

Teorie e tecniche della progettazione Web Lezione 6

Mario Verdicchio

Università degli Studi di Bergamo

Anno Accademico 2019-2020

Usability testing

- Lo **usability testing** è un complesso di metodi di ricerca con cui valutare un prototipo o un'esperienza
- Si valuta la capacità o non capacità di un utente di portare a termine un task specifico
- L'obiettivo è quello di scoprire le parti migliorabili del nostro prodotto

Usability secondo Jakob Nielsen

- **Learnability:** facilità con cui l'utente esegue task di base la prima volta che interagisce col prodotto, facilità con cui il prodotto viene appreso
- **Efficiency:** la velocità con cui un utente è in grado di concludere un task una volta che lo ha imparato
- **Memorability:** il tempo che un utente impiega a riacquistare dimestichezza con un prodotto dopo che lo ha imparato e se ne è allontanato
- **Errors:** il numero di errori commessi dagli utenti, la loro gravità, e il tempo necessario per recuperare
- **Satisfaction:** soddisfazione dell'utente nell'uso del prodotto

Tipologia di usability test

- **Moderati:** test in cui è prevista la presenza di un moderatore, una persona che facilita il test attraverso un copione
- **Non moderati:** test eseguiti attraverso uno strumento online (es.: usertesting.com) che raccoglie i dati del test svolto dagli utenti in maniera autonoma

Usability test: fondamentali

- **Obiettivi specifici:** è difficile poter testare tutte le funzionalità del prodotto in un'unica sessione di test, quindi è importante stabilire delle priorità sulla base degli obiettivi di business e informazioni pregresse (test precedenti, problemi riscontrati etc.)
- Ad es. per un prodotto nuovo potremmo decidere di testare lo user flow principale che segue il primo accesso (registrazione, funzionalità primarie, etc.)
- Ad es. per un prodotto già esistente in cui si è riscontrato un livello critico di abbandono al momento del checkout, possiamo testare il processo di pagamento con lo scopo di individuare criticità

Usability test: fondamentali

- **Direct task vs scenario task:** una volta stabilito quale funzionalità testare, si può creare un copione da seguire durante il test
- Direct task è la modalità in cui si chiede all'utente di svolgere un task specifico nel prodotto (es.: “crea una nuova nota”, “modifica una nota esistente”)
- Scenario task è la modalità in cui si “nasconde” il task all'interno di uno scenario (es.: “devi andare a fare la spesa e vuoi creare una nota con la lista della spesa”, “uscendo ti accorgi che ti sei dimenticato del latte e lo vuoi aggiungere alla lista”)
- Attenzione a non dare alcun tipo di suggerimento: “crea una nuova nota premendo su ‘+’” o “modifica una nota esistente cliccando sull'icona della matita” inficiano il valore del test

Usability test: fondamentali

- **Stato del prototipo:** prima di usare il prototipo per un test, gli sviluppatori devono essere consapevoli del suo stato
- Il prototipo va provato più e più volte prima di essere usato in un test: gli sviluppatori devono essere sicuri che non ci siano problemi e, nel caso in cui lo sviluppo del prototipo sia solo parziale, ciò deve essere fatto presente ai partecipanti in sede di test

Usability test: fondamentali

- **Partecipanti:** come per i sondaggi e le interviste per le user research, anche i partecipanti agli usability test andrebbero reclutati tra i gruppi che sono target primari del prodotto che si sta testando
- **Umanità:** trattare con garbo i partecipanti, essere precisi con le tempistiche, creare un'atmosfera accogliente per metterli a loro agio, offrire bevande e snack, assicurarli che non sono loro l'oggetto del test, bensì il prodotto che stanno per provare

Usability test: fondamentali

- **Equipaggiamento:** se il test avviene su più device (smartphone, tablet, laptop) ricordarsi di portarli tutti al test dopo un controllo del loro corretto funzionamento e del loro livello di batteria; lo stesso va applicato agli strumenti di registrazione audio e video nel caso siano da usare
- **Liberatorie:** Stampare e far firmare le corrette liberatorie (per il trattamento dei dati personali, dei dati personali di minorenni, per la registrazione audio e video etc.) a seconda del tipo di test da condurre

Copione di un usability test

- La struttura classica di un copione (script) per usability test contiene i seguenti elementi:
 - 1. Introduzione:** ci si presenta e si spiega al partecipante come verrà condotto il test, quanto durerà etc. È il momento di ricordargli che è il prodotto ad essere sotto esame e non il partecipante, e di fargli firmare tutte le liberatorie necessarie

Copione di un usability test

- 2. Raccolta di informazioni demografiche:** si chiedono le informazioni di base (es.: età, occupazione). In realtà se il partecipante al test fa parte di un gruppo di target precedentemente delineato, dovremmo essere già in possesso di queste informazioni. Esistono metodologie di test per le quali non abbiamo informazioni pregresse (es. “guerrilla testing” in cui si prendono utenti per strada) e allora questa raccolta si rende necessaria. Il formato delle domande aiuta ad avere più tatto (es.: “qual è la tua fascia d’età?” è meglio di “quanti anni hai?”; “qual è la tua fascia di reddito? dai 25mila ai 35mila euro annui? è meglio di “quanto guadagni al mese?”)

Copione di un usability test

- 3. Raccolta di informazioni di background:** a seconda della tipologia del prodotto che stiamo testando, è utile raccogliere informazioni sul rapporto che il partecipante ha con la relativa tecnologia e le sue competenze. Ad esempio, se il prototipo è una nuova alternativa a Snapchat, è utile conoscere le abitudini di comportamento dei partecipanti per quanto riguarda i social network con domande come “Per quante ore al giorno utilizzi i social network?” “In che circostanze li usi?” “Ne hai uno preferito?” etc.

Copione di un usability test

- 4. Inizio con domande aperte:** dopo aver concluso la parte sui dati del partecipante, passiamo a fargli domande sul prodotto da testare. In questa fase non gli facciamo eseguire un task specifico, ma gli poniamo domande più ampie come “Che impressione ti dà questo prodotto?” “Pensa ad alta voce e dimmi quali sono gli elementi più evidenti e con quali interagiresti subito.” “Quale pensi sia la finalità di questo prodotto?”

Copione di un usability test

- 5. Svolgimento del task:** è la fase centrale del test, in cui si chiede al partecipante di svolgere i task per i quali il test è stato organizzato. È molto importante ricordargli di pensare sempre a voce alta, in modo da comprendere i flussi del suo pensiero durante l'uso del prodotto. Altrettanto importante rimanere in silenzio e non aiutare il partecipante in alcun modo.

Copione di un usability test

- 6. Follow-up:** è la fase finale in cui possiamo fare domande integrative, ad esempio sulla soddisfazione del partecipante, sulla piacevolezza dell'esperienza, o addirittura fargli fare un sondaggio (es.: gli “after-scenario questionnaires” sono disegnati per misurare la difficoltà di un task percepita dal partecipante al test)

Usability test: quando?

- **Quando eseguire un usability test?** Non esiste una risposta universale. Potenzialmente, ogni fase di sviluppo del prodotto può essere verificata con un usability test. Una regola sempre valida: prima si testa qualcosa, prima se ne trovano i difetti, prima si sistemano. Il punto critico è capire quando una verifica è necessaria.

Usability test: con quanti utenti?

- **Qual è il numero giusto di partecipanti per avere risultati attendibili?** In un articolo oramai diventato famosissimo* l'esperto del settore Nielsen afferma che con 15 utenti si trova il 100% dei problemi di usabilità, mentre con 5 utenti si arriva all'85%, un compromesso ottimale tra costo e risultati. Addirittura si spinge a raccomandare, in presenza di un budget per 15 utenti, di fare 3 test con 5 utenti in 3 fasi diverse dello sviluppo del prodotto.

*<https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>

Le euristiche di Jakob Nielsen

- A proposito di Nielsen: danese, nato nel 1957, ha studiato Human-Computer Interaction (HCI) presso la Technical University of Denmark a Copenhagen, è considerato uno degli esperti mondiali di UX
- Sulla base della sua esperienza professionale, Nielsen ha sviluppato 10 euristiche dell'interaction design: principi generali che dovrebbero aiutarci a ridurre gli errori nel nostro progetto
- Euristica: regola emersa dall'esperienza che non ha pieno supporto teorico ma funziona la maggior parte delle volte nella pratica professionale

1. Visibilità dello stato di sistema

- Tenere sempre informato l'utente sullo stato del processo in corso con feedback puntuali (es.: stato del download di un file)

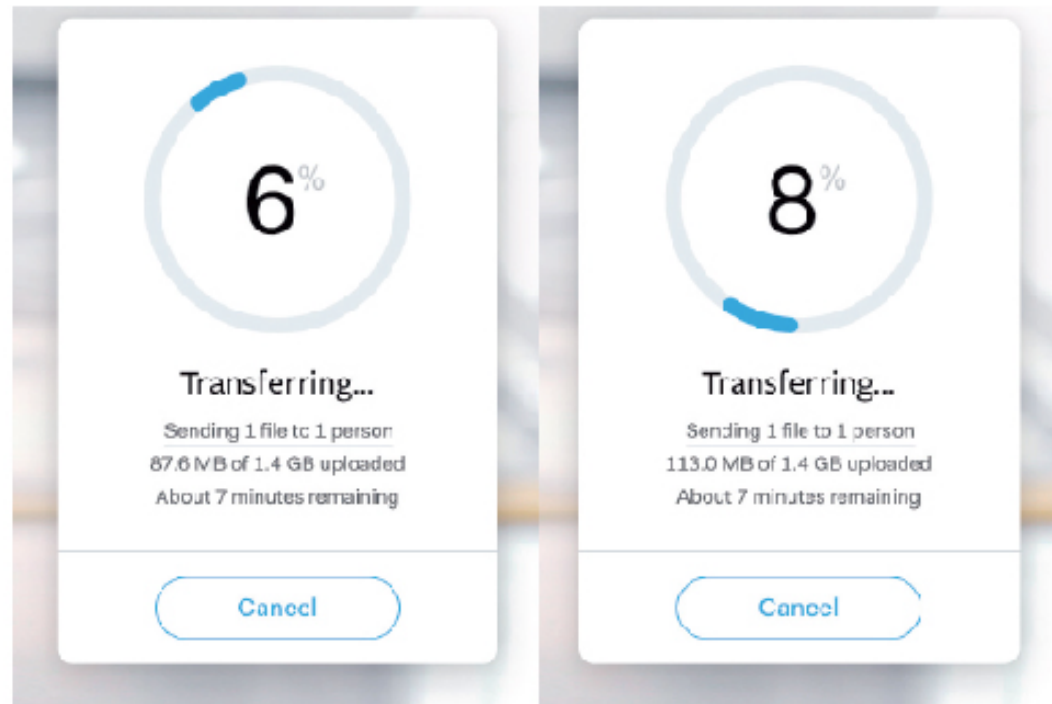


Figura tratta dal libro “Manuale di sopravvivenza per UX Designer” di Matteo di Pascale

2. Corrispondenza tra sistema e mondo reale

- Usare sempre icone, immagini e linguaggi che siano familiari all'utente e di facile intuizione (es.: icona a forma di carrello da cliccare per accedere agli acquisti fatti finora)



3. Controllo e libertà dell'utente

- Evitare di costringere l'utente che ha commesso un'operazione sbagliata a chiudere la app o il sito per ripartire da capo: l'utente deve sempre essere in grado di mantenere il controllo e avere la libertà di abbandonare uno stato indesiderato.

4. Coerenza e standard

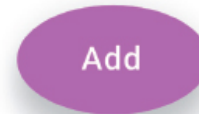
- Adottare la stessa convenzione e mantenerla attraverso tutte le pagine del sito: l'utente non deve mai trovarsi nella situazione in cui si chiede se diverse parole, pulsanti o icone significhino la stessa cosa

Prima pagina



NORMAL

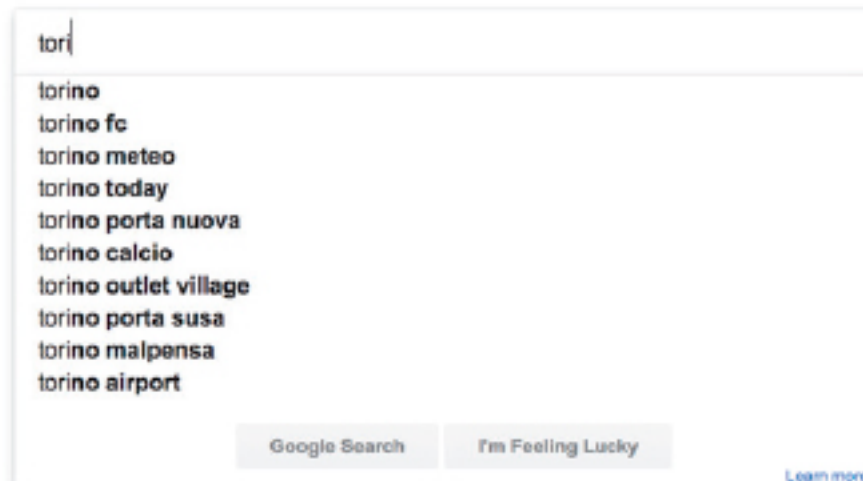
Seconda pagina



Button

5. Prevenzione degli errori

- Cercare sempre di incorporare meccanismi per prevenire errori (es.: l'autocompletamento delle ricerche su Google)



6. Riconoscimento invece che memoria

- Meglio non affidarsi alla memoria degli utenti per far loro eseguire task: i significati delle icone poco usate si dimenticano subito. Rendere tutte le funzionalità facilmente riconoscibili, senza fare affidamento a quanto imparato dall'utente

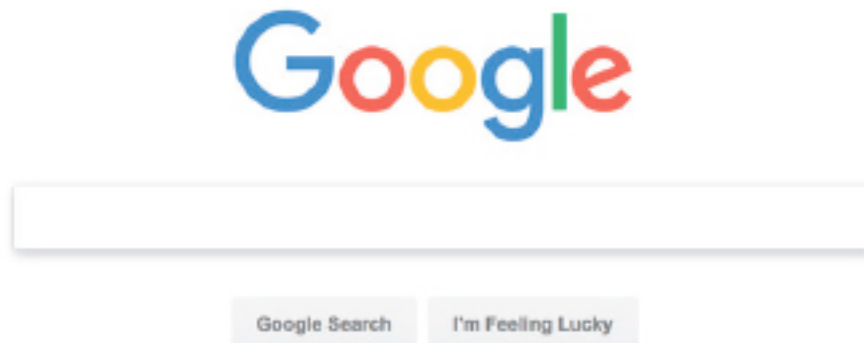


7. Flessibilità ed efficienza d'uso

- Un prodotto complesso (es.: strumento da disegno come Adobe Illustrator) deve essere flessibile nelle modalità d'uso e essere soddisfacente sia per utenti inesperti, sia per quelli avanzati: uno stesso task dovrebbe poter essere eseguito in più modi (es.: via menù a tendina, via shortcut su tastiera)
- Attenzione a non confondere questa euristica con la raccomandazione di minimizzare le modalità di interazione (es.: meglio non avere diversi modi di fare login su schermo): qui stiamo parlando di tipi di interazione diversi (es.: via schermo vs via tastiera)

8. Estetica e design minimalista

- “Less is more”: ogni elemento informativo non necessario toglie spazio alle informazioni necessarie, diminuendone la visibilità. Bisogna cercare sempre di essere essenziali.



9. Aiutare gli utenti a riconoscere e riprendersi dagli errori

- Evitare codici numerici oscuri e spiegare sempre con chiarezza qual è il problema per aiutare l'utente a correggersi e riprendere il corso normale del task
- Attenzione: sul libro di testo viene riportato l'esempio di una indicazione esplicita di errore su username oppure password, ma questa pratica è scoraggiata dagli esperti di sicurezza online (per minimizzare le informazioni date a potenziali utenti malintenzionati)

10. Aiuto e documentazione

- I sistemi sono sempre più spesso disegnati in modo tale da poter essere usati subito in maniera intuitiva. È comunque buona pratica accompagnarli con documentazione di supporto che sia
 - facile da consultare
 - focalizzata sui task che gli utenti devono compiere
 - fornita di liste di passaggi concreti da seguire