamente significativo di soggetti. Difficile obiettare ai risultati del test.
 - **Buon Senso (Common sense):** Si può testare qualsiasi cosa, su qualsiasi numero di soggetti, ma è più facile obiettare ai risultati.

Test Formativi vs. Sommativi

- Un **test formativo** è eseguito a beneficio del **team di design**, che è consapevole che il design non è ancora perfetto e ha bisogno di sapere dove e come migliorarlo.
- Un **test sommativo** è eseguito a beneficio del **cliente** (o stakeholder), che ha bisogno di essere sicuro che il design soddisfi gli **obblighi contrattuali** concordati all'inizio del progetto.
- Quindi, ci possono essere molti test formativi, ma **non si dovrebbe tentare un test sommativo a meno che non si sia fiduciosi che il design sia valido.**
- Inoltre, i test sommativi sono solitamente eseguiti sul prodotto completo o molto vicino al completamento.

Cicli di Valutazione (Evaluation Cycles)

Un tipico ciclo di valutazione potrebbe assomigliare a questo:

1. **Design Iniziale**
2. **Ispezione** (interna al team) -> identifica *issues*
3. **Ri-progettazione** (basata sui feedback dell'ispezione)
4. **Test Formativi** (con pochi utenti) -> identifica altre *issues*
5. **Ri-progettazione** (basata sui feedback dei test formativi)
6. (Eventuali ulteriori cicli di test formativi e ri-progettazione)
7. **Test Sommativi** (con un campione più ampio, per la validazione finale)
 - Se emergono *issues* gravi qui, significa che le fasi precedenti non sono state sufficientemente approfondite.
8. **Uscita (Exit)** / Rilascio del prodotto.

- I **test formativi** sono informali, privati, economici e frequenti.
- Il **test sommativo** è spesso pubblico (i risultati possono essere condivisi con il cliente) e dovrebbe essere tentato solo quando si è fiduciosi che il design passerà.
- **Un test sommativo NON è un test formativo che casualmente non produce problemi.** Ha uno scopo e una metodologia distinti.

"Evaluation cycles for the UUXD project" - Indicazioni specifiche *Per il progetto UUXD (User & Usability Experience Design)*:

- È necessario un **testing sommativo** (con due soggetti che devono essere gli stessi del test sulla versione esistente dello strumento, per confronto).
- Si possono fare **zero, uno o più test formativi** a seconda della fiducia nella qualità del proprio design.
- Nessun soggetto dei test formativi può essere usato per i test sommativi (per evitare bias da apprendimento).
- **Documentare tutto** (fondamentale in qualsiasi progetto).

3. Tipi di Test di Usabilità

- **Full Usability Testing (Test di Usabilità Completo o "Deluxe"):**
 - **Formalizzato:** Protocollo, ruoli, infrastrutture e attori ben definiti.
 - **Quantitativo:** Valutazioni oggettive e statisticamente giustificate.
 - **Parallelo:** Utilizzo di un numero sufficiente di partecipanti (es. tra 20 e 100) per dare solidità ai risultati statistici.
 - **Costoso:** Ogni partecipante riceve un compenso, costi del team di testing, ecc. Inoltre, lo sviluppo è sospeso durante il test.
 - **Utile solo come testing finale o sommativo** (verifica della conformità ai requisiti).
- **Discount Usability Testing (Test di Usabilità Scontato o "Guerrilla"):**
 - Anche noto come **Guerrilla Usability Testing**.
 - **Informale:** Un membro del team discute con il partecipante davanti al computer in una piccola stanza.
 - **Intuitivo:** I risultati dovrebbero essere considerati indicativi e non conclusivi.
 - **Sequenziale:** Gli input di OGNI test vengono valutati e risolti prima di iniziare il test successivo.
 - **Economico:** Pochi utenti (3-4), nessun specialista, parallelo alla produzione.
 - **Utile come test formativo** (identificazione dei problemi da risolvere il prima possibile).

I Costi di un Test di Usabilità

Dipendono molto dall'accuratezza dei risultati che si vogliono ottenere. Consideriamo alcuni fattori (per un test "deluxe"):

- **Candidati da testare:** 20-100 persone. Se fatto professionalmente, bisogna pagare i candidati (es. 25-50€). Se sono studenti, i costi sono inferiori (es. 25€ per test). Se sono dipendenti del cliente, i costi sono ridotti, ma la produttività generale è un problema.
- **Team di psicologi/esperti di usabilità:** Un capo psicologo e un team di assistenti (una persona per ogni test; molti test paralleli significano meno tempo complessivo ma più persone per gestire il test). Due giorni per preparare il setting, il test effettivo (dipende dal setting), due giorni per elaborare i risultati e scrivere il report. Costo minimo: 600€/giorno per il project leader e 300€/giorno per gli assistenti.
- **Infrastruttura:** Una o più postazioni di lavoro con il sistema funzionante e verificato. Tecnici per controllare il corretto funzionamento (un sistema che crasha durante il test rende i risultati INUTILIZZABILI). Eventualmente, telecamere e forse specchi unidirezionali.
- **È facile arrivare a 20.000€ per ogni test di usabilità "deluxe".**

Guerrilla Usability Testing (Dettagli)

Proposto da Jakob Nielsen (1994) per progetti a basso budget (tipici per i siti web):

- **Il problema:** L'alto costo del "deluxe usability testing" (parallelo, alto numero di partecipanti, psicologi professionisti, laboratori attrezzati).
- **La soluzione:** Discount usability testing, che include:
 - **Scenari:** Si basa su prototipi speciali per ridurre i costi.
 - I **prototipi verticali** implementano completamente una piccola parte di una funzionalità.
 - I **prototipi orizzontali** implementano una piccola parte dell'intero spettro di funzionalità.
 - Gli **scenari** (come prototipi) implementano solo una piccolissima parte delle funzionalità, strumentale all'esecuzione del compito.
 - Essendo piccoli e veloci da implementare, possono essere creati e usati con grande velocità e "buttati via" una volta terminato.
 -
 - **Thinking Aloud Informale (Pensare ad Alta Voce):**
 - Senza psicologi, laboratori, telecamere; solo una scrivania, un computer, due sedie.

- Gestito direttamente da un membro del team (appositamente preparato).
- Seguito immediatamente da una fase di valutazione euristica (basata anche sull'input del partecipante).
-
- **Valutazioni Euristiche (Informali):** Applicazione diretta delle euristiche di usabilità (es. Nielsen e Molich), basata anche sull'input del partecipante.
-
- **Vantaggi:** Sequenziale, basso numero di partecipanti (3-4), nessun laboratorio professionale o speciale.
- **Svantaggi:** Incapace di fornire metriche (es. per un confronto), ma solo suggerimenti per il miglioramento.

Giustificazione del Discount Usability Testing

Se si organizzano gli errori per gravità:

- **Catastrofico:** L'utente non completa il compito.
- **Grave:** L'utente completa il compito con significativi rallentamenti e/o con un considerevole compromesso sulla qualità dell'output.
- **Cosmetico:** L'utente identifica lievi rallentamenti o fastidi nell'esecuzione di uno o più compiti.

Il problema è che il testing "deluxe" non identifica tutti i problemi di un sito, anche con un gran numero di utenti:

- Molti utenti trovano gli stessi problemi gravi.
- Meno utenti trovano errori meno gravi.
- Gli errori gravi tendono a mascherare errori meno gravi.
- Solo dopo molte interazioni si iniziano a trovare errori cosmetici.

Il **discount usability testing** è **sequenziale** e quindi permette di accelerare il ciclo di identificazione degli errori:

- **THEN:** Il primo partecipante trova una considerevole percentuale di errori catastrofici (anche l'80%). Vengono valutati e risolti.
- **THEN:** Il secondo partecipante trova alcuni errori rimanenti (pochi catastrofici e molti gravi). Vengono valutati e risolti.
- **THEN:** Il terzo partecipante trova alcuni errori rimanenti (nessuno catastrofico, alcuni gravi e alcuni cosmetici). Ecc.

- In test sistematici si è scoperto che **dopo 5 iterazioni sono stati trovati il 100% degli errori catastrofici, l'80% degli errori gravi e il 50% di quelli cosmetici.**

Due elementi chiave per la convenienza del Discount UT:

1. La **valutazione della gravità dell'errore** è fatta da un designer, basata sulla sua interpretazione personale (e illuminata) delle difficoltà di un partecipante. Con il test deluxe, invece, lo psicologo riflette sui risultati complessivi e identifica l'esistenza e la gravità dell'errore basandosi su valori statistici oggettivi.
2. Le **correzioni ai problemi sono immediate**, e il partecipante successivo usa già un prodotto notevolmente migliorato rispetto al precedente. Così si possono identificare errori aggiuntivi, che all'inizio erano stati mancati. Con il test deluxe, invece, la ricerca di metriche oggettive richiede inesorabilmente che tutti i partecipanti abbiano lo stesso sistema da testare, per fornire statistiche significative.

4. Altri Tipi di Test

- **Tradizionale, Moderato:**
 - 5-10 persone con un moderatore, in un laboratorio. Possibile uso di specchi unidirezionali, keystroke logger, osservatori silenziosi.
 - Controllo completo delle variabili coinvolte, ma costoso.
-
- **Online (Non Moderato):**
 - Creazione di un sito web di test, reclutamento di persone (possibilmente anche centinaia), valutazione solo dei dati restituiti dall'applicazione del test.
 - Nessun controllo sul contesto d'uso, sull'impegno dei partecipanti o sulla tranquillità dell'ambiente, ecc. Grande potenziale di outlier (dati anomali).
 - Potenzialmente una grande quantità di dati su cui effettuare calcoli complessi per ottenere statistiche significative.
-
- **Sondaggio Online (Online Survey):**
 - Solo qualitativo, soggettivo e volontario. Tipicamente questionari e richieste di proposte.
-

5. Variabili di Valutazione

- **Quantitative (prestazione):**
 - **Efficienza (Efficiency):** tempo di esecuzione, esitazioni, ecc.

- **Efficacia (Effectiveness):** passi inutili, vicoli ciechi, backtracking, ecc.
-
- **Qualitative (soggettive):**
 - **Soddisfazione (Satisfaction):** parole positive, confronti positivi, passaparola positivo, ecc.
-

6. Fasi del Test

Un processo di testing ben strutturato comprende diverse fasi:

1. **Protocollo di Test:** Definizione del tipo e scopo del test, logistica, metodologia di valutazione.
2. **Selezione e Preparazione degli Assistenti.**
3. **Test Pilota (Dry Run / Prova generale).**
4. **Scelta dei Partecipanti.**
5. **Esecuzione del Test.**
6. **Valutazione Finale e Report.**

Il Protocollo di Testing

Dovrebbe definire:

- Approccio al testing (discount o full usability).
- Lista dei compiti da testare.
- Numero di test.
- Numero di soggetti.
- Metodologia di testing (es. Thinking Aloud o Bottom Line Data) e numero di test per metodologia.
- Descrizione dei risultati attesi (es. metodologia EEE: Effectiveness, Efficiency, Emotions).
- Scelta dei soggetti e giustificazione rispetto al target utente.
- Organizzazione del test, stesura delle fasi, del documento di presentazione iniziale e del questionario di valutazione finale.

Una Classificazione degli Errori (Nielsen)

La gravità di un problema di usabilità dipende da una combinazione di tre fattori:

- **Frequenza del problema:** Es. il numero di utenti diversi per i quali il problema viene notato.
- **Impatto del problema:** Es. l'effetto che il problema ha sul completamento del compito.

- **Persistenza del problema:** Es. il numero e la rilevanza dei compiti che sono affetti dal problema.

Per l'**impatto**, Nielsen propone la seguente classificazione:

- **Errore di Implementazione:** il completamento del compito richiede una feature che l'utente conosce ma che, sebbene annunciata dall'interfaccia, manca o non funziona. **Non è un errore di usabilità!** È un bug.
- **Fallimento Catastrofico:** l'utente non può eseguire il compito.
- **Errore Maggiore:** l'utente completa il compito dopo un tempo significativo e domande rilevanti, dubbi e scelte sbagliate.
- **Errore Minore:** l'utente esegue il compito in un tempo eccessivo e/o con un gran numero di errori e/o notati dall'utente stesso.
- **Errore Cosmetico:** l'utente completa il compito in un tempo ragionevole ma si vede una possibilità di miglioramento.

Analisi dei Dati Soggettivi e Oggettivi

I test genereranno una grande quantità di dati che dovrebbero essere organizzati e descritti. Esempi di osservazioni:

- "Tutti gli utenti si sono lamentati di un layout poco chiaro delle informazioni e di strumenti di ricerca poco raffinati."
- "Per il compito è necessario conoscere in anticipo la sezione giusta, altrimenti l'informazione può essere trovata solo navigando attraverso tutte le sezioni o tramite ricerca libera, evitando attentamente il primo risultato che è sempre 'un set completo di accessori da cucina'."
- "Gli utenti hanno prima cercato una sezione contatti o indirizzi email. Qualcuno ha provato la sezione 'vetrine'. Solo uno dei tre soggetti del test che è riuscito a completare il compito ci è riuscito per caso."
- "Nessuno è stato in grado di completare il compito X."

Curva di Urgenza (Urgency Curve)

- La curva di urgenza è un grafico bidimensionale (come "impatto vs. frequenza" o "impatto vs. persistenza").
- Tutti gli errori identificati vengono tracciati sul grafico.
- Decidendo una soglia di urgenza (al di sopra: correggere immediatamente; al di sotto: prossima release) si traccia la curva che separa i problemi urgenti dagli altri.

Stesura del Protocollo di Test (Dettagli)

- **Tipo di test:** Formativo (suggerimenti per migliorare l'applicazione) o Sommativo (verifica del soddisfacimento dei requisiti iniziali).
- **Scopo del test:**
 - **Globale:** tutte le funzioni per tutti i compiti per tutti gli utenti.
 - **Verticale:** tutte le feature rilevanti per uno o più compiti per uno o più tipi di utente.
 - **Orizzontale:** una o poche funzionalità per tutti i compiti rilevanti e tutti i tipi di utenti interessati.
-
- **Logistica:**
 - **Setting:** ambiente, attrezzatura di test, attrezzatura di registrazione.
 - **Assistenti:** team, competenze di dominio richieste, competenze tecniche.
 - **Partecipanti:** modalità di selezione, numero di partecipanti.
-
- **Metodologia:**
 - Dati di processo (qualitativi) o dati "bottom-line" (quantitativi).
 - Test qualitativo o quantitativo.
 - Interpretazione significativa o analisi statistica.
-

Test Pilota (Dry Run)

- Un test da eseguire "fuori quota" (i dati non entrano nelle statistiche).
- Il partecipante può anche essere un membro del team.
- Serve a verificare che tutto sia a posto:
 - L'attrezzatura funziona.
 - Le domande sono consistenti.
 - Il team di assistenti è preparato.
 - La durata del test è ragionevole.
 - Le "uscite" (es. completamento del task) funzionano.
 - **Il sistema funziona!!!**
-

Come Scegliere gli Utenti

- Ci deve essere **similarità o approssimazione con i segmenti di utenti target**.
- **Problemi Etici:**
 - Stress psicologico.

- Imbarazzo.
- Pressione ambientale.
- Gestione della privacy.
-
- **Soluzione:**
 - Consenso informato.
 - Monitoraggio costante del rilassamento dei partecipanti.
 - Nei casi peggiori: "congelamento" dell'attrezzatura (interruzione del test).
-

Come Scegliere i Compiti

- Abbiamo già una prima lista di compiti: quelli dell'analisi dei compiti.
- Ma a volte questi sono troppo lunghi o richiedono conoscenze di base che non ci si può aspettare dai partecipanti.
- Usare il compito descritto è positivo anche perché il compito non è troppo frammentato.
- I compiti della fase di analisi dei compiti sono già scelti per essere complessi.
- Ovviamente, quando si progettano test verticali, possiamo espandere e complicare la lista dei compiti originariamente pianificati.

7. Tipi di Dati di un Test

- **Dati di Processo (Process data) - Valutazione QUALITATIVA:**
 - Osservazioni su ciò che i partecipanti stanno facendo durante il compito.
 - Descrivono sia cosa gli utenti stanno facendo sia perché lo stanno facendo.
-
- **Dati "Bottom-line" (Risultato finale) - Valutazione QUANTITATIVA:**
 - Raccolte di dati sul tempo richiesto per il compito, il grado di successo, il numero e la qualità degli errori, ecc.
 - Forniscono solo informazioni sul "cosa", non sul "perché".
 - Usano metriche oggettive (es. EEE: Efficacia, Efficienza, Emozioni – quest'ultima spesso misurata qualitativamente ma può essere quantificata).
-

Non si possono MAI raccogliere entrambi i tipi di dati con lo stesso test: devono essere test separati. Questo perché l'atto di pensare e parlare ad alta voce (necessario per i dati di processo) altera i tempi delle azioni e il tipo di ragionamento del partecipante, invalidando le metriche quantitative.

Dati di Processo: Thinking Aloud (Pensare ad Alta Voce - TA)

- Fondamentalmente: l'assistente chiede al partecipante di eseguire un compito e, allo stesso tempo, di **parlargli ad alta voce di ciò che sta facendo**.
- Dobbiamo chiedere ai partecipanti di dire:
 - Cosa stanno pensando.
 - Cosa stanno cercando di fare.
 - Come pensano di dover procedere.
 - Quali dubbi stanno sorgendo.
 - Cosa leggono e come questo si collega al compito.
- **TA: Preparare i Partecipanti:**
 - "Dimmi cosa stai pensando mentre esegui questo compito."
 - "Non sono interessato ai tuoi pensieri segreti, ma solo a ciò che pensi riguardo al completamento del compito."
 - **Enfatizzare che è il sistema ad essere sotto test, non il partecipante.**
 - Spiegare che tipo di registrazione viene effettuata e qual è la politica di gestione della privacy.
- **TA: Il Ruolo dell'Assistente:**
 - L'assistente **deve**:
 - Spingere il partecipante ad agire e parlare contemporaneamente.
 - Aiutare il partecipante se è permanentemente bloccato.
 - **Deve evitare distorsioni:**
 - Nessuna domanda/risposta che possa influenzare.
 - Alcune parole scelte male possono concentrare l'attenzione su aspetti dell'interfaccia che sarebbero stati ignorati.
 -
 - *Domande Buone*: "Dimmi cosa stai pensando", "Cosa stai cercando di fare ora?", "Continua a parlare".
 - *Domande Cattive*: "Secondo te, qual è la funzione dei pulsanti a sinistra?" (focalizza l'attenzione su una parte dell'interfaccia che l'utente stava ignorando), "Perché hai fatto questa azione?" (fa dubitare l'utente di aver sbagliato).

- È appropriato specificare: "In questo test è permesso, anzi incoraggiato, fare domande, ma NON mi sarà permesso fornire la risposta."
- Si dovrebbe aiutare il partecipante solo dopo aver verificato che è effettivamente bloccato, e solo per impedirgli di abbandonare il test.

Dati Bottom-Line

- L'utente è **solo** nell'eseguire i compiti, senza intervento degli assistenti.
- I dati vengono raccolti invisibilmente dall'ambiente, dal sistema stesso, o dall'assistente (che NON interagisce con l'utente).
- Alla fine, all'utente può essere chiesto di compilare alcuni questionari.
- I dati quantitativi vengono valutati con strumenti statistici. I dati qualitativi (dai questionari) vengono esaminati in massa e in modo anonimo.
- Dati quantitativi: es. il tempo per il completamento di un compito, il numero di errori commessi, il numero di alternative esplorate, la soddisfazione dell'utente, ecc.

8. Metriche Frequenti (Bottom-Line)

- **Quantitative:**
 - **Successo (Success):** Il compito è stato completato? (Sì/No, o con sfumature).
 - **Tempo (Time):** Quanto tempo è stato impiegato?
 - **Errori (Errors):** Quanti e quali errori sono stati commessi?
 - **Efficienza (Efficiency):** Rapporto tra successo e risorse impiegate (es. tempo, click).
 - **Apprendibilità (Learnability):** Come migliorano le prestazioni nel tempo?
- **Qualitative:**
 - **Soddisfazione (Satisfaction):** Raccolta tramite questionari, interviste.

Scenari Tipici per i Test e Metriche Associate

- **La Metrica del Successo:**

- Valutare il completamento di un compito in modo binario (sì/no) o in modo sfumato (percentuale di completamento).
 - *Possibili sfumature:* compito completato con/senza aiuto, compito completato parzialmente con/senza aiuto, compito fallito (ma l'utente pensava di averlo fatto / te ne sei accorto tu).
 - *Tipi di fallimento:* Abbandono, Interruzione del moderatore, Esaurimento del tempo concesso, Risposta sbagliata.
- **La Metrica del Tempo:**
 - Misurare quanto tempo passa tra l'inizio e la fine dell'esecuzione di un compito.
 - *Chi tiene il tempo?* Un umano? Un software? Come si determina che il compito è finito? Come funziona per compiti molto brevi o molto lunghi?
 - Suddividere i tempi in categorie (es. percentili) e attivare soglie di accettabilità (es. qualsiasi completamento dopo X minuti è considerato non riuscito).
 - *Solo compiti completati o tutti i compiti?*
 - Se si utilizza anche un test di thinking aloud, i tempi non sono significativi.
- **La Metrica degli Errori:**
 - *Cos'è un errore?* Inserire un valore sbagliato in un campo, Selezionare un'opzione errata in un menu, Eseguire una sequenza di azioni errata, Non attivare un'azione fondamentale al momento giusto o affatto.
 - Come gestire gli errori che si propagano ad altri errori (effetto a cascata)? Come gestire errori ripetuti?
- **La Metrica dell'Efficienza:**
 - *Cosa si misura?* Numero di click, numero di battute, numero di visualizzazioni di pagina.
 - *Quando inizia l'azione?* Come si valuta, ad esempio, la lettura di una pagina web o il tempo per raggiungere un pulsante con il mouse?
 - *Come contare le azioni?* Scegliere una voce di menu è un'azione o è composta da molte azioni connesse? E usare una scrollbar?

- Un modo semplice per valutare l'efficienza è considerarla come una **relazione tra successo e tempo**.
- Il Common Industry Format for Usability Test Reports (ISO/IEC 25062:2006) specifica che la misura chiave dell'efficienza è il rapporto tra la percentuale di completamento dei compiti e il tempo medio per gli stessi compiti.
- **La Metrica dell'Apprendibilità (Learnability):**
 - Due tipi di apprendibilità:
 - Prestazioni per utenti che non sono mai stati esposti a un sistema (o molto brevemente).
 - Usabilità nel tempo. Si presume che le prestazioni del compito, misurate utilizzando le classiche metriche di usabilità, migliorino dopo ripetuti "trials".
 - *Qual è la scala temporale?* Minuti, giorni, anni?
 - *Qual è la frequenza d'uso?* Test multipli in una sessione? Test multipli in sessioni multiple?
 - Ci si aspetta una riduzione dei tempi di esecuzione. È sempre vero? Come determinare se questa riduzione è soddisfacente o no?
- **La Metrica dei Problemi (Issues):**
 - Esempi di problemi osservati:
 - Una scelta che porta il partecipante fuori strada.
 - Un'espressione di frustrazione.
 - Non notare qualcosa che avrebbe dovuto essere notato.
 - Un partecipante che dice che un compito è completato quando non lo è.
 - Un contenuto testuale il cui scopo, significato, ruolo è frainteso.
 - Eseguire un'azione che porta il partecipante più lontano dal completamento.

9. Le Metriche della Soddisfazione

- **Una Parola di Cautela (Wiklund et al, 1992):**
 - "Abbiamo riscontrato che i soggetti sono riluttanti a criticare i design quando viene chiesto loro di assegnare una valutazione al design. Nei nostri test di usabilità, vediamo lo stesso fenomeno anche quando incoraggiamo i soggetti a essere critici.

Ipotizziamo che i soggetti del test sentano che dare una valutazione bassa a un prodotto dia l'impressione che siano persone 'negative', che le valutazioni riflettano negativamente sulla loro capacità di utilizzare la tecnologia basata su computer, che parte della colpa per le scarse prestazioni di un prodotto ricada su di loro, o che non vogliano ferire i sentimenti della persona che conduce il test."

- **Scale Likert:**

- Sono scale psicometriche per questionari. Prendono il nome dal loro inventore, lo psicologo Rensis Likert (USA, 1903-1981). È la più usata nella ricerca tramite sondaggi.
- Gli intervistati specificano il loro livello di accordo su una scala simmetrica per una serie di affermazioni. L'intervallo cattura l'intensità dei loro sentimenti.
- Le scale Likert hanno 5, 6 (senza risposte neutre) o 7 livelli.

- **Questionari Post-Sessione (Post-session metrics):**

- **After-Scenario Questionnaire (ASQ):** Sviluppato da Lewis nel 1995, da somministrare a un soggetto dopo che ha completato uno scenario di compiti progettato. Tre domande dirette con una scala Likert a 7 livelli ("Sono soddisfatto della facilità...", "Sono soddisfatto della quantità di tempo...", "Sono soddisfatto delle informazioni di supporto..."). Il punteggio ASQ è la media delle 3 domande.
- **System Usability Scale (SUS):** Un metodo di testing molto generico, super veloce e "quick and dirty". Inventato nel 1986 da DEC, poi adottato da IBM nel 2006 per la sua semplicità e ragionevolezza. È un test con un protocollo fisso e criteri di valutazione standard. Dieci domande, proposte alternativamente con enunciazione positiva e negativa, a cui rispondere secondo una scala Likert a 5 livelli. Le valutazioni mancanti valgono 3. Nessuna variazione è consentita nella formulazione delle domande. Genera un punteggio da 0 a 100. Punteggi superiori a 68 indicano una buona usabilità. (Dettagli su measuringusability.com/sus.php).
- **SEC (Single Easy Question):** "Complessivamente, quanto difficile o facile hai trovato questo compito?" (Scala da 1-Molto Difficile a 7-Molto Facile). Etichette solo numeriche, domanda diretta formulata con precisione. Eccellente correlazione con metriche più complesse. Non chiede opinioni o spiegazioni, non separa in gruppi difficili da confrontare. È una valutazione "tutto o niente", facile da capire e applicare. Evita il fenomeno tipico degli utenti che segnano solo 7 o 4. Molto utile in casi di competizione tra proposte simili.

- **Misura dell'Aspettativa (Expectation measure):** Misura la relazione tra le difficoltà attese e le difficoltà incontrate.
- **Net Promoter Score (NPS, 2003):** Scopo: valutare la fedeltà del cliente a un marchio o azienda. Composto da una domanda diretta: *"Quanto è probabile che tu raccomandi la nostra azienda/prodotto/servizio a un amico o collega?"* Punteggio da 0 a 10: 9-10 Promotori, 7-8 Passivi, 0-6 Detrattori. $NPS = (\#Promotori - \#Detrattori) / \#Partecipanti$.
- **Microsoft Product Reaction Cards:** Studio del 2002. Un set di 118 termini (vocabolario controllato) usati frequentemente per descrivere le qualità di un software, posti su carte separate. Date ai partecipanti per associare le carte al design. Spesso 118 carte sono troppe, quindi si usa una sottoselezione (15-25). Vantaggio: facile omogeneizzare i risultati. Esempi di termini: Noioso, Occupato, Calmo, Economico, Creativo, All'avanguardia, Emozionante, Costoso, Familiare, Fresco, Impressionante, Innovativo, Ispiratore, Intimidatorio, Vecchio, Professionale, Affidabile, Non professionale.
- **Card Sorting (Ripasso dalla lezione sull'AI):** Tecnica di categorizzazione dei contenuti. Utile anche in fase di valutazione per capire come gli utenti raggruppano le informazioni.
- **Metriche su App Già Rilasciate:**
 - **Web Analytics:** Fornisce dati su visitatori, visite, visualizzazioni di pagina, pagina di arrivo, ecc.
 - **Click-Through Rate (CTR):** Il rapporto tra il numero di visitatori a cui viene mostrato un link o un pulsante e il numero di coloro che ci cliccano.
 - **Drop-Off Rate:** Il rapporto tra il numero di persone che iniziano una procedura complessa e il numero di coloro che la completano.
 - **A/B Test:** Test di confronto tra due ipotesi comparabili dello stesso design. Metà dei soggetti riceve l'opzione A, metà l'opzione B, casualmente. A volte, la maggior parte degli utenti vede il sistema così com'è attualmente, e un piccolo numero vede una pagina leggermente diversa, per vedere la differenza di apprezzamento.
 - **Dati sul Ritorno dell'Investimento (ROI):** Monetizzazione dei dati di usabilità. Ci si concentra solo sui dati che aumentano il valore del prodotto per l'azienda, o perché diminuisce errori costosi con gli utenti interni, o perché aumenta il numero di articoli venduti o diminuisce i costi del call center.
-
- **Dati Qualitativi - Domande Aperte (Open-ended questions):**

- Dare alle persone uno spazio alla fine del sondaggio per scrivere altre informazioni importanti che ritengono rilevanti. Alcune persone si sentono limitate dovendo solo barrare caselle.
- Si possono/dovrebbero separare le domande sui problemi segnalati dalle domande sui suggerimenti per miglioramenti.
- *Come valutare quantitativamente le risposte testuali semplici?* Buon vecchio senso comune, Applicare la grounded theory ai testi.
- **Analisi Quantitativa di Dati Qualitativi: Grounded Theory:**
 - Una metodologia sistematica nelle scienze sociali che implica la costruzione di una teoria quantitativa attraverso la raccolta e l'analisi sistematica di dati qualitativi.
 - Uno studio che utilizza la grounded theory inizia con una raccolta di dati qualitativi (es. testi).
 - I ricercatori esaminano i dati raccolti, identificano idee, concetti o elementi ripetuti e li etichettano con codici.
 - Man mano che vengono raccolti e riesaminati più dati, i codici vengono quindi raggruppati in concetti e poi in categorie.
 - Queste categorie possono diventare la base per una nuova teoria.
 - Uno strumento importante per la grounded theory è la **codifica (coding)**, cioè raccogliere sotto un'unica etichetta molte formulazioni diverse con lo stesso significato o scopo nei dati qualitativi. Questo avviene su due livelli successivi:
 - **Codifica Aperta (Open Coding):** un processo interpretativo in cui i dati vengono scomposti analiticamente con l'intento di ottenere insight sul fenomeno in studio. Gli eventi, le azioni e le interazioni vengono confrontati per arrivare a concetti e categorie (codici). **I codici scelti appartengono ai dati di origine.**
 - **Codifica Assiale (Axial Coding):** l'obiettivo qui è comprendere le relazioni dinamiche tra le categorie, che formano la base della teoria emergente. Le ipotesi provvisorie vengono formulate attraverso un processo deduttivo in questa fase. Vengono inventati nuovi codici che raccolgono, riassumono o sostituiscono i codici esistenti. **I codici scelti appartengono alla teoria.**
 - Una volta raggiunta una teoria plausibile, può essere eseguita l'analisi quantitativa dei codici selezionati (ponderazione, conteggio, assegnazione di punteggi, ecc.).

- *Pro*: fornisce una metodologia sistematica, ben esplorata e nota per analizzare quantitativamente una raccolta di dati qualitativi.
- *Contro*: per funzionare, richiede molto lavoro manuale, E introduce una discreta quantità di manipolazione soggettiva dei dati di origine, poiché sia la codifica aperta sia (molto visibilmente) quella assiale introducono la comprensione e le opinioni del ricercatore nella teoria che prende forma.

12 - Prestazioni Umane (Human Performances)

Introduzione alla Lezione

Benvenuti a questa lezione sulle prestazioni umane. Comprendere come gli esseri umani percepiscono, immagazzinano ed elaborano le informazioni è fondamentale per progettare sistemi interattivi efficaci e intuitivi. Esploreremo i meccanismi alla base della percezione sensoriale, della memoria e del ragionamento, e come questi influenzano le nostre interazioni con il mondo e con la tecnologia.

Argomenti della Lezione:

- **Percezione: i Sensi**
 - Vista (Sight)
 - Udito (Hearing)
 - Altri sensi (Tatto, Gusto, Olfatto)
 - Movimento (Propriocezione, Equilibrio)
- **Immagazzinamento (Memoria - Storage/Memory)**
 - Memoria Sensoriale (Sensory memory)
 - Memoria a Breve Termine (Short-term memory)
 - Memoria a Lungo Termine (Long-term memory)
- **Elaborazione (Ragionamento - Processing/Reasoning)**
 - Ragionamento (Reasoning)
 - Apprendimento (Learning)
 - Risoluzione di Problemi (Problem-solving)

- Errori (Errors)

Modello Cognitivo Generale

Il diagramma mostra un modello semplificato della cognizione umana:

- **Input** (stimoli esterni)
- **Percezione** (interpretazione degli input sensoriali)
- **Elaborazione** (ragionamento, presa di decisioni)
- **Memoria** (immagazzinamento e recupero di informazioni)
- **Azione** (risposta motoria o comportamentale)
- **Output** (risultato dell'azione)
- L'**Attenzione** gioca un ruolo centrale, modulando percezione, elaborazione e memoria.

Modello di Apprendimento e Memoria

Questo diagramma illustra il flusso dell'informazione attraverso i sistemi di memoria:

- Un **Evento Esterno** viene catturato dalla **Memoria Sensoriale**.
- Attraverso un processo di **Codifica** (soggetto a freschezza e importanza), l'informazione passa alla **Memoria a Breve Termine**.
- Dalla memoria a breve termine, l'informazione può essere trasferita alla **Memoria a Lungo Termine** attraverso la **Ritenzione** (consolidamento).
- L'informazione può essere recuperata dalla memoria a lungo termine e riportata alla memoria a breve termine attraverso il **Richiamo (Recall)**.

1. Percezione: i Sensi (Più di Cinque!)

- **1. Vista (Eyesight):** Percezione di luce, forme, colori. (Approfondiremo tra poco).
- **2. Udito (Hearing):** Percezione delle onde sonore. (Approfondiremo tra poco).
- **3. Tatto (Touch):**
 - **Termorecettori:** percezione della temperatura.
 - **Nocicettori:** percezione del dolore.
 - **Meccanorecettori:** percezione della pressione.

- Grandi differenze di risoluzione e precisione nelle diverse aree del corpo.
- **4. Gusto (Taste):**
 - Analisi chimica, termica e fisica degli oggetti attraverso sensori posti sulle papille gustative e sulle membrane della bocca.
 - N.B.: La lingua è cinque volte più sensibile alla temperatura e alla pressione rispetto ai polpastrelli.
- **5. Olfatto (Smell):**
 - Analisi chimica e tattile delle particelle sospese nell'aria, eseguita dalle membrane nasali.
- **6. Propriocezione o Cinestesia (Proprioception or kinesthesia):**
 - Consapevolezza della posizione del corpo e delle sue parti rispetto allo spazio esterno e ad altre parti.
 - Esistono sei tipi di recettori specifici in varie parti del corpo, dai muscoli ai tendini alla pelle.
 - (Riferimento: Oliver Sacks, "L'uomo che scambiò sua moglie per un cappello").
- **7. Equilibrio (Balance):**
 - Consapevolezza del centro di gravità del corpo e della velocità e direzione del movimento.
 - Un senso complesso basato su vista, propriocezione e un insieme specifico di recettori nel sistema vestibolare (orecchio).

Focus sulla Vista (Eyesight)

Senza dubbio, la fonte più importante di dati esterni per gli esseri umani.

La percezione visiva avviene in **due stadi**:

1. **Percezione fisica degli stimoli visivi da parte degli occhi.**
2. **Ulteriore elaborazione dei dati da parte del cervello.**

- **Aspetti Fondamentali della Vista Umana:**

- Hardware specializzato per movimenti, colori, luminosità.
- Pre-elaborazione di forme fondamentali negli occhi, post-elaborazione e attribuzione di senso nel cervello.
- La visione stereo fornita da entrambi gli occhi aiuta nel calcolo delle distanze e migliora l'interpretazione dei dati percepiti.
- Molto avviene ancora nell'elaborazione dell'immagine a livello cerebrale.

- **Percezione della Luce (Eyesight 2):**

- La luce viene riflessa dagli oggetti del mondo reale ed entra nell'occhio.
- La **cornea** protegge le parti interne dell'occhio e agisce come una lente a fuoco fisso, inviando la luce al **cristallino (lens)**, che può mettere a fuoco.
- L'**iride (iris)** è un muscolo che controlla la **pupilla** (il foro dell'occhio), regolando così la quantità di luce che entra.
- Il cristallino invia l'immagine (capovolta) sul fondo della **retina**, dove si trovano i **fotorecettori**.
- La **fovea** è il punto esatto di focalizzazione dell'immagine.
- La giunzione tra la retina e il nervo ottico è quasi priva di recettori ed è chiamata **punto cieco (blind spot)**.

- **Fotorecettori (Eyesight 3):** Esistono quattro tipi:

- **Bastoncelli (Rods):** Sparsi su tutta la retina, molto sensibili alla quantità di luce, permettono una discreta visione notturna. Non sono sensibili ai colori e si saturano facilmente (abbagliamento - glare).
- **Coni (Cones):** Principalmente localizzati nella fovea, molto sensibili ai colori. Non si attivano quando la quantità di luce è bassa (limitata percezione dei colori di notte).
- **Cellule Gangliari X (X-Ganglion):** Principalmente nella fovea, dedicate alla pre-identificazione di pattern visivi.
- **Cellule Gangliari W e Y (W- and Y-Ganglion):** Ovunque, ma più dense nella parte esterna della retina, dedicate alla pre-identificazione del movimento. Permettono di percepire il movimento sul fondo dell'occhio molto più velocemente, anche senza riconoscere le forme.

- **Distanza, Profondità, Luminosità (Eyesight 4):**

- *Percezione della distanza*: Non possiamo percepire oggetti più piccoli di 0.5 secondi d'arco. Oggetti più vicini appaiono più grandi di oggetti più lontani. Tuttavia, appaiono costanti anche quando la distanza aumenta (il cervello compensa la perdita di dettagli).
- *Percezione della profondità*: La visione stereoscopica è causata dal confronto delle piccole differenze negli stimoli percepiti dai due occhi. Altri indizi vengono dalla percezione della sovrapposizione e dalla familiarità con gli oggetti percepiti.
- *Percezione della luminosità (brillantezza - brightness)*: La brillantezza è la percezione soggettiva della quantità di luce. È diversa dalla quantità oggettiva di luce emessa da un corpo (luminanza - luminance). Il **contrasto** è la differenza di luminanza tra un oggetto e il suo sfondo. **Flicker**: la percezione di una luce che si accende e si spegne, percepibile fino a 50 Hz, ma a frequenze maggiori con alta luminanza o nella visione periferica.

- **Percezione dei Colori (Eyesight 5):**

- I colori sono compito dei coni, sensibili a tre colori (rosso, verde e blu).
- A causa delle loro densità relative, i colori sono percepiti meglio nella fovea e peggio nella visione periferica.
- I recettori blu sono considerevolmente inferiori in quantità, quindi siamo meno sensibili alle sfumature di blu.
- Gli esseri umani possono distinguere circa 150 diverse tonalità (lunghezza d'onda della luce riflessa), per un totale di circa 7 milioni di colori se consideriamo intensità (luminanza del colore) e saturazione (presenza di bianco nel colore). I colori identificabili separatamente sono solo 10-20. Non c'è differenza nella percezione del colore tra maschi e femmine (in assenza di deficit).
- *Daltonismo (Color blindness)*: Circa l'8% dei maschi e l'1% delle femmine ha una deficienza genetica nella funzionalità dei coni, che rende difficile distinguere i colori. Una mancanza o disfunzione nei coni verdi o rossi fa sì che gli altri compensino, causando confusione. Più rara è una deficienza nei coni blu, e ancora più rara la totale mancanza di funzione in tutti i coni (visione in bianco e nero).

- **Elaborazione dell'Immagine (Eyesight 6):**

- L'elaborazione dell'immagine genera concetti che possono essere interpretati dal cervello. La maggior parte si basa sulla percezione di **pattern** e **aspettative**.

- Questi permettono di fornire stabilità nelle immagini in movimento e quando ci muoviamo rispetto alle immagini.
 - Le **illusioni ottiche** sono solitamente cattive attivazioni di pattern o aspettative. (Es. Illusione di Muller-Lyer, Illusione di Ponzo).
- **Impatto dell'Esperienza sulla Visione:**
 - L'esperienza pregressa e la conoscenza influenzano ciò che percepiamo. Una volta che sappiamo cosa cercare (il cane dalmata che annusa il terreno), diventa più facile vederlo.
- **Aspettativa nella Percezione:**
 - Se ci viene chiesto di cercare delle forbici, la nostra attenzione si concentra su quella forma. Potremmo non notare altri oggetti (come un cacciavite) o non ricordarne i dettagli (colore), perché la nostra aspettativa guida la percezione selettiva.
- **Strutturare la Visione: i Principi della Gestalt:**
 - Inizio XX secolo: il sistema percettivo forma un **percepto (gestalt – o forma in tedesco)** indipendente dalle parti. Secondo Kurt Koffka, "Il tutto è altro dalla somma delle parti".
 - **Prossimità:** Oggetti/eventi più vicini nello spazio o nel tempo vengono percepiti come gruppo.
 - **Somiglianza:** Oggetti/eventi che condividono attributi o proprietà vengono percepiti come gruppo.
 - **Continuazione:** Oggetti organizzati attorno a una curva continua e prevedibile vengono percepiti come un'unità.
 - **Chiusura:** Oggetti/eventi separati formano una figura completa e riconoscibile (il cervello "chiude" le forme).
 - **Semplicità (o Pregnanza):** Oggetti/eventi con forme e strutture che semplificano la loro percezione tendono ad essere visti in quel modo.
 - **Figura/Sfondo:** La messa a fuoco implica l'identificazione di una figura come preminente e il trattamento del resto come sfondo.
 - **Destino Comune:** Oggetti che si comportano in modo simile vengono percepiti come gruppo.

- **Elaborazione del Testo (Eyesight 7):**

- *Tre fasi separate:* Input della percezione visiva del testo, Decodifica della/e parola/e basata sulla lingua di riferimento, Analisi sintattica e semantica del testo.
- *Durante la percezione visiva del testo:* 6% del tempo in movimenti a raffica della pupilla (saccadi), avanti e indietro. 94% del tempo in posizione fissa (tempo di elaborazione). La complessità del testo è proporzionale al numero di regressioni (movimenti all'indietro) dell'occhio.
- *Leggere non significa identificare singole lettere o parole:* Il tempo di riconoscimento per singole lettere, parole intere o frasi semplici è identico. La forma delle parole influisce direttamente sulla velocità di lettura: font non familiari, testo in maiuscolo e lingue sconosciute rallentano notevolmente la lettura.
- *Aiuti e ostacoli alla lettura:* Gli adulti alfabetizzati leggono circa 5.5 sillabe al secondo. Dimensioni dei font tra 9 e 12 punti sono ugualmente leggibili (più lento se più grandi o più piccoli). Larghezze di riga tra 6 e 14 cm sono ugualmente percepibili senza differenze di velocità. **Leggere su uno schermo di computer È più lento** (righe più lunghe, meno parole per schermo rispetto a una pagina, orientamento del testo). Il contrasto negativo (testo scuro su sfondo chiaro) riduce la luminanza e quindi ha un contrasto maggiore rispetto al contrasto positivo, ma è più incline al flickering.

- **Progettare per la Lettura:**

- Evitare termini non comuni o non familiari.
- Evitare font decorativi o non comuni, font piccoli, sfondi rumorosi.
- Organizzare il testo in blocchi inferiori a 14 cm, in modo che la testa non debba muoversi per leggerli.
- Organizzare numeri in gruppi, date in blocchi, testi in strutture gerarchiche che permettano di percepire la struttura generale prima del contenuto effettivo.
- Evitare testo centrato (difficile da leggere per lunghi periodi).
- Evitare testo ridondante.
- Minimizzare la necessità di leggere testi lunghi.
- Il testo in maiuscolo NON è appropriato per letture lunghe. Usarlo solo nei titoli.

L'Occhio Umano e la Percezione Visiva

L'**occhio** è senza dubbio la fonte più importante di dati esterni per gli esseri umani. La percezione visiva si svolge in due fasi principali: una **percezione fisica degli stimoli visivi** da parte degli occhi e una successiva **elaborazione dei dati da parte del cervello**.

La luce, riflessa dagli oggetti del mondo reale, entra nell'occhio. Il primo elemento che incontra è la **cornea**, che protegge le parti interne dell'occhio da agenti esterni come aria e polvere, e agisce come una lente a fuoco fisso, dirigendo la luce verso il cristallino. L'**iride** (iride) è un muscolo che controlla la **pupilla** (il foro dell'occhio), regolando così la quantità di luce che entra nell'occhio. Il **cristallino** è in grado di mettere a fuoco l'immagine, inviandola capovolta sul fondo della **retina**, dove si trovano i fotorecettori. La **fovea** è il punto esatto di focalizzazione dell'immagine sulla retina. È interessante notare che la giunzione tra la retina e il nervo ottico è quasi priva di recettori ed è chiamata **punto cieco**.

La nostra visione è anche caratterizzata da **hardware specializzato** per il movimento, i colori e la luminosità. La **pre-elaborazione di forme fondamentali** avviene negli occhi, mentre l'elaborazione successiva e l'attribuzione di significato avvengono nel cervello. La **visione stereoscopica**, fornita da entrambi gli occhi, è cruciale per calcolare le distanze e migliorare l'interpretazione dei dati percepiti, contribuendo alla **percezione della profondità**.

I Fotorecettori

All'interno della retina, troviamo **quattro tipi di fotorecettori**, essenziali per la nostra visione:

- **Bastoncelli (Rods):**
 - Sono diffusi su tutta la retina e sono **molto sensibili alla quantità di luce**.
 - Permettono una visione notturna abbastanza buona.
 - **Non sono sensibili ai colori**.
 - Sono facili da saturare, causando il fenomeno dell'**abbagliamento** (glare).
- **Coni (Cones):**
 - Sono posizionati principalmente nella **fovea**.
 - Sono **molto sensibili ai colori**.
 - Non si attivano quando la quantità di luce è bassa, motivo per cui abbiamo una **percezione limitata dei colori di notte**.
 - Sono responsabili della percezione dei colori sensibili a tre colori primari: **rosso, verde e blu**.
 - A causa della loro densità, i colori sono percepiti al meglio nella fovea e peggio nella visione periferica. I recettori per il blu sono meno numerosi, rendendoci meno sensibili alle sfumature di blu.
 - Gli esseri umani possono distinguere circa 150 diverse tonalità (la lunghezza d'onda della luce riflessa), per un totale di circa 7 milioni di colori quando si considerano l'intensità (luminanza del colore) e la saturazione (presenza di bianco nel colore). I colori separatamente identificabili sono solo 10-20. Non c'è differenza nella percezione dei colori tra maschi e femmine.
 - La condizione di **daltonismo** (color blindness) si verifica in circa l'8% dei maschi e

l'1% delle femmine a causa di una deficienza genetica nella funzionalità dei coni, rendendo difficile distinguere i colori. Una mancanza o disfunzione nei coni verdi o rossi causa confusione tra questi colori. Più rari sono la deficienza nei coni blu o la totale mancanza di funzione in tutti i coni (daltonismo in bianco e nero).

- **Ganglion X (X-Ganglion):**
 - Si trovano principalmente nella **fovea**.
 - Sono dedicati alla **pre-identificazione di modelli visivi**.
- **Ganglion W e Y (W- and Y-Ganglion):**
 - Sono distribuiti ovunque, con una maggiore densità nella parte esterna della retina.
 - Sono dedicati alla **pre-identificazione del movimento**.
 - Questo permette di percepire il movimento nella parte posteriore dell'occhio molto più velocemente, anche senza riconoscere le forme.

Parole Chiave e Concetti Fondamentali

- **Percezione Visiva Umana:** Il processo attraverso cui gli esseri umani interpretano gli stimoli visivi.
- **Occhio:** L'organo sensoriale responsabile della vista.
- **Cervello:** La parte del sistema nervoso centrale che elabora i dati visivi e dà loro significato.
- **Visione Stereoscopica:** La capacità di percepire la profondità grazie al confronto delle leggere differenze tra le immagini percepite dai due occhi.
- **Cornea:** La parte trasparente anteriore dell'occhio che protegge e funge da lente a fuoco fisso.
- **Iride:** Il muscolo che controlla la dimensione della pupilla.
- **Pupilla:** L'apertura centrale dell'occhio che regola la quantità di luce in ingresso.
- **Cristallino:** La lente dell'occhio che mette a fuoco l'immagine sulla retina.
- **Retina:** Il tessuto sensibile alla luce sul fondo dell'occhio, contenente i fotorecettori.
- **Fovea:** Il punto di massima acuità visiva sulla retina, ricco di coni.
- **Punto Cieco:** Un'area della retina priva di fotorecettori, dove il nervo ottico lascia l'occhio.
- **Bastoncelli:** Fotorecettori sensibili alla luce ma non ai colori, importanti per la visione notturna.
- **Coni:** Fotorecettori sensibili ai colori, principalmente nella fovea.
- **Ganglion X, W, Y:** Cellule gangliari coinvolte nella pre-identificazione di schemi e

movimenti.

- **Distanza:** La percezione della lontananza degli oggetti.
- **Profondità:** L'impressione tridimensionale derivante dal confronto delle immagini dei due occhi e altri indizi.
- **Luminosità** (brillantezza): La percezione soggettiva della quantità di luce.
- **Luminanza:** La quantità oggettiva di luce emessa da un corpo.
- **Contrasto:** La differenza di luminanza tra un oggetto e il suo sfondo.
- **Sfarfallio** (Flicker): La percezione di una luce che si accende e si spegne.
- **Colore:** La percezione della lunghezza d'onda della luce, mediata dai coni.
- **Saturazione:** La purezza o l'intensità di un colore (presenza di bianco).
- **Daltonismo:** Una deficienza genetica nella percezione dei colori.
- **Elaborazione dell'immagine:** Il processo mentale di generazione di concetti interpretabili dal cervello, basato su modelli e aspettative.
- **Illusione Ottica:** Fenomeni visivi che derivano da attivazioni errate di modelli o aspettative.
- **Aspettativa:** L'influenza delle nostre conoscenze pregresse su ciò che percepiamo visivamente.
- **Principi della Gestalt:** Regole di organizzazione percettiva che guidano il modo in cui il cervello raggruppa e interpreta gli stimoli visivi, come:
 - **Prossimità:** Oggetti o eventi vicini sono percepiti come un gruppo.
 - **Similarità:** Oggetti o eventi che condividono attributi sono raggruppati.
 - **Continuazione:** Gli oggetti sono organizzati lungo una curva continua e prevedibile.
 - **Chiusura:** Oggetti o eventi separati formano una figura completa e riconoscibile.
 - **Semplicità:** Gli oggetti o eventi hanno forme e strutture che ne semplificano la percezione.
 - **Figura/Sfondo:** La capacità di identificare una figura come prominente e trattare il resto come sfondo.
 - **Destino Comune:** Oggetti che si comportano in modo simile sono percepiti come aventi un destino comune.

Focus sull'Udito (Hearing)

- Apparentemente di minore importanza rispetto alla vista, fornisce un'enorme quantità di informazioni sul nostro ambiente.
- L'orecchio riceve vibrazioni dell'aria che vengono raccolte e amplificate dall'**auricola (padiglione auricolare)**, colpendo il **timpano (eardrum)**. Questo muove tre ossicini: **martello (malleus)**, **incudine (incus)** e **staffa (stape)**, che a loro volta muovono una sostanza gelatinosa nella **coclea (cochlea o orecchio interno)**. Ciò attiva minuscole cellule ciliate che trasmettono informazioni al nervo acustico e al cervello.
- La **tromba di Eustachio** serve a regolare la pressione interna ed esterna dell'orecchio.
- **Caratteristiche dei Suoni:**
 - **Altezza (Pitch)**: frequenza percepita del suono.
 - **Intensità (Loudness)**: pressione sonora percepita.
 - **Timbro (Timbre)**: qualità sonora percepita.
- L'orecchio umano può percepire frequenze tra 20Hz e 15000 Hz. Le frequenze più basse vengono percepite con le ossa, non con l'orecchio. La risoluzione cambia a seconda delle frequenze.
- Possiamo percepire la direzione e il movimento del suono attraverso la percezione stereofonica delle due orecchie.
- L'udito si basa fondamentalmente su meccanismi di filtraggio che permettono di isolare parti del percepito dal rumore circostante (**effetto cocktail party**).

2. Immagazzinamento: la Memoria (Storage)

Negli anni Sessanta fu proposta una teoria specifica per la memoria basata su un modello stratificato:

- **Memoria Sensoriale (Sensory memory)**
- **Memoria a Breve Termine o di Lavoro (Short-term or Working memory)**
- **Memoria a Lungo Termine o Permanente (Long-term or permanent memory)**

Non sappiamo se siano separate, né come interagiscano. Le usiamo come un modello con una chiara mappatura all'hardware del computer più che come una guida al funzionamento effettivo della mente umana.
- **Memoria Sensoriale (Sensory Memory):**

- *Specificità*: Memoria iconica per stimoli visivi, Memoria ecoica per stimoli uditivi (usata per confronto stereofonico, ma anche per ritenzione dello stimolo), Memoria tattile per il tatto, ecc.
- *Persistenza*: Lo stimolo è mantenuto per pochi decimi di secondo (circa 0.5 sec per la vista) e viene continuamente riscritto. Solo una quantità minima di dati viene mantenuta, la maggior parte viene persa immediatamente o durante l'elaborazione dei dati.
- *Attenzione*: È il meccanismo fondamentale per filtrare i dati non necessari.
- *Continuità*: La continuità della percezione è fondamentale per la consapevolezza del flusso temporale e la connessione con la realtà. Interruzioni o interferenze generano déjà-vu (paramnesia).
- *Progettare per la memoria sensoriale*: Ridurre il carico mentale necessario per interpretare lo stimolo sensoriale (pochi elementi semplici, ben differenziati, solidamente posizionati nell'interfaccia). Usare sistematicamente la teoria della Gestalt per aiutare la strutturazione del percepito (permettere raggruppamenti, rendere evidenti le differenze significative, nascondere quelle non significative).
-
- **Memoria a Breve Termine (Short Term Memory - STM)** (o Memoria di Lavoro):
 - Qui vengono mantenuti ed elaborati i dati rilevanti per i compiti attualmente in esecuzione.
 - Utilizza circa **7 ± 2 "chunk"** o blocchi (possibilmente strutturati) di informazione. Questi chunk sono sensorialmente modali e permettono il raggruppamento (es. 52147651210 vs 055 456 712 65).
 - La STM è **sempre piena**. Se un nuovo chunk di informazione viene immagazzinato, è solo sostituendo uno precedente: qualcosa deve essere rimosso.
 - La persistenza è di circa **15 secondi**. Può essere aumentata con la ripetizione uditiva (rehearsal).
 - La ritenzione è amplificata dall'**importanza** (prima) e dalla **freschezza** (poi).
 - *Progettare per la memoria a breve termine*: Ridurre il carico mentale necessario per mantenere il contesto del dialogo e la coerenza. Avere meno chunk (complessi se necessario) da tenere a mente (7±2 elementi distinti). Aiutare a recuperare chunk scomparsi dal contesto visivo. Aiutare a trattenere una maggiore quantità di informazioni attraverso il raggruppamento.
- **Memoria a Lungo Termine (Long Term Memory - LTM)**:

- Organizzata in:
 - **Memoria Episodica:** registrazione di eventi ed esperienze in modo seriale.
 - **Memoria Semantica:** registrazione di fatti, concetti e abilità apprese nel passato.
- La LTM non è mai una registrazione completa del percolato, ma una post-elaborazione delle parti rilevanti con forti filtri.
- *Memoria Eidetica (o fotografica):* Il ricordo con un gran numero di dettagli (ricordare il percolato piuttosto che la sua elaborazione). Avviene principalmente nei bambini in età prescolare, estremamente rara negli adulti. (Cfr. "Funes o della memoria" di J.L. Borges). Tranne in tali casi patologici, il cervello adulto ha imparato a non trattenere mai più dettagli del percolato di una sua forma estremamente semplificata.
- La ritenzione nella LTM è ottenuta attraverso l'elaborazione; non conosciamo né limiti né durata dell'immagazzinamento.
- L'organizzazione è gerarchica e associativa: molto probabilmente può essere mappata come una rete semantica che collega concetti in categorie e sottocategorie.
- Concetti disconnessi (senza connessioni logiche ad altri fatti della nostra conoscenza) sono più difficili da trattenere rispetto a quelli connessi.
- Concetti astratti sono più difficili da trattenere rispetto a concetti concreti.
- Concetti con un livello più alto di connessione emotiva (non necessariamente personale) sono più facili da trattenere rispetto a quelli "aridi".

La categorizzazione non è neutra e non si basa solo sulla motivazione semantica. La coloritura temporale ed emotiva gioca un ruolo importante.

- *Interferenza:* Accedere a un ricordo può causare accesso anche a un altro completamente disconnesso da un punto di vista logico.
- *Dimenticare (Due teorie):*
 - **Decadimento (Decay):** ricordi non frequentemente accessi decadono lentamente ma naturalmente e scompaiono – connesso all'"effetto punta della lingua".
 - **Interferenza:** la ritenzione di nuove informazioni sostituisce naturalmente e immediatamente informazioni simili più vecchie (es. codici ATM).
-
- La memoria funziona meglio nel **riconoscere** che nel **richiamare**. Recuperare una nota scritta con il vecchio codice ti farà immediatamente riconoscere il numero e il suo significato.

- *Progettare per la memoria a lungo termine*: Non fare mai affidamento sulla semplice memoria dei fatti. Cercare di trasmettere una narrazione nella sequenza dei passaggi (alternativamente, la memoria iconica o visiva è migliore di quella episodica o puntuale). Permettere una facile astrazione per consentire la ritenzione di chunk più semplici. Usare verbi per le azioni e nomi per i concetti. Creare una narrazione per l'interazione.
- **Riflessioni Aggiuntive sulla Memoria e il Design:**
 - Guardare e scegliere è più facile che ricordare e digitare.
 - Le immagini sono più facili da riconoscere delle parole (le miniature sono più utili delle descrizioni testuali per le immagini).
 - La visibilità deve essere proporzionale all'importanza e alla frequenza d'uso della funzione.
 - Usare aiuti visivi per ricordare all'utente dove si trova.
 - Rendere i dati di autenticazione facili da ricordare

3. Attenzione (Attention)

- L'attenzione è la **selezione di uno o pochi stimoli sensoriali** che ci raggiungono.
- Può essere **focalizzata** (scrivere) o **divisa** (guidare parlando o ascoltando la radio).
- **Effetto cocktail party**: la nostra abilità di filtrare (escludere) la maggior parte degli stimoli anche da un solo senso.
- Il compito decide su cosa concentrarsi. Le aspettative hanno un impatto sull'attenzione.
- L'attenzione è **modale**: eventi inaspettati o fatti fuori modalità sono fonte di distrazione.
- Sono necessari aiuti cognitivi per mantenere l'attenzione.
- **Progettare per l'Attenzione:**
 - Ridurre il carico cognitivo (anche se questo aumenta la quantità di azioni atomiche da eseguire).
 - Guidare gli utenti a scegliere percorsi d'azione più veloci o più facili.
 - Mostrare chiaramente lo stato del sistema e il livello di progresso nel completamento di un compito.
 - Rendere il sistema familiare.
 - Lasciare che il computer faccia i calcoli (es. totali).
 - Usare termini familiari.

●

4. Elaborazione: Ragionamento (Processing)

- Il ragionamento è come usiamo la conoscenza che già abbiamo per generare conclusioni sull'evento su cui ci stiamo concentrando. Usiamo il ragionamento per generare nuove informazioni e risolvere problemi nella nostra vita quotidiana.
- Questo avviene in modo semi-conscio: a volte raggiungiamo un risultato o una soluzione senza essere realmente consapevoli del processo seguito per arrivarci.
- Almeno tre tipi diversi di ragionamento:
 - **Ragionamento Deduttivo:**
 - Partendo da un'assunzione generale e un caso concreto, otteniamo una conclusione precisa.
 - Il **sillogismo** è lo strumento più famoso. (Es. Tutti gli uomini che respirano sono vivi; John respira; QUINDI John è vivo).
 - Non siamo sempre bravi con il ragionamento deduttivo, specialmente in presenza di assunzioni false o parziali.
 - **Ragionamento Induttivo:**
 - Partendo da molti casi omogenei, produco (induco) una regola generale (inferenza o generalizzazione).
 - (Es. Tutti gli elefanti che ho visto hanno la proboscide; QUINDI Tutti gli elefanti hanno la proboscide).
 - L'inferenza è inaffidabile e facile da smentire (basta mostrare un elefante senza proboscide). Inoltre, non può essere completamente provata a meno che non si possano esaminare sistematicamente tutti i casi.
 - Tuttavia, è il metodo usuale per generare nuove regole nella nostra vita quotidiana e nella scienza.
 - **Ragionamento Abduttivo:**
 - Dato un caso (a cui si possono applicare molte regole), scegliamo la regola migliore che si applica al caso.
 - (Es. John sta guidando troppo veloce; John guida veloce quando è ubriaco; QUINDI John è ubriaco).
 - Anche questo meccanismo è imperfetto: molte regole potrebbero applicarsi e potrebbero esserci regole sconosciute. (John potrebbe avere un'emergenza).
 - Inoltre, un'assunzione ingiustificata con l'abduzione è che esista una regola per spiegare il caso, e dobbiamo solo identificarla.
 - Nei sistemi informatici, spesso assumiamo che ciò che accade sullo schermo derivi dall'azione che abbiamo appena eseguito; quindi, se i due fatti sono disconnessi, questo crea confusione ed errore.

-
-

Risoluzione di Problemi (Problem Solving)

- Trovare una soluzione a un problema nuovo o non familiare. Gli esseri umani sono in grado di adattare qualsiasi conoscenza che hanno a nuove situazioni.
- **Comportamentismo (Behaviorism)** (fine XIX secolo): la risoluzione dei problemi si basa sull'applicazione di regole esistenti o per tentativi ed errori, esplorando possibili soluzioni finché non se ne trova una buona.
- **Teoria della Gestalt** (metà XX secolo): la risoluzione dei problemi si basa sull'applicazione di regole esistenti (approcci riproduttivi) così come sulla riflessione e ristrutturazione del problema in termini diversi (approcci produttivi). (Es. problema delle due corde appese al soffitto).
- **Teoria dello Spazio del Problema (Problem space theory)** (anni '70): il problema è espresso come la ricerca di un percorso all'interno di uno spazio dove c'è uno stato iniziale (il problema) e uno stato finale (la soluzione). Vengono identificati possibili luoghi intermedi desiderabili, e la risoluzione del problema è espressa come trovare un percorso verso un luogo intermedio più vicino, e poi l'analisi ricomincia dalla posizione più vicina. Ogni sottoproblema viene analizzato cercando euristiche, competenze e analogie.

Prestazioni degli Esseri Umani (Performances of Human Beings)

- I **tempi di risposta** dipendono fortemente dal tipo di stimolo (in particolare, rispondiamo molto più velocemente al suono che allo stimolo visivo).
- Esistono **compromessi tra velocità e accuratezza**.
- Le risposte automatiche diminuiscono in accuratezza.
- Le risposte ripetute diminuiscono in accuratezza e velocità.
- La **stanchezza** diminuisce velocità e accuratezza.
- Gli **errori di azione (imprecisione)** impattano sulle prestazioni tanto quanto le decisioni sbagliate.
- **Legge di Fitts (Fitt's Law)**: Il tempo necessario per raggiungere un bersaglio sullo schermo con un dispositivo di puntamento è proporzionale alla distanza e inversamente proporzionale alla dimensione del bersaglio.
 - $T = a + b \log_2(2D/W)$
 - dove: T = tempo per il bersaglio, a, b = costanti da determinare, D = distanza del bersaglio, W = dimensione del bersaglio.

Conclusioni

- Limiti e vincoli degli esseri umani come "dispositivi".
- Caratteristiche dei canali di I/O degli esseri umani.
- Tipi di meccanismi di memoria ed elaborazione umana.
- Prestazioni degli esseri umani.

Comprendere questi aspetti è essenziale per progettare interfacce uomo-macchina che siano non solo funzionali ma anche efficienti, efficaci e soddisfacenti per l'utente.

Riferimenti Bibliografici:

- Dix, A. et al. (1998). *HCI*. Prentice Hall, capitolo 1.
- Johnson, J. (2010). *Designing with the mind in mind*. Morgan Kaufmann.
- Egan, D. (1988). "Individual differences in HCI". In M. Helander (Ed.), *Handbook of HCI*. North-Holland.

PRESTAZIONI UMANE (HUMAN PERFORMANCES)

Prof. Fabio Vitali, Alma Mater Studiorum – Università di Bologna

Introduzione alla Lezione

Benvenuti a questa lezione sulle prestazioni umane. Comprendere come gli esseri umani percepiscono, immagazzinano ed elaborano le informazioni è fondamentale per progettare sistemi interattivi efficaci e intuitivi. Esploreremo i meccanismi alla base della percezione sensoriale, della memoria e del ragionamento, e come questi influenzano le nostre interazioni con il mondo e con la tecnologia.

Argomenti della Lezione:

- **Percezione: i Sensi**
 - Vista (Sight)
 - Udito (Hearing)
 - Altri sensi (Tatto, Gusto, Olfatto)
 - Movimento (Propriocezione, Equilibrio)

- **Immagazzinamento (Memoria - Storage/Memory)**
 - Memoria Sensoriale (Sensory memory)
 - Memoria a Breve Termine (Short-term memory)
 - Memoria a Lungo Termine (Long-term memory)
- **Elaborazione (Ragionamento - Processing/Reasoning)**
 - Ragionamento (Reasoning)
 - Apprendimento (Learning)
 - Risoluzione di Problemi (Problem-solving)
 - Errori (Errors)

Modello Cognitivo Generale

Il diagramma mostra un modello semplificato della cognizione umana:

- **Input** (stimoli esterni)
- **Percezione** (interpretazione degli input sensoriali)
- **Elaborazione** (ragionamento, presa di decisioni)
- **Memoria** (immagazzinamento e recupero di informazioni)
- **Azione** (risposta motoria o comportamentale)
- **Output** (risultato dell'azione)
- L'**Attenzione** gioca un ruolo centrale, modulando percezione, elaborazione e memoria.

Modello di Apprendimento e Memoria

Questo diagramma illustra il flusso dell'informazione attraverso i sistemi di memoria:

- Un **Evento Esterno** viene catturato dalla **Memoria Sensoriale**.
- Attraverso un processo di **Codifica** (soggetto a freschezza e importanza), l'informazione passa alla **Memoria a Breve Termine**.
- Dalla memoria a breve termine, l'informazione può essere trasferita alla **Memoria a Lungo Termine** attraverso la **Ritenzione** (consolidamento).
- L'informazione può essere recuperata dalla memoria a lungo termine e riportata alla memoria a breve termine attraverso il **Richiamo (Recall)**.

1. Percezione: i Sensi (Più di Cinque!)

- **1. Vista (Eyesight):** Percezione di luce, forme, colori. (Approfondiremo tra poco).
- **2. Udito (Hearing):** Percezione delle onde sonore. (Approfondiremo tra poco).
- **3. Tatto (Touch):**

- **Termorecettori:** percezione della temperatura.
- **Nocicettori:** percezione del dolore.
- **Meccanorecettori:** percezione della pressione.
- Grandi differenze di risoluzione e precisione nelle diverse aree del corpo.
- **4. Gusto (Taste):**
 - Analisi chimica, termica e fisica degli oggetti attraverso sensori posti sulle papille gustative e sulle membrane della bocca.
 - N.B.: La lingua è cinque volte più sensibile alla temperatura e alla pressione rispetto ai polpastrelli.
- **5. Olfatto (Smell):**
 - Analisi chimica e tattile delle particelle sospese nell'aria, eseguita dalle membrane nasali.
- **6. Propriocezione o Cinestesia (Proprioception or kinesthesia):**
 - Consapevolezza della posizione del corpo e delle sue parti rispetto allo spazio esterno e ad altre parti.
 - Esistono sei tipi di recettori specifici in varie parti del corpo, dai muscoli ai tendini alla pelle.
 - (Riferimento: Oliver Sacks, "L'uomo che scambiò sua moglie per un cappello").
-
- **7. Equilibrio (Balance):**
 - Consapevolezza del centro di gravità del corpo e della velocità e direzione del movimento.
 - Un senso complesso basato su vista, propriocezione e un insieme specifico di recettori nel sistema vestibolare (orecchio).
-

Focus sulla Vista (Eyesight)

Senza dubbio, la fonte più importante di dati esterni per gli esseri umani.

La percezione visiva avviene in **due stadi**:

1. **Percezione fisica degli stimoli visivi da parte degli occhi.**
 2. **Ulteriore elaborazione dei dati da parte del cervello.**
- **Aspetti Fondamentali della Vista Umana:**
 - Hardware specializzato per movimenti, colori, luminosità.
 - Pre-elaborazione di forme fondamentali negli occhi, post-elaborazione e attribuzione di senso nel cervello.

- La visione stereo fornita da entrambi gli occhi aiuta nel calcolo delle distanze e migliora l'interpretazione dei dati percepiti.
- Molto avviene ancora nell'elaborazione dell'immagine a livello cerebrale.
- **Percezione della Luce (Eyesight 2):**
 - La luce viene riflessa dagli oggetti del mondo reale ed entra nell'occhio.
 - La **cornea** protegge le parti interne dell'occhio e agisce come una lente a fuoco fisso, inviando la luce al **cristallino (lens)**, che può mettere a fuoco.
 - L'**iride (iris)** è un muscolo che controlla la **pupilla** (il foro dell'occhio), regolando così la quantità di luce che entra.
 - Il cristallino invia l'immagine (capovolta) sul fondo della **retina**, dove si trovano i **fotorecettori**.
 - La **fovea** è il punto esatto di focalizzazione dell'immagine.
 - La giunzione tra la retina e il nervo ottico è quasi priva di recettori ed è chiamata **punto cieco (blind spot)**.
-
- **Fotorecettori (Eyesight 3):** Esistono quattro tipi:
 - **Bastoncelli (Rods):** Sparsi su tutta la retina, molto sensibili alla quantità di luce, permettono una discreta visione notturna. Non sono sensibili ai colori e si saturano facilmente (abbagliamento - glare).
 - **Coni (Cones):** Principalmente localizzati nella fovea, molto sensibili ai colori. Non si attivano quando la quantità di luce è bassa (limitata percezione dei colori di notte).
 - **Cellule Gangliari X (X-Ganglion):** Principalmente nella fovea, dedicate alla pre-identificazione di pattern visivi.
 - **Cellule Gangliari W e Y (W- and Y-Ganglion):** Ovunque, ma più dense nella parte esterna della retina, dedicate alla pre-identificazione del movimento. Permettono di percepire il movimento sul fondo dell'occhio molto più velocemente, anche senza riconoscere le forme.
- **Distanza, Profondità, Luminosità (Eyesight 4):**
 - *Percezione della distanza:* Non possiamo percepire oggetti più piccoli di 0.5 secondi d'arco. Oggetti più vicini appaiono più grandi di oggetti più lontani. Tuttavia, appaiono costanti anche quando la distanza aumenta (il cervello compensa la perdita di dettagli).
 - *Percezione della profondità:* La visione stereoscopica è causata dal confronto delle piccole differenze negli stimoli percepiti dai due occhi. Altri indizi vengono dalla percezione della sovrapposizione e dalla familiarità con gli oggetti percepiti.
 - *Percezione della luminosità (brillantezza - brightness):* La brillantezza è la percezione soggettiva della quantità di luce. È diversa dalla quantità oggettiva di luce

emessa da un corpo (luminanza - luminance). Il **contrasto** è la differenza di luminanza tra un oggetto e il suo sfondo. **Flicker**: la percezione di una luce che si accende e si spegne, percepibile fino a 50 Hz, ma a frequenze maggiori con alta luminanza o nella visione periferica.

- **Percezione dei Colori (Eyesight 5):**

- I colori sono compito dei coni, sensibili a tre colori (rosso, verde e blu).
- A causa delle loro densità relative, i colori sono percepiti meglio nella fovea e peggio nella visione periferica.
- I recettori blu sono considerevolmente inferiori in quantità, quindi siamo meno sensibili alle sfumature di blu.
- Gli esseri umani possono distinguere circa 150 diverse tonalità (lunghezza d'onda della luce riflessa), per un totale di circa 7 milioni di colori se consideriamo intensità (luminanza del colore) e saturazione (presenza di bianco nel colore). I colori identificabili separatamente sono solo 10-20. Non c'è differenza nella percezione del colore tra maschi e femmine (in assenza di deficit).
- *Daltonismo (Color blindness)*: Circa l'8% dei maschi e l'1% delle femmine ha una deficienza genetica nella funzionalità dei coni, che rende difficile distinguere i colori. Una mancanza o disfunzione nei coni verdi o rossi fa sì che gli altri compensino, causando confusione. Più rara è una deficienza nei coni blu, e ancora più rara la totale mancanza di funzione in tutti i coni (visione in bianco e nero).

- **Elaborazione dell'Immagine (Eyesight 6):**

- L'elaborazione dell'immagine genera concetti che possono essere interpretati dal cervello. La maggior parte si basa sulla percezione di **pattern** e **aspettative**.
- Questi permettono di fornire stabilità nelle immagini in movimento e quando ci muoviamo rispetto alle immagini.
- Le **illusioni ottiche** sono solitamente cattive attivazioni di pattern o aspettative. (Es. Illusione di Muller-Lyer, Illusione di Ponzo).

- **Impatto dell'Esperienza sulla Visione:**

L'esperienza pregressa e la conoscenza influenzano ciò che percepiamo. Una volta che sappiamo cosa cercare (il cane dalmata che annusa il terreno), diventa più facile vederlo.

- **Aspettativa nella Percezione:**

- Se ci viene chiesto di cercare delle forbici, la nostra attenzione si concentra su quella forma. Potremmo non notare altri oggetti (come un cacciavite) o non ricordarne i dettagli (colore), perché la nostra aspettativa guida la percezione selettiva.

- **Strutturare la Visione: i Principi della Gestalt:**

- Inizio XX secolo: il sistema percettivo forma un **percepto (gestalt – o forma in tedesco)** indipendente dalle parti. Secondo Kurt Koffka, "Il tutto è altro dalla somma delle parti".
- **Prossimità:** Oggetti/eventi più vicini nello spazio o nel tempo vengono percepiti come gruppo.
- **Somiglianza:** Oggetti/eventi che condividono attributi o proprietà vengono percepiti come gruppo.
- **Continuazione:** Oggetti organizzati attorno a una curva continua e prevedibile vengono percepiti come un'unità.
- **Chiusura:** Oggetti/eventi separati formano una figura completa e riconoscibile (il cervello "chiude" le forme).
- **Semplicità (o Pregnanza):** Oggetti/eventi con forme e strutture che semplificano la loro percezione tendono ad essere visti in quel modo.
- **Figura/Sfondo:** La messa a fuoco implica l'identificazione di una figura come preminente e il trattamento del resto come sfondo.
- **Destino Comune:** Oggetti che si comportano in modo simile vengono percepiti come gruppo.

- **Elaborazione del Testo (Eyesight 7):**

- *Tre fasi separate:* Input della percezione visiva del testo, Decodifica della/e parola/e basata sulla lingua di riferimento, Analisi sintattica e semantica del testo.
- *Durante la percezione visiva del testo:* 6% del tempo in movimenti a raffica della pupilla (saccadi), avanti e indietro. 94% del tempo in posizione fissa (tempo di elaborazione). La complessità del testo è proporzionale al numero di regressioni (movimenti all'indietro) dell'occhio.
- *Leggere non significa identificare singole lettere o parole:* Il tempo di riconoscimento per singole lettere, parole intere o frasi semplici è identico. La forma delle parole

influisce direttamente sulla velocità di lettura: font non familiari, testo in maiuscolo e lingue sconosciute rallentano notevolmente la lettura.

- *Aiuti e ostacoli alla lettura:* Gli adulti alfabetizzati leggono circa 5.5 sillabe al secondo. Dimensioni dei font tra 9 e 12 punti sono ugualmente leggibili (più lento se più grandi o più piccoli). Larghezze di riga tra 6 e 14 cm sono ugualmente percepibili senza differenze di velocità. **Leggere su uno schermo di computer È più lento** (righe più lunghe, meno parole per schermo rispetto a una pagina, orientamento del testo). Il contrasto negativo (testo scuro su sfondo chiaro) riduce la luminanza e quindi ha un contrasto maggiore rispetto al contrasto positivo, ma è più incline al flickering.

- **Progettare per la Lettura:**

- Evitare termini non comuni o non familiari.
- Evitare font decorativi o non comuni, font piccoli, sfondi rumorosi.
- Organizzare il testo in blocchi inferiori a 14 cm, in modo che la testa non debba muoversi per leggerli.
- Organizzare numeri in gruppi, date in blocchi, testi in strutture gerarchiche che permettano di percepire la struttura generale prima del contenuto effettivo.
- Evitare testo centrato (difficile da leggere per lunghi periodi).
- Evitare testo ridondante.
- Minimizzare la necessità di leggere testi lunghi.
- Il testo in maiuscolo NON è appropriato per letture lunghe. Usarlo solo nei titoli.

-

Focus sull'Udito (Hearing)

- Apparentemente di minore importanza rispetto alla vista, fornisce un'enorme quantità di informazioni sul nostro ambiente.
- L'orecchio riceve vibrazioni dell'aria che vengono raccolte e amplificate dall'**auricola (padiglione auricolare)**, colpendo il **timpano (eardrum)**. Questo muove tre ossicini: **martello (malleus)**, **incudine (incus)** e **staffa (stape)**, che a loro volta muovono una sostanza gelatinosa nella **coclea (cochlea o orecchio interno)**. Ciò attiva minuscole cellule ciliate che trasmettono informazioni al nervo acustico e al cervello.
- La **tromba di Eustachio** serve a regolare la pressione interna ed esterna dell'orecchio.

- **Caratteristiche dei Suoni:**
 - **Altezza (Pitch):** frequenza percepita del suono.
 - **Intensità (Loudness):** pressione sonora percepita.
 - **Timbro (Timbre):** qualità sonora percepita.
-
- L'orecchio umano può percepire frequenze tra 20Hz e 15000 Hz. Le frequenze più basse vengono percepite con le ossa, non con l'orecchio. La risoluzione cambia a seconda delle frequenze.
- Possiamo percepire la direzione e il movimento del suono attraverso la percezione stereofonica delle due orecchie.
- L'udito si basa fondamentalmente su meccanismi di filtraggio che permettono di isolare parti del percepito dal rumore circostante (**effetto cocktail party**).

2. Immagazzinamento: la Memoria (Storage)

Negli anni Sessanta fu proposta una teoria specifica per la memoria basata su un modello stratificato:

- **Memoria Sensoriale (Sensory memory)**
 - **Memoria a Breve Termine o di Lavoro (Short-term or Working memory)**
 - **Memoria a Lungo Termine o Permanente (Long-term or permanent memory)**
- Non sappiamo se siano separate, né come interagiscano. Le usiamo come un modello con una chiara mappatura all'hardware del computer più che come una guida al funzionamento effettivo della mente umana.
- **Memoria Sensoriale (Sensory Memory):**
 - *Specificità:* Memoria iconica per stimoli visivi, Memoria ecoica per stimoli uditivi (usata per confronto stereofonico, ma anche per ritenzione dello stimolo), Memoria tattile per il tatto, ecc.
 - *Persistenza:* Lo stimolo è mantenuto per pochi decimi di secondo (circa 0.5 sec per la vista) e viene continuamente riscritto. Solo una quantità minima di dati viene mantenuta, la maggior parte viene persa immediatamente o durante l'elaborazione dei dati.
 - *Attenzione:* È il meccanismo fondamentale per filtrare i dati non necessari.
 - *Continuità:* La continuità della percezione è fondamentale per la consapevolezza del flusso temporale e la connessione con la realtà. Interruzioni o interferenze generano déjà-vu (paramnesia).
 - *Progettare per la memoria sensoriale:* Ridurre il carico mentale necessario per interpretare lo stimolo sensoriale (pochi elementi semplici, ben differenziati,

solidamente posizionati nell'interfaccia). Usare sistematicamente la teoria della Gestalt per aiutare la strutturazione del percetto (permettere raggruppamenti, rendere evidenti le differenze significative, nascondere quelle non significative).

- **Memoria a Breve Termine (Short Term Memory - STM)** (o Memoria di Lavoro):

- Qui vengono mantenuti ed elaborati i dati rilevanti per i compiti attualmente in esecuzione.
- Utilizza circa **7 ± 2 "chunk"** o blocchi (possibilmente strutturati) di informazione. Questi chunk sono sensorialmente modali e permettono il raggruppamento (es. 52147651210 vs 055 456 712 65).
- La STM è **sempre piena**. Se un nuovo chunk di informazione viene immagazzinato, è solo sostituendo uno precedente: qualcosa deve essere rimosso.
- La persistenza è di circa **15 secondi**. Può essere aumentata con la ripetizione uditiva (rehearsal).
- La ritenzione è amplificata dall'**importanza** (prima) e dalla **freschezza** (poi).
- *Progettare per la memoria a breve termine*: Ridurre il carico mentale necessario per mantenere il contesto del dialogo e la coerenza. Avere meno chunk (complessi se necessario) da tenere a mente (7±2 elementi distinti). Aiutare a recuperare chunk scomparsi dal contesto visivo. Aiutare a trattenere una maggiore quantità di informazioni attraverso il raggruppamento.

- **Memoria a Lungo Termine (Long Term Memory - LTM)**:

- Organizzata in:
 - **Memoria Episodica**: registrazione di eventi ed esperienze in modo seriale.
 - **Memoria Semantica**: registrazione di fatti, concetti e abilità apprese nel passato.
-
- La LTM non è mai una registrazione completa del percetto, ma una post-elaborazione delle parti rilevanti con forti filtri.
- *Memoria Eidetica (o fotografica)*: Il ricordo con un gran numero di dettagli (ricordare il percetto piuttosto che la sua elaborazione). Avviene principalmente nei bambini in età prescolare, estremamente rara negli adulti. (Cfr. "Funes o della memoria" di J.L. Borges). Tranne in tali casi patologici, il cervello adulto ha imparato a non trattenere mai più dettagli del percetto di una sua forma estremamente semplificata.

- La ritenzione nella LTM è ottenuta attraverso l'elaborazione; non conosciamo né limiti né durata dell'immagazzinamento.
- L'organizzazione è gerarchica e associativa: molto probabilmente può essere mappata come una rete semantica che collega concetti in categorie e sottocategorie.
- Concetti disconnessi (senza connessioni logiche ad altri fatti della nostra conoscenza) sono più difficili da trattenere rispetto a quelli connessi.
- Concetti astratti sono più difficili da trattenere rispetto a concetti concreti.
- Concetti con un livello più alto di connessione emotiva (non necessariamente personale) sono più facili da trattenere rispetto a quelli "aridi".
- La categorizzazione non è neutra e non si basa solo sulla motivazione semantica. La coloritura temporale ed emotiva gioca un ruolo importante.
- *Interferenza*: Accedere a un ricordo può causare accesso anche a un altro completamente disconnesso da un punto di vista logico
- *Dimenticare (Due teorie)*:
 - **Decadimento (Decay)**: ricordi non frequentemente accessi decadono lentamente ma naturalmente e scompaiono – connesso all'"effetto punta della lingua".
 - **Interferenza**: la ritenzione di nuove informazioni sostituisce naturalmente e immediatamente informazioni simili più vecchie (es. codici ATM).
- La memoria funziona meglio nel **riconoscere** che nel **richiamare**. Recuperare una nota scritta con il vecchio codice ti farà immediatamente riconoscere il numero e il suo significato.
- *Progettare per la memoria a lungo termine*: Non fare mai affidamento sulla semplice memoria dei fatti. Cercare di trasmettere una narrazione nella sequenza dei passaggi (alternativamente, la memoria iconica o visiva è migliore di quella episodica o puntuale). Permettere una facile astrazione per consentire la ritenzione di chunk più semplici. Usare verbi per le azioni e nomi per i concetti. Creare una narrazione per l'interazione.

- **Riflessioni Aggiuntive sulla Memoria e il Design:**

- Guardare e scegliere è più facile che ricordare e digitare.
- Le immagini sono più facili da riconoscere delle parole (le miniature sono più utili delle descrizioni testuali per le immagini).
- La visibilità deve essere proporzionale all'importanza e alla frequenza d'uso della funzione.
- Usare aiuti visivi per ricordare all'utente dove si trova.

3. Attenzione (Attention)

- L'attenzione è la **selezione di uno o pochi stimoli sensoriali** che ci raggiungono.
- Può essere **focalizzata** (scrivere) o **divisa** (guidare parlando o ascoltando la radio).
- **Effetto cocktail party**: la nostra abilità di filtrare (escludere) la maggior parte degli stimoli anche da un solo senso.
- Il compito decide su cosa concentrarsi. Le aspettative hanno un impatto sull'attenzione.
- L'attenzione è **modale**: eventi inaspettati o fatti fuori modalità sono fonte di distrazione.
- Sono necessari aiuti cognitivi per mantenere l'attenzione.
- **Progettare per l'Attenzione**:
 - Ridurre il carico cognitivo (anche se questo aumenta la quantità di azioni atomiche da eseguire).
 - Guidare gli utenti a scegliere percorsi d'azione più veloci o più facili.
 - Mostrare chiaramente lo stato del sistema e il livello di progresso nel completamento di un compito.
 - Rendere il sistema familiare.
 - Lasciare che il computer faccia i calcoli (es. totali).
 - Usare termini familiari.

4. Elaborazione: Ragionamento (Processing)

- Il ragionamento è come usiamo la conoscenza che già abbiamo per generare conclusioni sull'evento su cui ci stiamo concentrando. Usiamo il ragionamento per generare nuove informazioni e risolvere problemi nella nostra vita quotidiana.
- Questo avviene in modo semi-conscio: a volte raggiungiamo un risultato o una soluzione senza essere realmente consapevoli del processo seguito per arrivarci.

- Almeno tre tipi diversi di ragionamento:
 - **Ragionamento Deduttivo:**
 - Partendo da un'assunzione generale e un caso concreto, otteniamo una conclusione precisa.
 - Il **sillogismo** è lo strumento più famoso. (Es. Tutti gli uomini che respirano sono vivi; John respira; QUINDI John è vivo).
 - Non siamo sempre bravi con il ragionamento deduttivo, specialmente in presenza di assunzioni false o parziali.
 - **Ragionamento Induttivo:**
 - Partendo da molti casi omogenei, produco (induco) una regola generale (inferenza o generalizzazione).
 - (Es. Tutti gli elefanti che ho visto hanno la proboscide; QUINDI Tutti gli elefanti hanno la proboscide).
 - L'inferenza è inaffidabile e facile da smentire (basta mostrare un elefante senza proboscide). Inoltre, non può essere completamente provata a meno che non si possano esaminare sistematicamente tutti i casi.
 - Tuttavia, è il metodo usuale per generare nuove regole nella nostra vita quotidiana e nella scienza.
 - **Ragionamento Abduittivo:**
 - Dato un caso (a cui si possono applicare molte regole), scegliamo la regola migliore che si applica al caso.
 - (Es. John sta guidando troppo veloce; John guida veloce quando è ubriaco; QUINDI John è ubriaco).
 - Anche questo meccanismo è imperfetto: molte regole potrebbero applicarsi e potrebbero esserci regole sconosciute. (John potrebbe avere un'emergenza).
 - Inoltre, un'assunzione ingiustificata con l'abduzione è che esista una regola per spiegare il caso, e dobbiamo solo identificarla.
 - Nei sistemi informatici, spesso assumiamo che ciò che accade sullo schermo derivi dall'azione che abbiamo appena eseguito; quindi, se i due fatti sono disconnessi, questo crea confusione ed errore.

Risoluzione di Problemi (Problem Solving)

- Trovare una soluzione a un problema nuovo o non familiare. Gli esseri umani sono in grado di adattare qualsiasi conoscenza che hanno a nuove situazioni.
- **Comportamentismo (Behaviorism)** (fine XIX secolo): la risoluzione dei problemi si basa sull'applicazione di regole esistenti o per tentativi ed errori, esplorando possibili soluzioni finché non se ne trova una buona.
- **Teoria della Gestalt** (metà XX secolo): la risoluzione dei problemi si basa sull'applicazione di regole esistenti (approcci riproduttivi) così come sulla riflessione e ristrutturazione del problema in termini diversi (approcci produttivi). (Es. problema delle due corde appese al soffitto).
- **Teoria dello Spazio del Problema (Problem space theory)** (anni '70): il problema è espresso come la ricerca di un percorso all'interno di uno spazio dove c'è uno stato iniziale (il problema) e uno stato finale (la soluzione). Vengono identificati possibili luoghi intermedi desiderabili, e la risoluzione del problema è espressa come trovare un percorso verso un luogo intermedio più vicino, e poi l'analisi ricomincia dalla posizione più vicina. Ogni sottoproblema viene analizzato cercando euristiche, competenze e analogie.

Prestazioni degli Esseri Umani (Performances of Human Beings)

- I **tempi di risposta** dipendono fortemente dal tipo di stimolo (in particolare, rispondiamo molto più velocemente al suono che allo stimolo visivo).
- Esistono **compromessi tra velocità e accuratezza**.
- Le risposte automatiche diminuiscono in accuratezza.
- Le risposte ripetute diminuiscono in accuratezza e velocità.
- La **stanchezza** diminuisce velocità e accuratezza.
- Gli **errori di azione (imprecisione)** impattano sulle prestazioni tanto quanto le decisioni sbagliate.
- **Legge di Fitts (Fitt's Law)**: Il tempo necessario per raggiungere un bersaglio sullo schermo con un dispositivo di puntamento è proporzionale alla distanza e inversamente proporzionale alla dimensione del bersaglio.
 - $T = a + b \log_2(2D/W)$
 - dove: T = tempo per il bersaglio, a, b = costanti da determinare, D = distanza del bersaglio, W = dimensione del bersaglio.

Conclusioni

- Limiti e vincoli degli esseri umani come "dispositivi".
- Caratteristiche dei canali di I/O degli esseri umani.

- Tipi di meccanismi di memoria ed elaborazione umana.
- Prestazioni degli esseri umani.

Comprendere questi aspetti è essenziale per progettare interfacce uomo-macchina che siano non solo funzionali ma anche efficienti, efficaci e soddisfacenti per l'utente.

Riferimenti Bibliografici:

- Dix, A. et al. (1998). *HCI*. Prentice Hall, capitolo 1.
- Johnson, J. (2010). *Designing with the mind in mind*. Morgan Kaufmann.
- Egan, D. (1988). "Individual differences in HCI". In M. Helander (Ed.), *Handbook of HCI*. North-Holland.

13 - The Evaluation - Inspection

Introduzione al Capitolo

Nel capitolo precedente abbiamo introdotto il concetto generale di **valutazione** nel contesto dello User & UX Design, distinguendo tra attività condotte internamente al team di sviluppo e quelle che coinvolgono utenti esterni. Mentre il **testing con gli utenti** (valutazione esterna) fornisce feedback diretto sull'esperienza reale, esiste un'importante famiglia di metodi di valutazione che possono essere condotti **internamente al team di progettazione**: questi rientrano sotto il nome di **ispezione**.

Prima di addentrarci nelle specifiche tecniche di ispezione, ricordiamo brevemente alcuni principi generali già discussi:

- Si possono valutare diversi artefatti di progettazione, dagli **schizzi** iniziali ai **mockup**, ai **prototipi** fino al **sistema funzionante**.
- La **regola fondamentale delle valutazioni** rimane valida: *"Quanto prima si valuta il sistema, tanto meno precisi saranno i risultati, ma tanto meno costoso sarà risolvere i problemi identificati."* Questo sottolinea l'importanza di tecniche di valutazione rapide ed economiche nelle fasi iniziali, come appunto le ispezioni.

L'ispezione, dunque, è una fase di valutazione che avviene **all'interno del team di design**. Proprio per questa sua natura interna, rappresenta uno strumento **economico** (non richiede il reclutamento di utenti esterni, né laboratori specifici) per la valutazione dell'usabilità di un sistema. Tuttavia, è importante riconoscere che, essendo condotta da persone che conoscono già il progetto (o che

sono esperti del dominio), può essere anche **molto imprecisa** nel prevedere tutti i problemi che utenti reali potrebbero incontrare, poiché manca la prospettiva "ingenua" dell'utente finale.

Nonostante questa potenziale imprecisione, i metodi di ispezione sono preziosi per identificare una classe significativa di problemi, specialmente quelli legati a violazioni di principi di usabilità consolidati o a incoerenze logiche nel design.

1. Le Principali Attività di Ispezione

Nella fase di ispezione, possiamo identificare tre attività rilevanti e complementari:

1. **Cognitive Walkthrough (Passeggiata Cognitiva)**: Una simulazione fittizia e passo-passo dell'esecuzione di un compito da parte di un utente target, e la valutazione empirica (basata sulla conoscenza dell'utente e dell'interfaccia) della verosimiglianza di tale simulazione.
2. **Action Analysis (Analisi delle Azioni)**: Un'analisi, spesso quantitativa, delle specifiche azioni che devono essere eseguite per compiere una determinata operazione o compito.
3. **Heuristic Analysis (Analisi Euristica o Applicazione di Linee Guida)**: La valutazione delle interfacce basata su regole di buon senso, principi di usabilità riconosciuti o linee guida specifiche, derivate dall'esperienza e dalla ricerca.

Analizziamo ora ciascuna di queste attività in dettaglio.

2. Il Cognitive Walkthrough (CW)

Il **Cognitive Walkthrough (CW)** è un metodo formalizzato per **immaginare i pensieri e le azioni degli utenti** quando utilizzano un'interfaccia per eseguire un compito per la prima volta (o come se fosse la prima volta). È particolarmente utile per valutare la facilità di apprendimento e l'intuitività di un sistema.

2.1. Il Processo del Cognitive Walkthrough

Si prende il sistema, un prototipo o anche solo una serie di disegni dell'interfaccia. Si seleziona un **compito specifico** che un utente target dovrebbe eseguire con quell'interfaccia. A questo punto, il team di valutazione (o un valutatore singolo) deve **raccontare una storia credibile** su ogni azione che l'utente deve eseguire per completare il compito.

La storia è **credibile** se si può motivare ogni azione dell'utente facendo affidamento su:

- Conoscenza generale dell'**utente presunto** (le sue abilità, esperienze pregresse, aspettative).
- Le **indicazioni (affordance, segnali, etichette)** e il **feedback** forniti dall'interfaccia stessa.

Se non si riesce a raccontare una storia credibile, cioè se le azioni richieste all'utente non sembrano una logica conseguenza delle sue conoscenze e di ciò che l'interfaccia comunica, allora c'è un **problema di interfaccia**.

L'**obiettivo** del CW è determinare la **plausibilità dell'usabilità** dell'interfaccia per il segmento di utenti scelto, rispetto a compiti specifici.

2.2. Ingredienti Necessari per un Cognitive Walkthrough

Un CW efficace richiede quattro ingredienti fondamentali:

1. Una **descrizione o un prototipo dell'interfaccia**, il più dettagliato possibile.
2. La **descrizione di un compito**, possibilmente uno di quelli rappresentativi identificati nella progettazione basata sui compiti (task-based design) o sugli obiettivi (goal-based design).
3. Un **elenco completo e scritto delle azioni necessarie** per completare il compito. Questa sequenza ideale di azioni è talvolta chiamata **"Happy Path"** (Percorso Felice), perché rappresenta il modo ottimale e senza errori per raggiungere l'obiettivo.
4. Una **chiara descrizione dell'Utente Target** (la *persona* o il profilo utente), incluse le sue abilità, conoscenze pregresse e aspettative.

Basandosi su questi ingredienti, compito per compito, il team costruisce la "storia" dell'interazione e ne valuta la credibilità.

2.3. Esempio di Cognitive Walkthrough: L'Interfaccia di una Fotocopiatrice

Supponiamo di dover valutare l'interfaccia di una fotocopiatrice. Ci vengono forniti:

- Il disegno di un tastierino numerico.
- Un pulsante con l'icona "documento con freccia" (o la scritta "123").
- Un pulsante "soft" (un interruttore a bilanciere) sul retro della macchina per accenderla e spegnerla.
- L'informazione che la macchina si spegne automaticamente dopo cinque minuti di inattività.

Compito: Copiare una singola pagina.

Utente: Una segretaria neo-assunta.

Storia Proposta (Happy Path):

"La segretaria deve fare una copia. Sa che la fotocopiatrice deve essere accesa, quindi preme il pulsante di accensione. Inserisce il foglio da copiare nell'apposito alloggiamento e preme il pulsante 'copia' (quello con l'icona)."

Valutazione della Credibilità della Storia:

Questa storia potrebbe **non essere molto credibile**:

- Come fa la segretaria a sapere se la fotocopiatrice è spenta o accesa (manca un feedback visibile sullo stato)?
- Come fa a sapere dov'è il pulsante di accensione, se è sul retro e non visibile?
- Come fa a sapere come inserire correttamente il foglio di carta (orientamento, faccia in alto/basso)?
- Siamo sicuri che capisca che il pulsante con l'icona "documento con freccia" o "123" significhi "copia"?

Attenzione: Durante il CW, **non dobbiamo valutare direttamente l'interfaccia in sé ("questo pulsante è brutto")**, ma la **credibilità della storia dell'utente che** . Questa analisi può portare a revisioni sia dell'interfaccia (per renderla più chiara) sia della storia (per renderla più plausibile).

Nell'esempio, potremmo decidere di:

- Aggiungere un display che indica quando la fotocopiatrice è pronta.
- Aggiungere un disegno sul fronte che indica come inserire i fogli.
- Spostare l'interruttore principale in una posizione visibile.
- Cambiare l'etichetta del pulsante in un più esplicito "Copia".

A questo punto, si ripete l'esperimento mentale con la nuova interfaccia e la nuova storia.

2.4. Tipi di Problemi Scoperti dal CW

Il CW può scoprire diversi tipi di problemi:

- **Assunzioni errate del designer sul ragionamento degli utenti:** ("Perché l'utente dovrebbe pensare che la fotocopiatrice debba essere accesa prima di usarla? Magari si aspetta che sia sempre in standby.")
- **Comandi ovvi per il designer ma non per l'utente:** ("L'utente sa che vuole accendere la macchina, ma sa dove trovare l'interruttore?")
- **Problemi con etichette e prompt (inviti all'azione):** ("Come fa a sapere come inserire il foglio? Siamo sicuri che capisca l'icona 'X'?")
- **Problemi di feedback:** ("Come fa l'utente a sapere se la fotocopiatrice è accesa, spenta, o sta elaborando la copia?")

2.5. Errori Comuni nella Progettazione di un CW

Ci sono due errori comuni da evitare:

1. **Confondere l'elenco delle azioni (Happy Path) con il walkthrough stesso:** Il senso del CW è raccontare *credibilmente* come l'utente esegue le azioni ottimali per completare il compito, non descrivere come l'utente *scopre* queste azioni attraverso tentativi ed errori. Il CW valuta se l'Happy Path è facile da seguire per un utente "guidato" dall'interfaccia.
2. **Confondere il CW con un test utente effettivo:** Il CW identifica una classe di problemi (spesso legati alla comprensibilità e alla logica della sequenza di azioni) che un test con un numero limitato di utenti (5-10) potrebbe non far emergere con la stessa chiarezza. Tuttavia, il test con utenti reali identifica aspetti dell'esperienza vissuta (frustrazione, errori inaspettati, strategie alternative) che il CW da solo non può scoprire. Sono metodi complementari.

2.6. Differenze tra Cognitive Walkthrough e Scenari/Use Case

Un CW è simile per certi versi a uno **scenario** o a un **caso d'uso (use case)**, ma ci sono differenze fondamentali:

- Uno scenario o use case tipicamente **non include un prototipo dettagliato dell'interfaccia né la descrizione analitica delle azioni** (l'Happy Path) per eseguire il compito.
- Uno scenario o use case mira spesso a **definire i requisiti e a costruire l'interfaccia** e l'Happy Path, non primariamente a valutarli.
- Gli scenari sono **credibili per costruzione** (raccontano una storia plausibile d'uso), mentre un CW diventa **accettabile solo dopo che la storia dell'interazione è stata resa credibile** attraverso l'analisi e le eventuali modifiche al design.

2.7. Domande di Autovalutazione durante un CW

Durante l'analisi di ogni passo della "storia" del CW, il team dovrebbe porsi domande come:

- È realistico che l'utente (il nostro personaggio/persona) cercherà di compiere questa specifica azione in questo momento?
 - Il controllo (pulsante, link, menu) che comanda questa azione è disponibile e visibile?
 - C'è un legame ovvio e comprensibile tra il controllo e l'azione che l'utente vuole compiere?
 - Il feedback fornito dall'interfaccia dopo l'azione è appropriato, tempestivo e comprensibile?
-

3. L'Analisi delle Azioni (Action Analysis)

L'**Analisi delle Azioni** è un processo di valutazione che esamina da vicino la **sequenza di azioni** da compiere per completare un compito. Esistono due tipi principali:

3.1. Analisi Formale delle Azioni (es. Keystroke-Level Analysis, GOMS)

L'approccio formale all'analisi delle azioni è utilizzato per fare **predizioni accurate del tempo** impiegato da un **utente esperto** nell'eseguire un **compito di routine**.

- **Processo:** Per fare ciò, è necessario stimare i tempi per eseguire ogni singolo passo (fisico e mentale) del compito e poi sommarli. Il passo tipico è la **pressione di un tasto (keystroke)**, da cui il nome **Keystroke-Level Analysis (KLM)**.
- L'**obiettivo** è il calcolo specifico dei tempi medi di utilizzo di un sistema o, più spesso, di un singolo widget o componente dell'interfaccia.
- La stima del tempo di ogni azione elementare (es. premere un tasto, muovere il mouse, pensare al prossimo passo) è derivata da **tabelle standard** ottenute testando centinaia di utenti in migliaia di situazioni e mediando i risultati. Se un controllo non è descritto in queste tabelle, si approssima con uno simile o si devono condurre test specifici per quel nuovo controllo.
- Per ottenere il tempo totale di un compito, si adotta un approccio **top-down**: si scompone il compito nel suo percorso ottimale (Happy Path) in azioni elementari e si associano i tempi appropriati a ciascuna azione.

Il modello **GOMS (Goals, Operators, Methods, Selection rules)** è uno dei primi e più noti modelli di analisi delle azioni per la performance. Descrive il comportamento dell'utente in termini di:

* **Goals (Obiettivi):** Cosa l'utente vuole raggiungere.

* **Operators (Operatori):** Azioni elementari percettive, motorie o cognitive (es. premere un tasto, leggere un'etichetta).

* **Methods (Metodi):** Sequenze apprese di operatori per raggiungere un obiettivo (es. il metodo per salvare un file).

* **Selection rules (Regole di Selezione):** Regole usate dall'utente per scegliere tra più metodi disponibili per raggiungere lo stesso obiettivo.

L'input per un'analisi GOMS è una descrizione dettagliata dell'interfaccia e dei compiti; l'output sono misure quantitative (tempo di esecuzione previsto) e qualitative (descrizione della strategia dell'utente).

Il **KLM (Keystroke-Level Model)** è una delle molte evoluzioni di GOMS ancora ampiamente utilizzata. Prevede sei operatori principali:

* **K (Keystroke):** Pressione di un tasto o di un pulsante del mouse. (NB: "A" maiuscola sono due K: Shift + A).

* **P (Pointing):** Posizionare il puntatore del mouse su un oggetto. Muovere il mouse è separato dal clic (K).

* **H (Homing):** Muovere le mani dalla tastiera al mouse o viceversa, o posizionare le dita su un touchscreen.

* **D (Drawing):** Tracciare manualmente linee (usato meno frequentemente).

* **M (Mental preparation):** Tempo necessario per pensare, pianificare o prendere una decisione.

* **R (System Response time):** Tempo di attesa per la risposta del sistema (anche W da Wait).

Esempio di calcolo KLM: Inserire "Via Irnerio 36" in un campo di testo.

Tempi standard (esempi indicativi): K (dattilografo medio) = 0.28s, B (Button press/release) = 0.1s, P = 1.1s, H = 0.4s, M = 1.35s.

1. Iniziare l'azione (M)
2. Trovare il campo di testo corretto sullo schermo (M)
3. Portare il puntatore del mouse sul campo (P)
4. Premere il pulsante del mouse (B - inizio click)
5. Rilasciare il pulsante del mouse (B - fine click)
6. Spostare le mani dal mouse alla tastiera (H)
7. Digitare "Via Irnerio 36" (14 caratteri, di cui 2 maiuscole, 2 numeri. Assumendo che maiuscole richiedano 2K e numeri come K normali: es. 16K).

Tempo Totale = $2M + 1P + 2B + 1H + 16K = (2 \cdot 1.35) + 1.1 + (2 \cdot 0.1) + 0.4 + (16 \cdot 0.28) = 2.7 + 1.1 + 0.2 + 0.4 + 4.48 = 8.88$ secondi.*

Limiti dell'Analisi Formale:

Un'analisi formale di un'interfaccia complessa è un compito arduo: 10 minuti di interazione possono richiedere la descrizione di migliaia di azioni e relativi tempi. Inoltre, la descrizione del compito e delle azioni dell'utente è a discrezione del valutatore, portando a possibili discrepanze. Per questo, è utile solo in circostanze speciali:

- * Per esaminare aspetti molto specifici di un'interfaccia (es. la performance di un nuovo widget).
- * Per esaminare compiti molto strutturati e controllati (es. analizzare il carico di lavoro di un operatore di helpdesk).

Pro e Contro dell'Analisi Formale (Keystroke-Level):

- **Pro:**
 - **Pratica:** Non c'è bisogno di verificare i dati con utenti reali in questa fase.
 - **Generale:** Non richiede una profonda conoscenza di modelli psicologici complessi per essere applicata (una volta che si hanno le tabelle).
 - **Utilizzo precoce:** Non necessita di prototipi funzionanti (possono bastare specifiche dettagliate).
-
- **Contro:**
 - Restituisce principalmente il **tempo** come metrica fondamentale di performance, trascurando altri aspetti dell'esperienza.
 - È progettata per **utenti esperti** che eseguono **compiti di routine** e **non commettono errori**.
 - Non c'è spazio per modellare apprendimento, esplorazione, esitazione, errori e "slip" (sviste).
 - Il modello per le operazioni mentali (M) è molto **semplificato**.
 - Produrre la lista completa delle azioni può essere **dispendioso in termini di tempo** per compiti lunghi.
 - Più lungo è il compito, più impattante è il numero di operazioni mentali (M) stimate rispetto alle azioni fisiche, e quindi più impreciso può essere il risultato calcolato. È principalmente utile per compiti **brevi e ben noti**.
-

3.2. Analisi Informale delle Azioni

L'analisi informale ignora il micro-dettaglio e si concentra sul **quadro generale ("big picture")**, elencando una serie "naturale" di azioni e valutandole globalmente. Invece di "prendere la mano dalla tastiera e afferrare il mouse", le azioni qui descritte sono del tipo: "scegliere l'opzione X dal

menu Y". È anche chiamata "**back-of-the-envelope action analysis**" (analisi "sul retro della busta", per indicare la sua natura approssimativa e rapida).

L'**obiettivo** è la determinazione euristica (basata sull'esperienza) dei passi con il maggior "peso" (in termini di tempo percepito o numero di azioni atomiche) nell'esecuzione di un compito, e quindi delle potenziali fonti di eccessiva complessità, perdita di tempo o disorientamento.

L'enfasi non è sul decimo di secondo per valutare la performance di un widget, ma sul valutare risposte a domande come:

- Posso eseguire un compito semplice in modo semplice?
- Posso eseguire un compito frequente rapidamente?
- Quanti passi e informazioni devo imparare prima di poter eseguire un compito?
- Abbiamo descritto chiaramente ogni passo nella documentazione (se presente)?

Senza essere precisa come l'analisi formale, l'analisi informale è più **robusta** e meno soggetta a inaccuratezze dovute alla micro-stima, e può essere usata per:

- Verificare che l'esecuzione di un compito non richieda tempi comparabili (o superiori) al farlo a mano, su carta, o con un'applicazione diversa.
- Decidere se l'aggiunta di una nuova feature complicherà o meno il resto dell'interfaccia.
- Decidere se aggiungere modi multipli per compiere lo stesso compito (scorciatoie vs. percorsi standard).
- Verificare quali operazioni possono generare un errore e quanto grave potrebbe essere tale errore (in termini di tempo necessario per correggerlo).

4. L'Analisi Euristica (Heuristic Analysis / Evaluation)

Sia il Cognitive Walkthrough sia l'Analisi delle Azioni sono valutazioni **orientate al compito (task-oriented)**. Anche il testing con utenti, come descritto nel capitolo precedente, è spesso task-oriented.

Le valutazioni task-oriented hanno punti di forza e di debolezza:

- **Appropriatezza:** Valutano le caratteristiche di un sistema all'interno di un flusso di lavoro credibile e guidato da obiettivi indipendenti dall'interfaccia stessa.
- **Copertura (Coverage):** Permettono di valutare solo alcuni compiti, ignorandone molti altri.
- **Interazioni tra compiti:** Permettono di valutare come si comporta il sistema quando l'utente esegue molte azioni contemporaneamente o passa da un compito all'altro.

L'**Analisi Euristica** (o **Valutazione Euristica**) è uno strumento di ispezione (quindi eseguibile dal team di sviluppo) per valutare l'usabilità di un sistema **indipendentemente dai compiti specifici** per cui è stato progettato (è quindi una valutazione più **indipendente dal dominio applicativo**). Consiste nel **confronto dell'applicazione con principi generali e universalmente riconosciuti di usabilità**.

Si basa sul principio fondamentale della **consistenza esterna**, applicato sia in senso positivo che negativo:

- Se in altre applicazioni (note all'utente) la scelta X è stata positiva (ha portato a una buona usabilità), è probabile che sia positiva anche in questo sistema.
- Se in altre applicazioni la scelta Y è stata negativa, è probabile che sia negativa anche in questo sistema.

Questa valutazione è chiamata "euristica" perché fornisce **linee guida (euristiche)** per la **scoperta (appunto, euristica) di problemi di usabilità**. L'**obiettivo** è verificare l'aderenza del sistema alle linee guida identificate per il progetto e giustificare qualsiasi deviazione (o, più probabilmente, modificare il design per conformarsi).

4.1. Le Linee Guida (Guidelines)

Esiste una vasta quantità di linee guida di usabilità.

Possiamo distinguere tra linee guida generali e specifiche (es. governative):

A. Linee Guida Generali (Esempi):

- **Le 10 Euristiche di Usabilità di Nielsen e Molich (1994):** Probabilmente il set più famoso e utilizzato.
 - **Visibilità dello stato del sistema:** Il sistema dovrebbe sempre tenere informato l'utente su cosa sta succedendo, attraverso un feedback appropriato entro un tempo ragionevole.
 - **Corrispondenza tra il sistema e il mondo reale:** Il sistema dovrebbe parlare il linguaggio dell'utente, con parole, frasi e concetti familiari all'utente piuttosto che termini orientati al sistema. Seguire le convenzioni del mondo reale, facendo apparire le informazioni in un ordine naturale e logico.
 - **Controllo e libertà dell'utente:** Gli utenti spesso scelgono funzioni del sistema per errore e necessitano di una "uscita di emergenza" chiaramente segnalata per lasciare lo stato indesiderato senza dover passare attraverso un dialogo complesso. Supportare annulla (undo) e ripeti (redo).

- **Consistenza e standard:** Gli utenti non dovrebbero chiedersi se parole, situazioni o azioni diverse significano la stessa cosa. Seguire le convenzioni della piattaforma.
 - **Prevenzione degli errori:** Ancora meglio di buoni messaggi di errore è un design attento che previene il verificarsi del problema in primo luogo. Eliminare le condizioni soggette a errore o verificarle e presentare agli utenti un'opzione di conferma prima che si impegnino nell'azione.
 - **Riconoscimento piuttosto che ricordo (Recall):** Minimizzare il carico di memoria dell'utente rendendo oggetti, azioni e opzioni visibili. L'utente non dovrebbe dover ricordare informazioni da una parte all'altra del dialogo. Le istruzioni per l'uso del sistema dovrebbero essere visibili o facilmente recuperabili quando appropriate.
 - **Flessibilità ed efficienza d'uso:** Gli acceleratori — non visti dall'utente novizio — possono spesso velocizzare l'interazione per l'utente esperto, in modo che il sistema possa soddisfare sia utenti inesperti che esperti. Consentire agli utenti di personalizzare azioni frequenti.
 - **Estetica e design minimalista:** I dialoghi non dovrebbero contenere informazioni irrilevanti o raramente necessarie. Ogni unità extra di informazione in un dialogo compete con le unità di informazione rilevanti e ne diminuisce la visibilità relativa.
 - **Aiutare gli utenti a riconoscere, diagnosticare e recuperare dagli errori:** I messaggi di errore dovrebbero essere espressi in linguaggio semplice (senza codici), indicare precisamente il problema e suggerire costruttivamente una soluzione.
 - **Aiuto e documentazione:** Anche se è meglio se il sistema può essere usato senza documentazione, potrebbe essere necessario fornire aiuto e documentazione. Qualsiasi informazione di questo tipo dovrebbe essere facile da cercare, focalizzata sul compito dell'utente, elencare passi concreti da compiere e non essere troppo estesa.
- **Le linee guida di UserFocus.co.uk (commerciali, UK, 2014):** Userfocus è una società di consulenza UX che offre molte risorse. Le loro linee guida (circa 247 organizzate in 9 capitoli) coprono aree come:
 - Usabilità della home page (20 linee guida)
 - Orientamento al compito (44)
 - Navigazione e Architettura dell'Informazione (IA) (29)
 - Moduli e inserimento dati (23)
 - Fiducia e credibilità (13)
 - Qualità della scrittura e dei contenuti (23)

- Layout di pagina e design visivo (38)
- Usabilità della ricerca (20)
- Aiuto, feedback e tolleranza agli errori (37)

Offrono anche un file Excel per supportare la valutazione e generare grafici.

(Link: <http://www.userfocus.co.uk/resources/guidelines.html>)

B. Linee Guida Governative (Esempi): Molti governi ed enti pubblici pubblicano linee guida per i propri siti e servizi digitali, spesso con un forte accento sull'accessibilità.

- **USA: Research-Based Web Design & Usability Guidelines (2006):** Nate per il Dipartimento della Salute e dei Servizi Umani, poi standardizzate per tutti i siti governativi USA. Contengono 209 linee guida in 18 capitoli, coprendo l'intero processo di design e aspetti specifici (accessibilità, hardware/software, homepage, layout, navigazione, contenuti, test, ecc.).

(Link: https://www.usability.gov/sites/default/files/documents/guidelines_book.pdf)

- **EU: Europa Web Guide:** Regole e linee guida per i siti della Commissione Europea, coprendo aspetti editoriali, legali, tecnici, visivi e contrattuali. Include 10 regole e 4 blocchi di linee guida (Architettura e Navigazione, Raccomandazioni di Design, Linee Guida sui Contenuti, Ottimizzazione per i Motori di Ricerca - SEO).

(Link: https://commission.europa.eu/resources-partners/europa-web-guide_en)

- **EU: Usability guidelines for websites and products of statistical organisations (2020):** Focalizzate sulla disseminazione di statistiche ufficiali, ma con molti consigli utili per i siti della Pubblica Amministrazione in generale. Coprono temi come User Experience, Layout, Struttura e Navigazione, Ricerca e Filtri, Componenti di Design, Linguaggio Chiaro, Accessibilità Web, Funzionalità Avanzate, Design per Mobile.

(Link:

https://cros-legacy.ec.europa.eu/system/files/usability_guidelines_for_websites_and_products_of_statistical_organisations.pdf)

L'applicazione di queste (o altre) linee guida costituisce il cuore dell'analisi euristica. Tipicamente, uno o più valutatori esperti esaminano l'interfaccia confrontandola sistematicamente con ciascun punto delle euristiche scelte, annotando le violazioni e la loro potenziale gravità.

Conclusione del Capitolo sull'Ispezione

L'ispezione, nelle sue varie forme (Cognitive Walkthrough, Analisi delle Azioni, Valutazione Euristica), rappresenta un insieme di strumenti potenti ed economici a disposizione del team di progettazione per una valutazione interna dell'usabilità. Sebbene non possa sostituire completamente il testing con utenti reali, l'ispezione permette di identificare precocemente una vasta gamma di problemi, specialmente quelli legati alla logica dell'interazione, alla coerenza con standard e convenzioni, e alla facilità di apprendimento dei percorsi principali. Integrata in un processo di design iterativo, l'ispezione contribuisce significativamente a migliorare la qualità dell'esperienza utente prima ancora che il prodotto raggiunga gli utenti finali.

14 - The Evaluation - Testing

Introduzione: Perché Testare con Utenti Reali?

Valutare un sistema interattivo è una fase imprescindibile del processo di design. Come abbiamo visto, questa valutazione può avvenire in due modi principali: attraverso l'**ispezione**, una revisione critica interna condotta dal team di sviluppo, e attraverso il **testing**, che coinvolge utenti esterni. Sebbene l'ispezione sia un metodo rapido ed economico per identificare problemi di logica e coerenza, il testing con utenti reali rimane insostituibile.

Nessuna teoria o analisi interna, per quanto raffinata, può prevedere tutte le difficoltà e i comportamenti che emergeranno quando una persona reale interagirà con il prodotto. Gli utenti portano con sé aspettative, modelli mentali e differenze individuali (tecniche, psicologiche, fisiche) che solo l'osservazione diretta può rivelare. Testare con loro è l'unico modo per ottenere un riscontro credibile e dimostrabile sulla reale esperienza d'uso.

Una regola fondamentale guida l'intero processo di valutazione: **valutare presto e spesso**. Quanto prima si identifica un problema, tanto meno costoso sarà risolverlo. Per questo, si possono testare artefatti a diverso livello di fedeltà: dai semplici **schizzi su carta** ai **mock-up digitali**, fino a prototipi più avanzati che utilizzano la tecnica del **Mago di Oz** (dove un operatore umano simula le risposte del sistema) e, infine, il sistema funzionante. L'obiettivo è rimuovere i grossi ostacoli nelle fasi iniziali, lasciando le valutazioni finali per i dettagli e le verifiche formali.

1. Le Finalità del Testing: Test Formativi e Sommativi

I test di usabilità non sono tutti uguali; la loro natura dipende principalmente dallo scopo. La distinzione più importante è tra test formativi e sommativi.

- **Test Formativi:** Hanno lo scopo di **raccogliere problemi per migliorare il design**. Sono condotti a beneficio del team di progettazione, che è consapevole delle imperfezioni del prodotto e cerca indicazioni su dove e come intervenire. Questi test sono tipicamente informali, frequenti, economici e si concentrano sulla raccolta di dati **qualitativi** (il "perché" di un problema).
- **Test Sommativi:** Hanno lo scopo di **verificare la correttezza e il raggiungimento degli obiettivi**. Sono eseguiti a beneficio del cliente o per una validazione formale, per dimostrare che il prodotto soddisfa requisiti contrattuali o standard di usabilità. Sono test formali, condotti su un prodotto quasi completo, e mirano a raccogliere dati **quantitativi** (il "cosa", misurato con metriche oggettive).

È fondamentale non confonderli: un test sommativo non è un test formativo che casualmente non ha trovato problemi. Ha un rigore, una metodologia e uno scopo completamente diversi. Il ciclo di sviluppo ideale prevede molteplici cicli di test formativi e ri-progettazione, per poi arrivare a un test sommativo finale con la ragionevole certezza di superarlo.

2. Metodologie a Confronto: Full Usability Testing vs. Discount Usability Testing

A seconda del budget, del tempo e degli obiettivi, si possono adottare approcci molto diversi.

- **Full Usability Testing (o "Deluxe")**

È l'approccio tradizionale e più rigoroso. È **formalizzato**, con protocolli dettagliati, e si svolge in **laboratori di usabilità** attrezzati (con specchi unidirezionali, telecamere, software di tracciamento). Coinvolge un numero di partecipanti statisticamente significativo (**20-100 persone**) e un team di professionisti UX. L'obiettivo è raccogliere dati **quantitativi** robusti. È molto **costoso** (può facilmente superare i 15.000-20.000€) e viene utilizzato principalmente per test sommativi finali.

- **Discount Usability Testing (o "Guerrilla")**

Proposto da Jakob Nielsen come alternativa agile e a basso costo, è l'approccio più comune in contesti con risorse limitate.

- **Informale:** Spesso basta un computer, una stanza tranquilla e un membro del team che modera.
- **Economico:** Richiede pochi partecipanti per ciclo (**3-5 utenti** sono spesso sufficienti per scoprire la maggior parte dei problemi comuni).
- **Sequenziale:** Questo è il suo punto di forza. Dopo ogni sessione, il team analizza i problemi emersi e, se possibile, li corregge **prima di iniziare il test con l'utente successivo**. Questo permette di non "sprecare" utenti a riscontrare sempre gli stessi errori catastrofici e di far emergere problemi via via più nascosti.

- **Qualitativo:** Si basa su tecniche come il **Thinking Aloud informale** e produce principalmente spunti per il miglioramento, non metriche statistiche.

Sebbene il Discount Testing non abbia il rigore statistico dell'approccio "deluxe", la sua natura iterativa e sequenziale lo rende estremamente efficiente nell'identificare e risolvere rapidamente i problemi più gravi.

3. Progettare e Condurre un Test di Usabilità

Un test efficace, anche se informale, richiede una struttura.

1. **Definire il Protocollo di Test:** È il documento di pianificazione che definisce l'approccio, i compiti da testare, il profilo e il numero dei partecipanti, la logistica e le metriche da raccogliere.
2. **Scegliere i Partecipanti e i Compiti:** I partecipanti devono essere rappresentativi degli utenti target. I compiti devono riflettere scenari d'uso realistici e significativi, spesso derivati dall'analisi dei compiti fatta in precedenza.
3. **Condurre un Test Pilota (Dry Run):** Una prova generale con un partecipante "fuori quota" (es. un collega) è fondamentale per verificare che l'attrezzatura, i compiti e il protocollo funzionino correttamente.
4. **Considerare l'Etica:** È cruciale ottenere il **consenso informato** dei partecipanti, spiegando lo scopo del test, garantendo l'anonimato e il diritto di ritirarsi in qualsiasi momento. Bisogna rassicurare l'utente che si sta testando il sistema, non la sua abilità.
5. **Raccogliere i Dati:** La raccolta può concentrarsi su due tipi di dati, che idealmente richiedono sessioni separate:
 - **Dati di Processo (qualitativi):** Si ottengono osservando *come* e *perché* un utente agisce, principalmente attraverso la tecnica del **Thinking Aloud (TA)**, in cui si chiede all'utente di verbalizzare i propri pensieri, dubbi e intenzioni mentre esegue un compito.
 - **Dati Consuntivi (quantitativi):** Si raccolgono metriche oggettive sul risultato finale, come il **tasso di successo** nel completamento dei compiti, il **tempo impiegato** (Time on Task) e il **numero di errori**.

4. Le Metriche della Valutazione

Per misurare l'usabilità in modo oggettivo, si ricorre a metriche specifiche.

Metriche Quantitative di Performance:

- **Tasso di Successo:** La percentuale di utenti che completa con successo un compito.
- **Tempo sul Task:** Il tempo necessario per completare un compito.
- **Errori:** Il numero e la gravità degli errori commessi (es. catastrofico, grave, minore).
- **Efficienza:** Una metrica combinata che mette in relazione il successo con le risorse impiegate (es. tempo o numero di click).
- **Apprendibilità (Learnability):** La capacità degli utenti di migliorare le proprie prestazioni con l'uso ripetuto del sistema.

Metriche di Soddisfazione (spesso qualitative, ma quantificate tramite scale):

La soddisfazione è più difficile da misurare a causa del "bias di cortesia" (gli utenti sono restii a criticare), ma esistono strumenti standardizzati molto efficaci:

- **System Usability Scale (SUS):** Un questionario di 10 domande, rapido e affidabile, che fornisce un punteggio di usabilità percepita da 0 a 100 (un punteggio > 68 è considerato sopra la media).
- **Single Ease Question (SEC):** Una singola domanda post-compito ("Quanto hai trovato facile/difficile questo compito?") su una scala a 7 punti, ottima per confronti rapidi.
- **Net Promoter Score (NPS):** Misura la lealtà e la propensione al passaparola con una singola domanda ("Quanto è probabile che consiglieresti questo prodotto?").
- **Domande Aperte:** Forniscono un feedback qualitativo ricco e non strutturato. Per analizzarle in modo sistematico si possono usare tecniche come la **Grounded Theory**, che attraverso un processo di codifica (aperta e assiale) permette di far emergere categorie e temi dai dati testuali.

-
- **Valutare Presto, Valutare Spesso:** Il principio guida per ridurre i costi di sviluppo e migliorare la qualità finale del prodotto.
 - **Test Formativi vs. Sommativi:** I primi servono per *migliorare* il design (per il team), i secondi per *validarlo* (per il cliente).
 - **Discount UT è Sequenziale:** Il suo potere sta nel correggere i problemi tra una sessione e l'altra, accelerando il miglioramento del prodotto.

- **Dati di Processo vs. Dati Consuntivi:** Non si possono misurare efficacemente entrambi nello stesso test. Il "Thinking Aloud" (processo) altera le metriche di performance (consuntivi).
- **Il SUS è uno standard de facto:** Per una misurazione rapida e confrontabile dell'usabilità percepita, il System Usability Scale è uno strumento eccellente.
- **Ispezione del Design:** Valutazione interna al team (es. analisi euristica).
- **Testing:** Valutazione con utenti esterni.
- **Mago di Oz (Wizard of Oz):** Tecnica di prototipazione in cui un umano simula le risposte di un sistema avanzato.
- **Test Formativo:** Test per trovare e risolvere problemi durante lo sviluppo.
- **Test Sommativo:** Test per validare il prodotto finale rispetto a requisiti.
- **Dati Quantitativi:** Dati numerici che misurano il "cosa" (es. tempo, errori).
- **Dati Qualitativi:** Dati descrittivi che spiegano il "perché" (es. opinioni, commenti).
- **Full Usability Testing:** Metodologia formale, costosa, con molti utenti, per dati statisticamente validi.
- **Discount Usability Testing (Guerrilla UT):** Metodologia informale, economica, con pochi utenti, basata su cicli sequenziali di miglioramento.
- **Thinking Aloud (TA):** Tecnica in cui l'utente verbalizza i propri pensieri durante un test.
- **Protocollo di Test:** Il documento che pianifica e guida ogni aspetto di un test di usabilità.
- **System Usability Scale (SUS):** Questionario standard di 10 item per misurare l'usabilità percepita.
- **Net Promoter Score (NPS):** Metrica per misurare la lealtà del cliente e la propensione al passaparola.
- **Grounded Theory:** Metodologia sistematica per analizzare dati qualitativi e derivarne categorie e teorie.