

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

SCUOLA DI SCIENZE

Informatica - 8009

**IL DESIGN THINKING COME STRUMENTO DI INNOVAZIONE
NELLO SVILUPPO SOFTWARE:**

Riprogettazione dell'esperienza utente nel caso Spotify

**Relatore:
Chiar.mo Prof.
FABIO VITALI**

**Presentata da:
BEATRICE ZAMAGNA**

**IIIa Sessione
Anno Accademico 2021/2022**

Indice

1	Introduzione	1
2	Esperienza utente e Design Thinking	5
2.1	Metodologie popolari per lo sviluppo software	5
2.2	Usabilità, esperienza utente e interfaccia utente	6
2.3	Il ruolo fondamentale della UX nello sviluppo software	8
2.4	Misurare l'esperienza utente: modelli e strumenti	9
2.4.1	Metodologie di valutazione dell'esperienza utente dei software	10
2.4.2	Modelli standard di misurazione dell'esperienza utente	12
3	Il Design Thinking: un approccio innovativo	21
3.1	Fasi e strumenti utilizzati	22
3.2	Vantaggi, sfide e raccomandazioni	24
3.3	Design Thinking: esempi di applicazione e successo	25
4	Applicazione del DT alla riprogettazione di Spotify	29
4.1	Raccolta dei dati	30
4.2	Definizione del problema	36
4.3	Ideazione	37
4.4	Prototipazione	38
5	Valutazione dei risultati ottenuti	45
6	Conclusione e sviluppi finali	49
	Bibliografia	53

Capitolo 1

Introduzione

Negli ultimi anni, l'industria dello sviluppo software ha assistito ad una crescente domanda di prodotti tecnologici innovativi e incentrati sull'utente. Tuttavia, i tradizionali processi di sviluppo software spesso faticano a soddisfare queste aspettative, portando a prodotti che non rispondono completamente alle esigenze degli utenti. Per risolvere questo problema, sempre più aziende stanno integrando il Design Thinking come metodologia di problem solving per guidare il processo di sviluppo software. Questa tesi mira a esplorare il potenziale del Design Thinking e il suo impatto sul processo di sviluppo software e sui prodotti che ne risultano.

Nonostante i numerosi benefici riscontrati nel Design Thinking applicato allo sviluppo software, la sua implementazione pratica nei progetti è ancora scarsa e molte aziende stentano ad applicare tale metodologia di problem solving ai loro progetti per varie ragioni. In questa tesi, esamineremo le ragioni dietro questa scarsa implementazione e suggeriremo possibili soluzioni per migliorare l'adozione del Design Thinking nell'industria dello sviluppo software.

La dissertazione è divisa in cinque capitoli. Per prima cosa, nel Capitolo 2.1, verranno presentati le metodologie più popolari seguite dalle organizzazioni per lo sviluppo software. Verrà posta attenzione al passaggio storico tra metodologia Waterfall e approccio Agile avvenuto intorno ai primi anni 2000 e alle motivazioni che hanno portato a questo passaggio. In quel periodo le aziende si sono infatti rese conto dell'importanza cruciale del coinvolgimento dell'utente nello sviluppo software per ottenere prodotti di successo e soddisfacenti, fattore che in approcci come quello Waterfall non era considerato.

Seguirà un approfondimento dei concetti di usabilità, esperienza utente e interfaccia utente con l'obiettivo di dare una definizione il più preciso possibile a questi concetti, ancora poco nitidi, e che saranno cardine per lo sviluppo del lavoro presentato. Per anticipare quello che verrà affrontato nel capitolo 2.2, si può dire che l'usabilità e l'esperienza utente sono concetti strettamente correlati, ma non sono la stessa cosa. L'usabilità è un concetto più pragmatico e utilitaristico e si riferisce all'efficacia e all'efficienza di un pro-

dotto o un sistema, mentre l'esperienza utente è un concetto più ampio e fa riferimento a tutti gli aspetti che contribuiscono all'esperienza che l'utente ha durante l'interazione con un prodotto o un sistema, inclusi gli aspetti emotivi e soggettivi. Per quanto riguarda l'interfaccia utente, essa è una delle componenti più importanti, ma non l'unica, che concorrono alla realizzazione dell'esperienza utente nella sua totalità.

Il coinvolgimento dell'utente finalizzato alla creazione di una esperienza utente appagante è un elemento fondamentale nella realizzazione di un software e spesso viene ancora sottovalutato dai team nel processo di sviluppo, portando a prodotti non pienamente soddisfacenti per l'utente e, di conseguenza, con limitato successo sul mercato. Nel capitolo 2.3 si approfondisce proprio l'impatto e i benefici che il coinvolgimento dell'utente ha sullo sviluppo software e le conseguenze negative che una esperienza utente trascurata può avere per un prodotto o servizio.

Per determinare se effettivamente un prodotto sia valido, la tecnica migliore è quella di basarsi su fattori misurabili. Stabilendo chiaramente delle metriche si possono quantificare i progressi e apportare migliorie al prodotto durante il processo. Questo è valido anche per i prodotti software e in particolare per la relativa esperienza utente. Per questo motivo, nei Capitoli 2.4.1 e 2.4.2, vengono presentati gli strumenti comunemente utilizzati per la valutazione dell'esperienza utente. Nel primo capitolo vengono approfondite tre delle metodologie più diffuse, ovvero *test di usabilità*, *valutazione euristiche* e *gruppi di discussione*. Nel capitolo successivo viene posta l'attenzione su una tipologia particolare di test di usabilità, ovvero i questionari standardizzati e, per la precisione verranno esaminati i questionari SUPR-Q, SUS e UEQ.

Quelli citati sono i meccanismi standard utilizzati per la valutazione dell'esperienza utente e un nuovo approccio che integra queste tecniche e ne propone di nuove è il Design Thinking, una metodologia di problem solving che si centra sull'utente e che si focalizza sulla comprensione dei bisogni, dei desideri e dei comportamenti degli utenti finali con l'obiettivo di sviluppare soluzioni all'avanguardia e soddisfacenti.

Si tratta di un approccio iterativo e molto flessibile composto da cinque fasi principali: *raccolta ed empatia*, *definizione del problema*, *ideazione della soluzione*, *prototipazione* e *testing*. Queste fasi, con i relativi strumenti a disposizione, sono analizzate in maniera approfondita nel Capitolo 3.1.

Nonostante il Design Thinking abbia numerosi vantaggi e si sia dimostrato un approccio molto valido per lo sviluppo software in molte situazioni differenti, il suo impiego nei progetti è limitato da alcune difficoltà. Nel capitolo 3.3 verrà posta attenzione proprio sugli ostacoli percepiti dai team di sviluppo, fornendo alcune possibili soluzioni. Nel capitolo 3.4 saranno invece presentati alcuni esempi di applicazioni di successo del Design Thinking a progettazioni e riprogettazioni di prodotti software.

Per dimostrare le vere potenzialità del Design Thinking come metodologia di problem

solving si è pensato di attuare la riprogettazione di una famosa piattaforma dedicata allo streaming musicale, ovvero Spotify. L'idea della riprogettazione di questo servizio aveva come scopo principale quello di individuare suggerimenti non banali per il miglioramento dell'esperienza utente di una piattaforma che moltissime persone utilizzano ogni giorno e che, a primo impatto, potrebbe sembrare non avere problematiche. Grazie allo scambio di idee e al confronto, tipici del processo di Design Thinking, è stato però possibile far emergere problemi inesplorati dell'esperienza utente.

Seguendo i passi del Design Thinking si è cercato quindi di proporre soluzioni a questi problemi e di valutare, in seguito, il miglioramento percepito dalle persone. Le fasi dettagliate di *raccolta ed empatia*, *definizione del problema*, *ideazione della soluzione*, *prototipazione* possono essere esaminate nel Capitolo 4, mentre il Capitolo 5 è dedicato alla fase di *testing* in cui viene valutato il miglioramento percepito dall'utente nell'esperienza generale. I risultati ottenuti sono molto positivi e in linea con le aspettative.

Nel capitolo finale viene dato spazio alle conclusioni e possibili sviluppi futuri del Design Thinking, che bisogna precisare, essere un processo molto flessibile e orientato al futuro e quindi perfettamente adattabile a molti ambiti del settore informatico, come l'intelligenza artificiale e l'accessibilità.

Capitolo 2

Esperienza utente e Design Thinking

2.1 Metodologie popolari per lo sviluppo software

Nel contesto dello sviluppo software esistono svariati approcci e metodologie che possono essere seguiti per il raggiungimento dell'obiettivo finale, ovvero la realizzazione di un prodotto software che possa soddisfare le richieste del cliente.

Se fino a qualche anno fa la metodologia più utilizzata era quella Waterfall, attualmente l'approccio più sfruttato dalle aziende è quello Agile ed è interessante, a tal proposito analizzare i fattori che hanno portato a questa evoluzione.

La metodologia Waterfall è nata negli anni '70 e ha dominato il settore dello sviluppo software per molti anni. Tale approccio è noto per la sua rigidità, dovuta alla sua natura lineare e sequenziale. Esso prevede, infatti, un processo di sviluppo diviso in fasi ben distinte (raccolta dei requisiti, progettazione, implementazione, testing e manutenzione) e ogni fase deve essere completata prima che possa iniziare la successiva.

Tale metodologia non presenta però solo problemi legati alla rigidità, ma anche relativi allo scarso coinvolgimento del cliente e dell'utente finale. Questo sistema può risultare molto rischioso dato che prevede essenzialmente una pianificazione iniziale e soltanto in questa fase vengono considerate le opinioni del cliente.

Il problema è che spesso il cliente non ha le idee chiare fin dall'inizio e, durante il processo di sviluppo software, i requisiti tendono ad evolversi e cambiare. Un approccio come quello Waterfall non riesce ad adattarsi a questi continui cambiamenti e il prodotto finale potrebbe, di conseguenza, risultare non soddisfacente e non rispecchiare i reali bisogni del cliente. A lavoro concluso ogni cambiamento diventa oneroso, sia in termini di tempo che di denaro.

Durante la fine degli anni '90 le organizzazioni hanno cominciato a riconoscere i limiti dell'approccio Waterfall e di come essi potessero avere un effetto negativo sul prodotto finale. Avviene quindi intorno ai primi anni 2000 il passaggio da Waterfall ad Agile.

La metodologia Agile è stata sviluppata proprio come risposta ai problemi di rigidità e di scarso coinvolgimento del cliente e si tratta di un sistema iterativo e incrementale, in cui le esigenze del cliente sono al centro dell'attenzione.

Alla fine di ogni ciclo, che prevede le fasi di analisi dei requisiti, progettazione, sviluppo e testing, avviene una consegna del prototipo ottenuto con successivo feedback del cliente, in base al quale, durante il ciclo successivo, possono essere apportate modifiche e migliorie, con l'obiettivo di consegnare un prodotto finale che includa tutte le funzionalità e che sia soddisfacente per il cliente e di conseguenza per l'utente finale.

L'adozione dell'approccio Agile è stata graduale e ha richiesto tempo per essere accettata dalle organizzazioni. Tuttavia, negli ultimi anni, l'approccio Agile è diventato sempre più popolare e ampiamente utilizzato in tutto il settore dello sviluppo del software.

2.2 Usabilità, esperienza utente e interfaccia utente

Da questa evoluzione si capisce come, per le organizzazioni, sia diventato cruciale il coinvolgimento del cliente e in particolare dell'utente finale per la realizzazione di un prodotto che possa effettivamente avere successo sul mercato. A tal proposito è bene introdurre due nozioni cruciali su cui si baserà l'intera dissertazione, ovvero il concetto di *usabilità* e di *esperienza utente*.

Il concetto di *usabilità* è stato standardizzato per la prima volta nel 1990 con l'introduzione della norma ISO 9241 ed è ancora ampiamente usato per fornire una corretta definizione di usabilità. Questa norma definisce l'usabilità come *la capacità di un prodotto software di essere utilizzato da utenti specifici per raggiungere obiettivi specifici con efficacia, efficienza e soddisfazione in uno specifico contesto d'uso*. Nella ricerca di Rusu et al. [Rus+15] viene chiarito il significato di alcuni termini utilizzati nella norma:

- Utente: l'individuo che interagisce con il prodotto
- Efficacia: accuratezza e completezza con cui un utente riesce a raggiungere i propri obiettivi, senza deviazioni superflue o errori.
- Efficienza: risorse impiegate per il raggiungimento degli obiettivi in relazione all'accuratezza e alla completezza con cui essi sono stati raggiunti. Ad esempio, la quantità di compiti completati correttamente per unità di tempo.
- Soddisfazione: atteggiamento positivo nei confronti del prodotto
- Contesto d'uso: utenti, compiti, attrezzatura (hardware, software e materiali) e il contesto sociale in cui il prodotto viene utilizzato.

Durante gli anni '90 il significato centrale di usabilità ruotava quindi intorno ai suoi aspetti più utilitaristici; la valutazione del sistema software si basava sulle performance dell'utente e di quanto esse fossero efficaci ed efficienti.

La soddisfazione dell'utente era considerata direttamente proporzionale a questi due fattori e veniva, per questo, ignorata e messa in secondo piano.

La tecnologia però tendeva ad evolversi e rapidamente si passò da applicazioni che supportavano poche attività individuali a sistemi multi-task, più grandi e più flessibili. Nonostante questo però, la vera evoluzione nel concetto di usabilità arrivò alla fine degli anni '90, quando il concetto di usabilità catturò l'attenzione di professionisti non appartenenti all'ambito informatico, come sociologi, artisti e designer. Grazie a queste nuove prospettive e metodologie vennero conferite nuove dimensioni al tradizionale concetto puramente utilitaristico di usabilità.

Si espanse in questo modo il significato di usabilità e i professionisti furono incoraggiati a dedicare alla soddisfazione e alla qualità dell'esperienza utente lo stesso impegno che veniva dedicato alle misurazioni delle performance.

In questo frangente quindi avviene il passaggio storico da un concetto ristretto, utilitaristico e basato sulle performance come quello di usabilità ad un concetto molto più ampio, che è quello di esperienza utente. Questa naturale evoluzione avvenne di proprio quando si capì quanto fosse importante la componente emotiva nei prodotti software, in perfetta corrispondenza con il passaggio da metodologia Waterfall a metodologia Agile.

L'*esperienza utente* è quindi una nozione più complessa che riguarda l'esperienza umana nella sua totalità, comprendendo anche aspetti sociali ed emozioni, come piacere, appagamento, divertimento, fiducia e così via. Queste emozioni sono veicolate da fattori tra cui l'estetica visiva che, come viene affermato da Lavie et al. già nel 2004 [LT04], risulta essere un fattore determinante per la soddisfazione e il piacere degli utenti e , dopo quasi 20 anni, questa affermazione è più vera che mai.

Analizzato quindi il passaggio da usabilità a esperienza utente, è doveroso fare un approfondimento su quest'ultima, in modo da dargli una definizione più precisa. Il termine *esperienza utente* (user experience o UX) è stato coniato da Donald Norman nel 1995 [NMH95], ma, come già discusso, solo dal 2000 la UX ha iniziato ad essere riconosciuta come una pratica a sé stante.

Nel 2010, è stata introdotta la norma ISO 9241-210 che definisce l'esperienza utente come segue: *le percezioni e le reazioni della persona risultanti dall'uso di un sistema o servizio. Si ritiene che la UX includa tutte le emozioni, le convinzioni, le preferenze, le percezioni, le risposte fisiche e psicologiche, i comportamenti e le realizzazioni degli utenti che si verificano prima, durante e dopo l'uso del sistema*

Una delle note della norma afferma inoltre che la UX sia *una conseguenza dell'immagine del marchio, della presentazione, della funzionalità, delle prestazioni del sistema, del comportamento interattivo e delle capacità di assistenza del sistema interattivo, dello stato interno e fisico dell'utente derivante da precedenti esperienze, atteggiamenti, abilità e personalità e dal contesto di utilizzo.*

Per riassumere, quindi, l'esperienza utente è più ampia del semplice concetto di usabilità e si riferisce a tutti gli aspetti dell'interazione dell'utente con un prodotto o un sistema, inclusi gli aspetti emotivi e soggettivi.

Come viene affermato nella ricerca di Rusu et al. [Rus+15], per più di 30 anni il concetto di usabilità è stato discusso e ridefinito da esperti, professionisti e note organizzazioni ma non si è ancora giunti ad una unica e chiara definizione globalmente accettata. In quasi tutte le definizioni, anche se con nomi diversi, vengono comunque citati tre fattori fondamentali che ne stanno alla base, ovvero l'efficienza, l'efficacia e la soddisfazione. Stessa sorte tocca anche alla definizione di esperienza utente, anche se si tratta di un concetto più recente.

Per concludere la panoramica sui concetti più importanti, vale la pena definire anche il concetto di *interfaccia utente*. L'interfaccia utente (UI) è la componente grafica di un sistema che permette all'utente di interagire con esso. Essa include tutti gli elementi che permettono di mettere in comunicazione l'utente e il sistema, come pulsanti, menu, caselle di testo, icone e così via. Come afferma Norman et al. [NN21b], è importante distinguere l'esperienza utente totale dall'interfaccia utente che ne è solo una sua componente estremamente importante che contribuisce insieme ad altri fattori all'esperienza totale.

Per quanto riguarda il rapporto che intercorre tra questi elementi, l'interfaccia utente è parte integrante dell'esperienza utente e solitamente l'esperienza utente è considerata un'estensione dell'usabilità e, nonostante abbiano due significati ben diversi, questi due termini vengono spesso interscambiati.

2.3 Il ruolo fondamentale della UX nello sviluppo software

L'utente finale risulta un tassello molto importante da considerare durante il processo di sviluppo software per varie ragioni e il suo coinvolgimento ha molti benefici tra cui:

- Possibilità di comprendere al meglio le sue esigenze in modo da poter progettare e sviluppare un prodotto che sia utile e soddisfacente. Questo significa che il prodotto finale sarà più orientato all'utente e meno centrato sulle esigenze del team di sviluppo e ciò incrementa le probabilità di successo.
- Raccolta continua di feedback che permettono al team di sviluppo di apportare correzioni al progetto durante il suo sviluppo, ovvero quando l'attuazione delle modifiche non è molto costosa, piuttosto che alla fine del processo quando il lavoro è già concluso e ogni modifica risulterebbe molto onerosa.
- Miglioramento della percezione del brand e del team di sviluppo. Gli utenti, infatti, apprezzano le aziende che sanno ascoltare e che prendono in considerazione le loro opinioni ed esigenze, creando di conseguenza un vero rapporto di fiducia.

Ciò che si vuole ottenere analizzando l'utente e le sue abitudini è quindi una user experience piacevole, intuitiva, efficiente. Il fruitore del software deve essere in grado di svolgere le sue attività senza incontrare ostacoli e raggiungere i suoi obiettivi in modo efficace e soddisfacente.

La fase di sviluppo della user experience si concentra proprio sullo studio e progettazione dell'esperienza che le persone avranno utilizzando il sistema.

Se la prima impressione che l'utente ha di un software è negativa, il sentimento di insoddisfazione condiziona la sua interazione con il software e porterà l'utente a prestare ancora più attenzione a tutti i problemi del sistema, con conseguente abbandono, anche prima di aver considerato l'offerta. La stessa cosa vale per una prima impressione positiva che condiziona in meglio tutta l'esperienza e l'utente sarà più indulgente con eventuali problemi di usabilità.

Consideriamo come esempio i siti web e il concetto di bounce. Con il termine bounce facciamo riferimento a quando l'utente abbandona il sito dopo aver preso visione di una singola pagina entro pochi secondi. Il bounce rate di un sito è il numero di visitatori che visitano una sola pagina del sito diviso per il numero complessivo di visitatori. Un bounce rate alto è sintomo di disinteresse degli utenti nei confronti del sito web e se la UX risulta poco curata o accattivante il risultato potrebbe essere proprio questo.

In conclusione si può quindi affermare che una user experience trascurata può avere conseguenze negative per un prodotto o servizio, riducendo la soddisfazione dell'utente, aumentando l'abbandono del prodotto o servizio, riducendo la fidelizzazione del cliente e portando a una riduzione dei profitti. Per evitare queste conseguenze, è importante dedicare attenzione e risorse alla progettazione dell'esperienza utente.

Nella ricerca di Rusu et. al [Rus+15] si sottolinea in particolare quanto sia cruciale educare i professionisti nell'ambito dello sviluppo software all'importanza dell'esperienza utente e a come svilupparla correntemente. Oggigiorno, infatti, il concetto di Human-Computer Interaction (interazione uomo-computer) è ancora messo in secondo piano e spesso considerato un argomento non pertinente all'informatica, nonostante il fatto che molti studi affermino che, nei progetti software, almeno il 50% del codice scritto è dedicato alla realizzazione dell'interfaccia utente [Dar+22].

2.4 Misurare l'esperienza utente: modelli e strumenti

Il modo migliore per determinare se un prodotto sia di valido è quello di basarsi su fattori misurabili. Stabilendo chiaramente una metrica si possono quantificare i progressi e apportare migliorie durante il processo, in modo da ottenere il risultato desiderato. Senza obiettivi precisi il rischio è quello di rimanere bloccati in uno stato di supposizione.

Questa regola vale anche per i prodotti software e in particolare per la realizzazione dell'esperienza utente che è uno degli aspetti dei software più difficile da valutare, dal momento che non esiste un'unica metodologia o una procedura consolidata.

L'analisi metodica dell'esperienza utente permette non solo di determinare le aree di miglioramento, ma anche di mettere a confronto il proprio prodotto con quello dei competitors presenti sul mercato [SHT14].

L'obiettivo di questa sezione è quindi quella di delineare le metodologie di valutazione dell'esperienza utente e approfondire alcuni modelli comunemente usati nell'ambiente.

2.4.1 Metodologie di valutazione dell'esperienza utente dei software

Esistono diversi approcci e metodologie per la valutazione dell'usabilità del software. Tra le più comuni si possono trovare i test di usabilità, le valutazioni euristiche e i gruppi di discussione.

I **test di usabilità** sono una metodologia di valutazione dell'usabilità che implica la raccolta di dati e l'osservazione degli utenti mentre interagiscono con il software. L'obiettivo è quello di identificare i problemi relativi alla user experience. Esistono molteplici test di usabilità che sfruttano strumenti diversi e che possono essere combinati tra loro per ottenere una valutazione dell'esperienza utente complessiva più accurata:

- **Registrazione video:** consente di registrare l'interazione degli utenti con il software, in modo da poterla analizzare successivamente.
- **Think-aloud protocol:** agli utenti viene chiesto di esprimere ad alta voce i loro pensieri e le loro sensazioni durante l'interazione con il prodotto.
- **Misurazione del tempo di completamento delle attività e tasso di successo:** questa tecnica permette di calcolare la velocità con cui gli utenti riescono a completare le attività assegnate, identificando di conseguenza ostacoli o problemi nell'usabilità. Il tasso di successo indica la percentuale di utenti che sono riusciti a completare con successo le attività assegnate.
- **Heatmap:** rappresentazione visiva dell'interazione degli utenti con il software, in cui le aree che vengono cliccate più di spesso sono messe in evidenza con colori caldi (rosso e arancione). In figura 2.1 si può vedere un esempio di heatmap relativo ad una semplice ricerca sul internet, ma questa tecnica può essere utilizzata per qualsiasi sito web o applicazione.
- **Questionari standardizzati:** esistono molti questionari che possono essere utilizzati per valutare l'esperienza degli utenti e le loro percezioni sul software. Alcuni esempi sono SUS, SUMI, HEART, NPS, UEQ che saranno oggetto di un'analisi più approfondita nella prossima sezione.

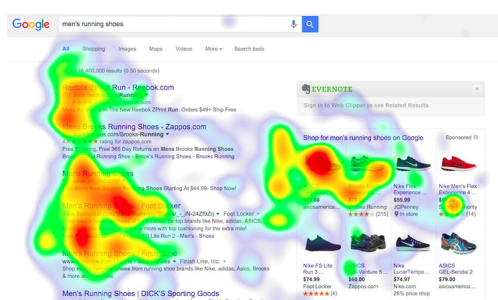


Figura 2.1: Esempio di heatmap

La **valutazione euristica** punta al coinvolgimento di un gruppo di esperti di usabilità che hanno il compito di valutare il software sfruttando euristiche, ovvero linee guida che aiutino a identificare problemi. Tra le euristiche più comuni ci sono le *10 euristiche di Nielsen*, sviluppate dall'informatico e pioniere dell'usabilità software, Jakob Nielsen. Nielsen ha introdotto questo metodo di valutazione euristica composto da 10 principi utilizzati da esperti per rilevare le problematiche nell'usabilità di un prodotto software e le loro gravità (estetici, minori, maggiori, gravi). In figura 2.2 sono riportati i 10 principi su cui si basa questo approccio.

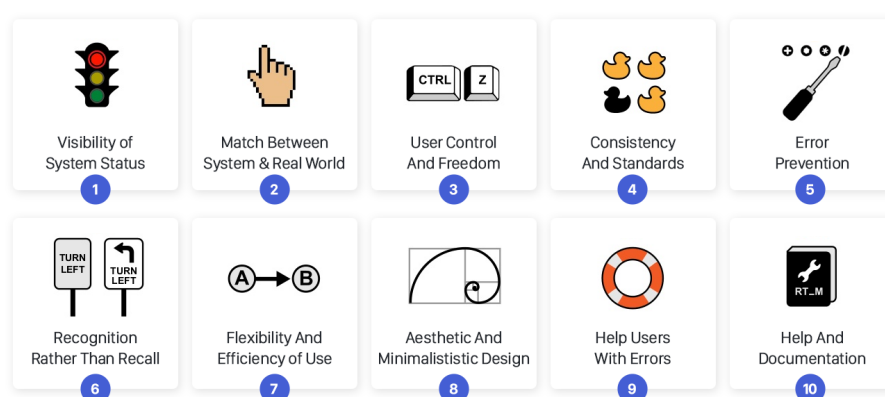


Figura 2.2: Le 10 euristiche di Nielsen

Un esempio di applicazione di questa valutazione euristica ci viene fornita da Bou-raghi et al. [Bou+22]. La ricerca ha come obiettivo quello di valutare e identificare i problemi di usabilità nel sistema di sanitario nazionale chiamato AVAB, sfruttando proprio le *10 euristiche di Nielsen*. Tre valutatori esperti sono stati coinvolti nella ricerca e l'analisi ha portato alla luce un totale di 86 problemi e più della metà dei quali sono stati categorizzati come maggiori o gravi.

La maggior parte dei problemi rilevati ricadevano nel decimo principio "Aiuto e Documentazione", denotando una notevole mancanza di aiuto laddove sarebbe servito. Gli

altri problemi gravi facevano riferimento alle categorie di "flessibilità ed efficienza nell'utilizzo" in particolare sottolineando come i tasti funzione non siano organizzati in gruppi logici, e di "riconoscimento piuttosto che richiamo", questo significa che all'utente era spesso chiesto di ricordare informazioni e il sistema non ne permette la visualizzazione in modo facile quando servono.

La maggior parte dei problemi rilevati sono però di facile risoluzione, quindi spesso con delle semplici accortezze e rapide modifiche è possibile ottenere un'esperienza utente molto più robusta e soddisfacente .

Nello studio viene sottolineato come questo metodo abbia numerosi vantaggi tra cui il risparmio di tempo e di risorse economiche, dato che prevede il coinvolgimento di un ristretto numero di esperti che possono eseguire la valutazione in modo rapido. Inoltre, si tratta di un approccio semplice, efficiente ed essendo eseguito da esperti e standardizzato, i risultati possono essere considerati affidabili.

Rispetto ad altri metodi di valutazione dell'usabilità, come i test di usabilità, la valutazione euristica può far emergere un numero più alto di problematiche.

Per concludere questa sezione è doveroso menzionare i **gruppi di discussione**, spesso noti anche come focus group. In questo caso è prevista la discussione sul tema dell'usabilità del team di sviluppo con un gruppo di fruitori del software che rappresentino, a livello demografico, il pubblico target.

I partecipanti hanno il compito di esprimere impressioni, emozioni e perplessità e in generale fornire feedback sulla loro esperienza con il software. Questo approccio risulta molto utili quando si deve sviluppare un nuovo prodotto o servizio, in quanto permettono di comprendere le necessità e preferenze degli utenti, facilitando la raccolta dei requisiti.

I gruppi di discussione sono tuttavia considerati la metodologia più debole tra quelle citate dato che le informazioni raccolte potrebbero essere condizionate da numerosi fattori. Le dinamiche che si creano all'interno del gruppo, esperienze passate, l'importanza del marchio e le modalità con cui il gruppo viene guidato potrebbero influenzare i partecipanti. Per questo bisogna creare un ambiente di discussione rilassato, in cui a tutti è concesso uno spazio equo. Inoltre è importante che il moderatore lavori in modo oggettivo senza condizionare i partecipanti. Si consiglia in ogni modo di combinare questo approccio con uno più standardizzato per avere risultati attendibili su cui basarsi.

2.4.2 Modelli standard di misurazione dell'esperienza utente

Tra i test di usabilità visti nella sezione precedente sono stati citati i questionari standardizzati. La loro popolarità è determinata da vari aspetti e in particolare dalla possibilità di raccogliere in modo pratico e rapido feedback riguardo al prodotto sotto analisi.

Secondo lo studio condotto da J. Sauro [Sau15] un valido questionario dovrebbe essere:

- Generalizzabile: un buon questionario dovrebbe avere sufficienti dimensioni per poter descrivere in modo esaustivo le qualità di un prodotto software, ma d'altro canto non dovrebbe essere troppo specifico da non poterlo applicare ad altri tipi di prodotti. Ad esempio, un sito informativo differisce da un sito di e-commerce, ma lo stesso questionario dovrebbe essere applicabile ad entrambi per poter essere considerabile valido.
- Multidimensionale: il questionario dovrebbe riuscire ad analizzare tutti gli aspetti principali del software in analisi.
- Breve: il tempo è denaro e spesso i partecipanti non sono disposti a spenderne troppo per la compilazione di un questionario.
- Normalizzato: il questionario dovrebbe essere supportato da un database normalizzato, cioè privo di ridondanze e incoerenze. In questo modo è possibile capire chiaramente in quale ambito il prodotto ha ottenuto un determinato punteggio. Se il questionario è normalizzato è possibile anche fare dei confronti più accurati con i competitors diretti.

L'obiettivo della ricerca di J. Sauro era quello di creare un questionario standardizzato che presentasse tutte queste caratteristiche e che avesse come obiettivo quello di valutare la soddisfazione complessiva dell'utente. Il risultato dello studio è un questionario chiamato **SUPR-Q**, composto da otto domande, consultabili in Figura 2.3 divise in quattro sezioni: usabilità, fiducia, aspetto e fedeltà.

# Questions	Strongly Disagree				Strongly Agree
1 This website is easy to use.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
2 It is easy to navigate within the website.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3 The information on the website is credible.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4 The information on this website is trustworthy.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5 I feel comfortable purchasing from this website.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6 I found the website to be attractive.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7 The website has a clean and simple presentation.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
8. How likely are you to recommend this website to a friend?					
Not at all likely		Neutral		Extremely Likely	
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Figura 2.3: Il questionario SUPR-Q

Il **System Usability Scale (SUS)** è stato sviluppato da John Brooke [Bro96] tra la fine degli anni '80 e l'inizio degli anni '90. Al momento della scrittura, il modello SUS risulta menzionato da più di 5000 pubblicazioni e si conferma uno dei questionari più utilizzati nell'ambiente, come già affermavano J. Sauro in una pubblicazione del 2011 [SL11] e Borsci et al. in una ricerca del 2015 [Bor+15].

Il modello SUS è un metodo di valutazione dell'usabilità dei prodotti software e viene definito *tecnologicamente neutrale*, dal momento che può essere facilmente applicato a svariate tipologie di software (siti web, software per il business, sistemi operativi e così via). SUS ha dimostrato il suo valore negli anni, in particolare per essersi adattato facilmente al travolgente progresso tecnologico cominciato negli anni '90 e caratterizzato dalla continua nascita di nuove tecnologie. SUS è stato infatti concepito prima che il termine sito web venisse coniato, eppure risulta perfetto per valutarne l'usabilità.

Nasce inizialmente con l'obiettivo di creare un modello che misurasse le tre componenti che, secondo il modello ISO 9241-11, determinano una buona usabilità: efficacia (se le persone riescono a completare i propri compiti), efficienza (quante risorse vengono spese per raggiungere i propri obiettivi), soddisfazione (quanto è appagante l'esperienza).

L'approccio SUS è composto da 10 brevi affermazioni, consultabili in figura 2.4, e fornisce in output un unico valore complessivo (modello unidimensionale) che valuta l'usabilità. Si tratta, in particolare, di una scala Likert in cui le affermazioni sono state scelte con giudizio e approfondite ricerche. Una delle caratteristiche più interessanti è il fatto che le affermazioni dispari hanno un'accezione positiva, mentre quelle pari hanno un'accezione negativa. Questa scelta è stata fatta in modo da evitare bias nelle risposte; infatti, data la natura rapida del questionario, il rischio era che gli intervistati non leggessero attentamente i dieci punti e rispondessero in modo approssimativo. Con questo espediente il lettore è portato a concentrarsi e capire ognuna delle affermazioni.

Gli intervistati devono dare una valutazione da 1 (*fortemente in disaccordo*) a 5 (*fortemente d'accordo*) ad ogni punto.

Per il calcolo del punteggio finale, il valore dato ad ogni affermazione viene mappato in una scala che va da 0 a 4 in cui i numeri alti indicano una buona percezione dell'usabilità. I valori convertiti vengono sommati e la somma viene moltiplicata per 2.5, ottenendo in questo modo un valore compreso tra 0 (*pessimo*) e 100 (*ottimo*) [Bro96].

	Strongly disagree				Strongly agree
1. I think that I would like to use this ballot frequently.	1	2	3	4	5
2. I found the ballot unnecessarily complex.	1	2	3	4	5
3. I thought the ballot was easy to use.	1	2	3	4	5
4. I think that I would need the support of a poll official to be able to use this system.	1	2	3	4	5
5. I found the various parts of this ballot were well integrated.	1	2	3	4	5
6. I thought there was too much inconsistency in this ballot.	1	2	3	4	5
7. I would imagine that most people would learn to use this ballot very quickly.	1	2	3	4	5
8. I found the ballot very awkward to use.	1	2	3	4	5
9. I felt very confident using the ballot.	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this ballot.	1	2	3	4	5

Figura 2.4: Le 10 affermazioni del modello SUS [Bro96]

Anche se il modello SUS è stato pensato come un modello unidimensionale, negli anni varie ricerche ne hanno adottato una versione bidimensionale, basata sul fatto che le 10 affermazioni potessero essere raggruppate in due gruppi: usabilità e apprendibilità [Bor+15].

Nello studio condotto da J. Sauro [SL11] viene fatta una rassegna delle qualità positive e negative del modello SUS. Il modello è riassumibile attraverso i seguenti aggettivi:

- Affidabile: gli utenti rispondono in modo coerente e il modello è in grado di rilevare differenze tra i prodotti anche con piccoli gruppi utenti e informazioni ridotte.
- Valido: misura con precisione ciò per cui è stato concepito
- Non diagnostico: non permette di individuare i problemi ma solo di valutare l'esperienza complessiva

Uno dei limiti che è bene sottolineare di questo approccio è la difficoltà per intervistati non di madrelingua inglese di cogliere correttamente le sfumature semantiche delle affermazioni [Fin06]. Ci sono stati negli anni tentativi di traduzione in altre lingue ma nessuno

di essi ha avuto successo, per questo si è optato, in alcune ricerche, di sostituire parole troppo complicate come *cumbersome*, con parole più facili come *awkward* (domanda 8).

Un altro questionario che è opportuno menzionare è il modello **User Experience Questionnaire (UEQ)**. Prodotto in Germania nel 2005, si tratta di un modello che si concentra sull'analizzare l'esperienza utente nella sua complessità. Il test è composto da 26 punti formati da coppie di caratteristiche in antitesi tra loro che il prodotto può avere e hanno la seguente forma:

attraente ○○○○○○○○ non attraente

Per ognuno di essi si può quindi fornire una valutazione che va da -3 a +3, valori che rappresentano rispettivamente l'estremo negativo e l'estremo positivo. La valutazione intermedia 0 equivale ad una risposta neutrale. Metà dei punti, in ordine casuale, iniziano con il termine negativo e metà con il termine positivo, sempre con l'obiettivo di non raccogliere risposte bias. Con una valutazione come la seguente si dichiara di stimare il prodotto più *attraente* che *non attraente*

attraente ○○○○○○Ⓢ○ non attraente

Le scale di valutazione in cui si dividono i 26 punti sono 6, ovvero:

1. Attrattività: impressione generale del prodotto.
2. Apprendibilità: è facile prendere familiarità con il prodotto e capire come va utilizzato?
3. Efficienza: l'utente riesce a svolgere le proprie attività senza particolari sforzi?
4. Controllabilità: l'utente sente di avere il controllo durante l'interazione con il prodotto?
5. Stimolazione: l'uso del prodotto è stimolante e motivante?
6. Originalità: il prodotto risulta innovativo e creativo? Riesce ad attirare l'attenzione dell'utente?

Attrattività si basa sulla pura percezione che gli utenti hanno sull'aspetto generale del prodotto, che può essere gradevole o sgradevole. *Apprendibilità*, *efficienza* e *controllabilità* sono scale pratiche, dette anche goal-directed, mentre *stimolazione* e *originalità* sono scale astratte, o not goal-directed [SHT17]. Di seguito, in figura 2.6, lo scheletro di UEQ [Rau+13] e il test nella sua interezza sia in inglese che in italiano.

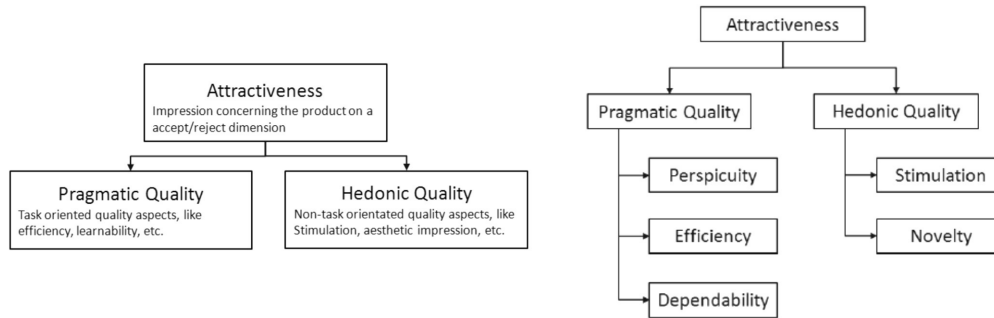


Figura 2.5: Struttura delle 6 scale base di UEQ

	1	2	3	4	5	6	7		
annoying	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	enjoyable	1
not understandable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	understandable	2
creative	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	dull	3
easy to learn	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	difficult to learn	4
valuable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	inferior	5
boring	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	exciting	6
not interesting	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	interesting	7
unpredictable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	predictable	8
fast	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	slow	9
inventive	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	conventional	10
obstructive	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	supportive	11
good	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	bad	12
complicated	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	easy	13
unlikable	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	pleasing	14
usual	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	leading edge	15
unpleasant	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	pleasant	16
secure	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	not secure	17
motivating	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	demotivating	18
meets expectations	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	does not meet expectations	19
inefficient	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	efficient	20
clear	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	confusing	21
impractical	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	practical	22
organized	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	cluttered	23
attractive	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	unattractive	24
friendly	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	unfriendly	25
conservative	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	innovative	26

	1	2	3	4	5	6	7		
fastidioso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	piacevole	1
incomprensibile	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	comprensibile	2
creativo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	privo di fantasia	3
facile da apprendere	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	difficile da apprendere	4
di grande valore	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	di poco valore	5
noioso	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	appassionante	6
non interessante	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	interessante	7
imprevedibile	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	prevedibile	8
veloce	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	lento	9
originale	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	convenzionale	10
ostruttivo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	di supporto	11
buono	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	scarso	12
complicato	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	facile	13
repellente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	attraente	14
usuale	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	moderno	15
sgradevole	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	gradevole	16
sicuro	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	insicuro	17
attivante	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	soporifero	18
conforme alle aspettative	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	non conforme alle aspettative	19
inefficiente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	efficiente	20
chiaro	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	confuso	21
non pragmatico	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	pragmatico	22
ordinato	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	sovaccarico	23
invitante	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	non invitante	24
congeniale	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	ostile	25
conservativo	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	innovativo	26

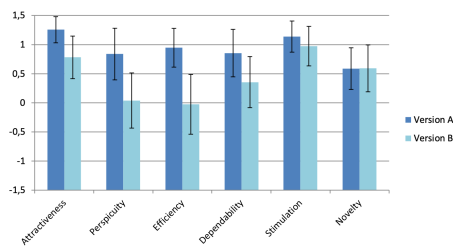
Figura 2.6: Versione in inglese e in italiano del test UEQ

Nel manuale fornito con il questionario [Sch19] sono riportati tipici scenari in cui il sistema UEQ può essere utilizzato.

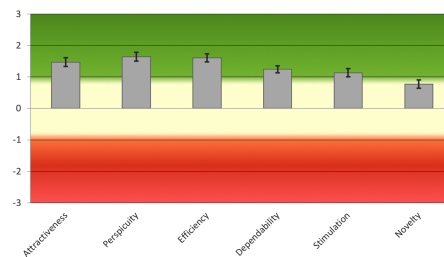
- Confronto della user experience di due prodotti: solitamente i software vengono sottoposti a vari redesign durante la loro vita. La versione di partenza viene migliorata di volta in volta in base ai feedback ricevuti da clienti e utenti. Grazie ad un questionario come UEQ è possibile realizzare un confronto tra la versione vecchia e quella nuova, in modo da capire se c'è effettivamente stato un miglioramento

nell'esperienza utente. La stessa idea può essere applicata al confronto tra due prodotti diversi; infatti, essendo UEQ un questionario standardizzato, è possibile fare una comparazione tra due prodotti differenti per capire in quali aspetti uno è migliore rispetto all'altro. Esempio di un confronto tra due ipotetici prodotti può essere osservato nella figura 2.7a .

- Verificare che un prodotto abbia una user experience adeguata: Ogni scala può essere stimata da -3 (molto negativo) a +3 (molto positivo). In base a tale valore è possibile valutare la qualità di ogni scala. Un punteggio compreso tra -0.8 e +0.8 è considerato neutrale, sopra positivo e sotto negativo. Esempio di una valutazione molto positiva di un prodotto può essere osservata nella figura 2.7b .
- Determinare le aree di miglioramento: con un sistema quantitativo come quello UEQ non è possibile determinare in modo specifico gli aspetti che devono essere migliorati, ma dà la possibilità comunque restringere il campo, identificando le aree che hanno bisogno di essere migliorate. Abbinando la raccolta quantitativa del questionario ad una raccolta di qualitativa è possibile avere risultati più accurati.



(a) confronto



(b) valutazione

Il questionario, accompagnato da una guida sul suo utilizzo e un documento Excel per l'analisi dei dati, è disponibile gratuitamente sul sito www.ueq-online.org. Per via di sfumature semantiche è sempre preferibile compilare il questionario nella propria lingua, per questo sul sito è possibile trovare il modello tradotto in numerose lingue, tra cui l'italiano. Il completamento del questionario richiede usualmente 3-5 minuti e l'analisi dei dati può essere fatta in modo efficiente grazie al documento Excel.

In conclusione, in questo capitolo sono stati passati in rassegna tre diversi questionari utilizzati comunemente per la valutazione dell'esperienza utente e dell'usabilità, descrivendone caratteristiche, scale di valutazione e i loro possibili utilizzi nella pratica.

Da questa analisi è chiaro come tutti e tre siano validi strumenti e che, a seconda delle necessità della singola ricerca, può esserne scelto uno piuttosto che un altro o anche combinati tra loro per avere risultati più completi.

Le accortezze emerse da tenere in considerazione sono in particolare due: somministrare agli intervistati questionari nella loro lingua madre se possibile e combinare la

tecnica di raccolta di dati quantitativa dei questionari con raccolte di dati qualitative, ad esempio organizzando focus group, in modo da quantificare la validità dell'esperienza utente ma anche identificare eventuali problemi da migliorare. Altri questionari standardizzati utilizzati sono: SUMI, HEART e NPS.

Quelli citati sono i meccanismi standard utilizzati per la valutazione dell'esperienza utente e un nuovo approccio che integra queste tecniche e ne propone di nuove è il Design Thinking, che verrà analizzato in maniera approfondita nel prossimo capitolo.

Capitolo 3

Il Design Thinking: un approccio innovativo

Negli ultimi anni si sono diffuse nuove metodologie di sviluppo software che ruotano totalmente intorno all'utente e alle sue esigenze e che dovrebbero essere integrate agli attuali processi di sviluppo, tra cui il Design Thinking.

Il Design Thinking è un approccio che si centra sull'utente e che si focalizza sulla comprensione dei bisogni, dei desideri e dei comportamenti degli utenti finali con l'obiettivo di sviluppare soluzioni all'avanguardia e soddisfacenti. Si tratta di una metodologia che combina l'empatia, la creatività e il pensiero analitico per risolvere problemi complessi.

Questa metodologia è stata originariamente sviluppata nell'ambito del design, ma è stata adottata in molti altri settori, tra cui lo sviluppo software; infatti, grazie proprio alla sua capacità di creare soluzioni orientate all'utente, adattabili, flessibili e innovative, permette di progettare software completi e con esperienze utente curate.

3.1 Fasi e strumenti utilizzati

Nella ricerca condotta da Darmawan et al. [Dar+22] vengono illustrati in modo chiaro e dettagliato le fasi principali che costituiscono il Design Thinking. Uno schema riassuntivo di tali fasi tratto dalla ricerca è mostrato nell'immagine 3.1.

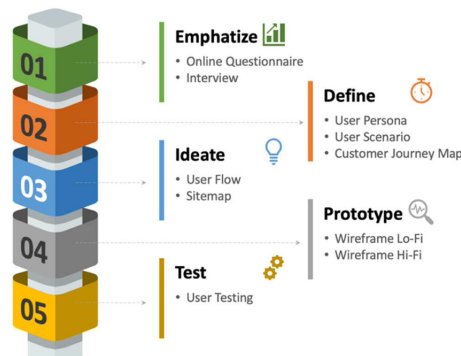


Figura 3.1: Fasi caratteristiche del Design Thinking

1. **Raccolta e Empatia:** in questa fase iniziale viene svolta una ricerca approfondita sugli utenti in modo da capire esigenze e definire i requisiti iniziali del progetto. Durante questa fase il team di ricerca deve empatizzare con il proprio campione cercando di capire sentimenti, bisogni ed eventuali problemi nell'interazione con il prodotto. Spesso vengono utilizzati strumenti come questionari online, un esempio è il User Experience Questionnaire (UEQ) che verrà approfondito successivamente.

L'output di questa fase è una Empathy Map, ovvero uno strumento visuale che permette di rappresentare tutto quello che sappiamo su un particolare utente, evidenziando in particolare quattro aspetti dell'esperienza utente: quello che l'utente nel complesso dice, pensa, fa e prova.

2. **Definizione:** la fase di define è la prima vera e propria fase de Design Thinking. In base ai dati raccolti nella fase preliminare, vengono definiti degli user persona, ovvero una raccolta di dati demografici, comportamentali, motivazionali ma anche influenze e obiettivi, con lo scopo di identificare i prototipi di persone che utilizzeranno il servizio una volta completato. Oltre agli user persona vengono definiti anche le customer journey map, cioè il percorso che un cliente compie durante l'interazione con il prodotto e gli user scenario, ovvero brevi racconti che descrivono un ipotetico utilizzo del prodotto e sono utili per identificare eventuali punti di difficoltà o di miglioramento.
3. **Ideazione:** in questa fase si raccolgono le idee ottenute durante la fase precedente e dal successivo il brainstorming. Ne deriverà una user flow map e una sitemap.

Il primo rappresenta il percorso che un utente segue per raggiungere un obiettivo, mentre il secondo delinea visivamente l'architettura del servizio, elencando tutte le pagine e mostrandone collegamenti e struttura gerarchica.

4. Prototipazione: durante la fase di prototipazione l'obiettivo è quello di creare i prototipi delle soluzioni ideate durante la fase di ideazione, per testare la fattibilità e la validità delle soluzioni prima dell'implementazione. Ci sono due sotto fasi, ovvero:
 - (a) Wireframe a bassa fedeltà, o low fidelity: durante questa fase l'obiettivo è quello di creare un prototipo semplice e senza troppi dettagli. Spesso viene realizzato a mano e aiuta a visualizzare gli aspetti essenziali come bottoni, testi, griglie, elementi per la navigazione e così via. Risulta utile per avere una prima idea dell'efficacia delle soluzioni pensate in modo rapido ed economico
 - (b) Wireframe ad alta fedeltà, o high fidelity: in questa fase ci si concentra invece sulla creazione di prototipi più dettagliati e avanzati, creati con software appropriati. Questi prototipi contengono maggiori dettagli, come colori, immagini e funzionalità avanzate. Per la loro realizzazione è ovviamente richiesto un maggiore investimento di risorse, ma grazie alla fase precedente ogni spreco dovrebbe essere ridotto al minimo.
5. Test: Nella fase finale di testing verranno valutati i risultati ottenuti dalla fase di prototipazione. Un gruppo di utenti verrà scelto in modo che possano rispecchiare tutte le user persona definite nella fase 2a e verranno coinvolte per ottenere feedback sul prodotto.

Il processo di Design Thinking è iterativo, il che significa che le fasi possono essere ripetute più volte fino a quando non si arriva a una soluzione soddisfacente per gli utenti finali. Si tratta anche di un processo molto flessibile, infatti retrocedere alle fasi precedenti risulta poco dispendioso. Una volta ottenuto un risultato appagante si può procedere con l'implementazione.

Un altro aspetto importante da sottolineare quando si parla di Design Thinking è il suo carattere fortemente collaborativo. Questo tipo di approccio richiede un team di lavoro multidisciplinare e i suoi componenti devono essere in grado di lavorare insieme, combinando le loro competenze e prospettive per trovare soluzioni innovative ai problemi.

Per concludere questa sezione sul Design Thinking è doveroso menzionare il suo orientamento al futuro. Si tratta, infatti, di una metodologia che guarda al futuro e punta a creare soluzioni che non solo siano efficaci ma che durino nel tempo. Per questo motivo è richiesta una buona comprensione delle tendenze e capacità di adattamento ai rapidi cambiamenti del mercato e delle esigenze degli utenti.

3.2 Vantaggi, sfide e raccomandazioni

Come abbiamo già sostenuto nei paragrafi precedenti, il Design Thinking va a supporto dello sviluppo software, consentendo di capire al meglio le necessità degli utenti, dando anche lo stimolo di creare prodotti moderni, con funzionalità sempre nuove; infatti, come afferma Bhowmik et al. [Bho+15], una delle sfide più impegnative dell'industria del software è proprio la creazione di prodotti innovativi che permettano di distinguersi in un mercato ormai saturo e di raggiungere un vantaggio competitivo.

Lo studio di Pereira et al. [Per+21] condotto nel 2021 riesce a mettere in luce i vantaggi e le sfide dell'applicazione del Design Thinking allo sviluppo software. In particolare, lo studio raccoglie le opinioni di 39 professionisti con background lavorativi diversi in merito ai benefici e alle sfide percepite nell'utilizzo del Design Thinking per la definizione dei requisiti. Tra i benefici si trovano:

- Migliore comprensione delle necessità e facilitazione nel raccoglimento delle informazioni: ciò si riflette in una più accurata definizione dei requisiti.
- Incoraggiamento alla collaborazione e allo scambio di idee.
- Aggiunta di valore: il coinvolgimento dell'utente nel processo di definizione dei requisiti apporta un valore aggiunto al prodotto.
- Evoluzione della mentalità: il Design Thinking permette di cambiare il punto di vista con cui percepiamo i problemi, permettendo un approccio nuovo e la creazione di soluzioni innovative.

Alcune raccomandazioni che vengono suggerite per sfruttare il reale potenziale del Design Thinking sono: il coinvolgimento di diversi ruoli nel team di sviluppo in modo da ottenere più punti di vista professionali e un'analisi approfondita della fascia di mercato.

Malgrado ciò, vari studi affermano che siamo ancora in una fase in cui la maggior parte dei prodotti realizzati dalle compagnie software non soddisfa le reali necessità degli utenti [Ard+14]. Questo potrebbe essere dovuto dal fatto che, molto spesso, con l'obiettivo di risparmiare tempo e risorse, le aziende e i provider di servizi software sottovalutano l'importanza di uno studio sistematico dell'utente, limitandosi solo a ipotizzare i loro bisogni, e la rilevanza che i feedback forniti dagli utenti possono avere.

Ci si potrebbe dunque domandare perché un approccio come quello del Design Thinking, nato proprio per analizzare in modo più opportuno gli utenti e le loro necessità, non venga integrato nel processo di sviluppo delle aziende. Come già accennato, il lavoro svolto da Pereira et al. [Per+21] cerca di definire anche le sfide che i professionisti si trovano a dover affrontare nell'applicazione del Design Thinking. I risultati sono:

- Scarsa creatività e possibilità di utilizzare soluzioni "preconfezionate".

- Limitata conoscenza e sottovalutazione dell'importanza del Design Thinking.
- Mancanza di tempo e di risorse economiche.
- Barriere culturali e resistenza al cambiamento.

Vari professionisti hanno infatti sottolineato che una delle difficoltà principali è quella di collaborare con manager e coordinatori che sono spesso restii alla cooperazione con i team e che preferiscono prendere le decisioni in modo indipendente, senza ascoltare le opinioni del team e men che meno quelle degli utenti finali.

Tra i suggerimenti per ovviare a queste problematiche ci sono: l'utilizzo di "rompighiaccio e giochi" per favorire la creatività (come brainstorming e storytelling), incentivare la formazione del team sui temi di Design Thinking e UX attraverso corsi, workshop e materiale formativo, raccogliere e condividere i risultati positivi ottenuti durante lo sviluppo in modo da conferire valore e importanza al processo.

3.3 Design Thinking: esempi di applicazione e successo

Il Design Thinking ha la potenzialità di essere un'importante fonte di innovazione e miglioramento nell'esperienza utente. In questa sezione, verranno riportati alcuni casi di successo in cui è stato applicato questo approccio per risolvere i problemi dell'utente e migliorare, di conseguenza, la sua esperienza. Attraverso questi esempi concreti, potranno essere messi in evidenza gli aspetti più importanti e suggerimenti relativi all'applicazione del Design Thinking.

Il **primo caso di studio** è stato svolto da Darmawan et al. [Dar+22] nel 2022 e aveva come obiettivo quello di progettare SIMAK WEB e SIMAK MOBILE come supporto al sistema di informazione dell'Università di Siliwangi.

La ricerca segue diligentemente le fasi caratteristiche del Design Thinking, iniziando con un'analisi del portale web già esistente mediante questionario UEQ distribuito a studenti di facoltà diverse, in modo da coprire nel modo più completo possibile il pubblico target. I partecipanti sono stati 156 studenti, di cui la metà provenienti dalla facoltà di Ingegneria. In figura 3.2 sono riportati i risultati ottenuti dall'analisi del questionario UEQ mediante il documento Excel predisposto.

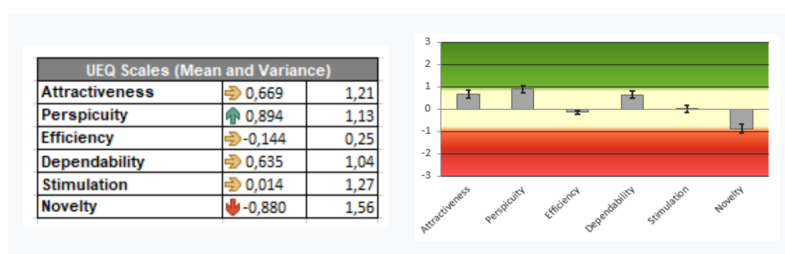
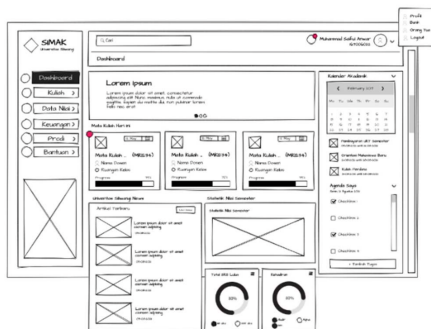


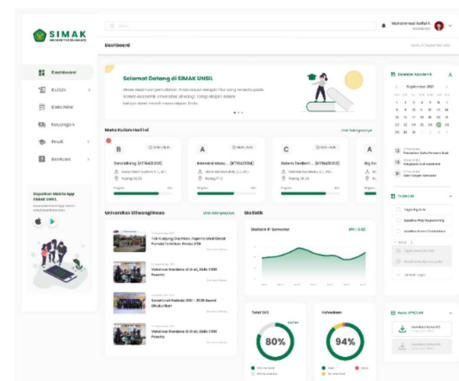
Figura 3.2: Risultati test UEQ somministrato agli studenti dell'Università Siliwangi

Facile notare come *Attrattività*, *Efficienza*, *Accuratezza* e *Stimolazione* abbiano avuto una valutazione neutrale, mentre *Originalità* ha riscontrato una valutazione negativa. L'esperienza utente risulta quindi da riprogettare.

Nello studio viene mostrato il Wireframe a bassa fedeltà (figura 3.3a) e la sua evoluzione in Wireframe ad alta fedeltà (figura 3.3a) relativo alla homepage di SIMAK WEB. Viene riportato qui di seguito dato che si tratta un esempio molto chiaro e ben esemplificativo del passaggio da un Wireframe all'altro.



(a) Wireframe a bassa fedeltà



(b) Wireframe ad alta fedeltà

Per concludere l'analisi di questo caso di studio, vengono riportati di seguito i risultati del test UEQ somministrato a seguito della progettazione dell'applicazione e della riprogettazione del portale web. In questo caso sono stati coinvolti un totale di 82 utenti, di cui 10 esperti. I risultati sono molto positivi e possono essere consultati di seguito in figura 3.3.

Scala UEQ	SIMAK OLD		SIMAK WEB NEW				SIMAK MOBILE NEW			
			Expert		End User		Expert		End User	
Attractiveness	0,669	1,21	2,611	0,30	2,214	0,65	2,854	0,04	2,083	0,71
Perspicuity	0,894	1,13	2,361	0,67	1,994	0,77	2,688	0,07	1,971	0,66
Efficiency	-0,144	0,25	-0,083	0,08	-0,141	0,16	-0,063	0,03	-0,029	0,15
Dependability	0,635	1,04	2,194	0,57	1,776	0,69	2,469	0,05	1,625	0,77
Stimulation	0,014	1,27	2,528	0,18	1,923	0,88	2,688	0,22	2,074	0,89
Novelty	-0,880	1,56	2,611	0,20	1,795	1,21	2,531	0,44	1,853	0,88

Figura 3.3: Risultati test UEQ a confronto

Per quanto riguarda SIMAK WEB, essendo presentenuna versione precedente è possibile fare un confronto tra vecchio e nuovo. Si può notare come ci sia stato un ottimo miglioramento generale in tutte le scale, specialmente in quello della *Originalità*, sia dal punto di vista degli utenti medi che da quello degli utenti esperti.

Passando invece all'analisi dei risultati per SIMAK APP, non è possibile fare un confronto con una versione precedente ma è comunque interessante analizzare i valori ottenuti dalle varie scale. Esclusa infatti l' *Efficienza*, tutte le altre scale hanno ottenuto

subito valutazioni molto positive, a riprova del fatto che il Design Thinking non permette solo di attuare riprogettazioni valide partendo da basi già esistenti ma è uno strumento molto potente anche per la progettazione da zero di sistemi software, permettendo fin dall'inizio una raccolta dei requisiti corretta che si riflette in un prodotto soddisfacente sotto tutti i punti di vista.

Per entrambi i sistemi è stato inoltre registrato un tasso di successo nel completamento di task molto alto, 90% per SIMAK WEB e 100% per SIMAK mobile, confermano ulteriormente le conclusioni ottenute sopra.

Il **secondo caso di studio** riportato è stato realizzato nel 2021 da Nasution et al. [NN21a] con l'obiettivo finale di realizzare il prototipo di un'applicazione web dedicata all'insegnamento, chiamata "IdeIn" in modo da dare un supporto all'educazione in Indonesia. Anche per lo sviluppo di questo sistema è stato utilizzato la metodologia del Design Thinking.

In questo caso non esiste un progetto precedente da cui cominciare e lo studio deve partire da zero. Le varie fasi del Design Thinking vengono seguite con attenzione ma una differenza dal processo descritto nella sezione 3.1 è il fatto che la user flow map non viene generata come output della fase di ideazione ma a seguito della prototipazione del prodotto. È stato creato un mockup cliccabile attraverso lo strumento Figma e per verificare il corretto collegamento tra le pagine è stata generata una user flow map che può essere trovata in figura 3.4.

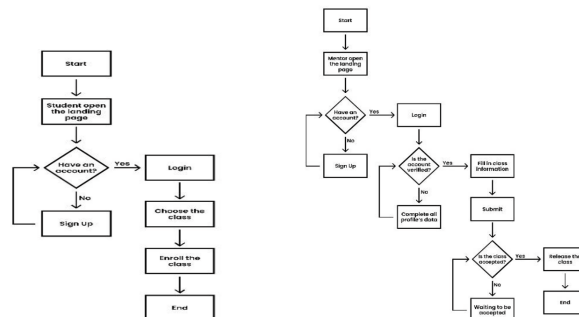


Figura 3.4: Flow map del'utente studente e flow map dell'utente mentore

Per quanto riguarda la fase di testing, rispetto al caso di studio presentato in precedenza, in questo caso viene utilizzato il sistema SUS. Gli utenti coinvolti sono 5 persone che potenzialmente potevano usufruire dell'applicazione web. Il risultato finale è molto positivo, infatti la user experience è stata valutata mediante il sistema SUS con un punteggio di 90/100. Anche in questo caso l'applicazione del Design Thinking risulta vincente per la creazione di un'applicazione web, nonostante l'assenza di un servizio vecchio da cui partire.

Capitolo 4

Applicazione del DT alla riprogettazione di Spotify

Durante i primi anni 2000 si era creato nell'industria musicale un serio problema legato alla pirateria, ovvero il download di contenuti musicali in modo illegale. Questo portò ad un notevole calo dei profitti per gli artisti e le case discografiche. Per risolvere questo problema i due imprenditori svedesi Daniel Ek and Martin Lorentzon pensarono di creare una piattaforma che permettesse agli iscritti di ascoltare musica e agli artisti di guadagnare mediante gli ascolti. A partire da questa idea venne creato quello che ora è conosciuto come il più grande servizio di streaming musicale, con più di 180 milioni di utenti premium in tutto il mondo, ovvero Spotify.

L'idea di una riprogettazione della versione desktop di Spotify nasce dalla volontà di voler mostrare le vere potenzialità del Design Thinking. La scelta di Spotify come oggetto della riprogettazione è stata ben ponderata: il prodotto selezionato è, infatti, volutamente noto e utilizzato da moltissimi utenti che hanno già un'idea chiara e per questo potrebbe sembrare più complicato effettuare una riprogettazione. Ma, in realtà, attraverso l'applicazione di questa metodologia è stato però possibile far emergere osservazioni non banali e problemi che quotidianamente non vengono notati, ma che, se risolti, possono portare a un miglioramento complessivo dell'esperienza utente. Ciò è stato possibile proprio grazie ad uno degli elementi cardine del Design Thinking, ovvero lo scambio di idee e il confronto.

Attraverso questa attività sperimentale si è quindi cercato di individuare problematiche inesplorate percepite dai fruitori dell'applicazione, proponendone poi una possibile soluzione.

Lo scopo finale del lavoro non è quindi solo quello di mostrare la riprogettazione sviluppata ma, soprattutto, quello di esaltare il potenziale del Design Thinking, come metodologia flessibile, adattabile e innovativa per la definizione di soluzioni moderne.

Di seguito, in figura 4.1, sono riportate le fasi principali del processo scelto.



Figura 4.1: Fasi principali del Design Thinking

4.1 Raccolta dei dati

La prima fase del processo di Design Thinking è quella di *Raccolta ed Empatia*. L'obiettivo di questa fase era quello di fare una ricerca sugli utenti, con lo scopo di rilevare problemi nell'esperienza utente. Sono state adottate due metodologie per la raccolta di dati, un gruppo di discussione e la distribuzione di un questionario per la valutazione dell'esperienza utente (UEQ).

Per prima cosa è stato organizzato un **gruppo di discussione** composto, per la precisione, da 4 persone. I partecipanti al focus group sono quattro giovani, di cui tre di genere femminile e uno di genere maschile, tutti di età compresa tra i 16 e i 24 anni. 3 di loro utilizzano l'applicazione tutti i giorni, due per la musica e uno per i podcast. L'ultimo candidato utilizza la piattaforma di tanto in tanto. Sono stati scelti questi candidati con l'idea di rappresentare al meglio il pubblico target di Spotify che, da una ricerca di start.io [Sta22], risulta essere composto per più del 50% da giovani adulti.

Essendo un piccolo gruppo, la discussione non è stata diretta da un moderatore ed è semplicemente stata posta una domanda iniziale da cui far partire il dibattito che si è poi sviluppato grazie agli input dati dai vari componenti. La domanda iniziale è stata:

"Quali sono i problemi che riscontri nell'utilizzo quotidiano di Spotify?"

Nonostante l'esperienza utente sia tutto sommato positiva sono stati rilevati delle problematiche nell'utilizzo della piattaforma. I problemi principali emersi durante l'attività di brainstorming sono i seguenti:

- Disorganizzazione delle pagine di principale interesse (*homepage, cerca, artista*)
- Monotonia e piattezza dell'interfaccia

Inoltre sono stati sottolineati i seguenti aspetti più specifici:

- Difficoltà nel trovare nuova musica e rimanere aggiornati sulle nuove uscite
- Complessità nel trovare le sezioni: *classifiche, eventi dal vivo e nuove uscite*
- Poca visibilità della sezione podcast
- Assenza della barra di ricerca durante la navigazione

Per capire appieno queste problematiche, è necessario esaminarle più attentamente. Di seguito, fornirò un'analisi di ciascuno di questi problemi, evidenziandone le sue cause.

La homepage di una piattaforma come Spotify dovrebbe avere un ruolo cruciale nel fornire ai propri utenti un'esperienza utente stimolante e coinvolgente. La sua progettazione dovrebbe aiutare gli utenti non solo a gestire le proprie playlist, ma anche a scoprire nuova musica e rimanere aggiornati sulle ultime tendenze musicali.

Uno degli elementi più di successo di Spotify sono sicuramente le playlist personalizzate create dalla piattaforma stessa. Il loro scopo è quello di permettere all'utente di rimanere aggiornato con l'uscita di nuove canzoni e scoprire nuove canzoni e artisti.

Attualmente, nella *homepage* vengono presentate centinaia di queste playlist, divise in numerose categorie spesso ridondanti e che cambiano ogni giorno, disorientando l'utente e impedendogli di apprezzare i consigli offerti dalla piattaforma.

Di seguito, nell'immagine 4.2 è riportata una parte dell'attuale homepage di Spotify, in cui possiamo notare la quantità eccessiva e la ridondanza delle raccomandazioni (evidenziato in rosso), a cui oltretutto viene dedicata la sezione finale della pagina, dando la precedenza inefficacemente a elementi che già conosco (evidenziato in blu). A questo si aggiunge anche un'interfaccia sostanzialmente monotona in cui a tutti gli elementi è dedicato lo stesso spazio e la stessa resa grafica - "*(...) album e podcast sono praticamente indistinguibili a livello visivo*" (P03).

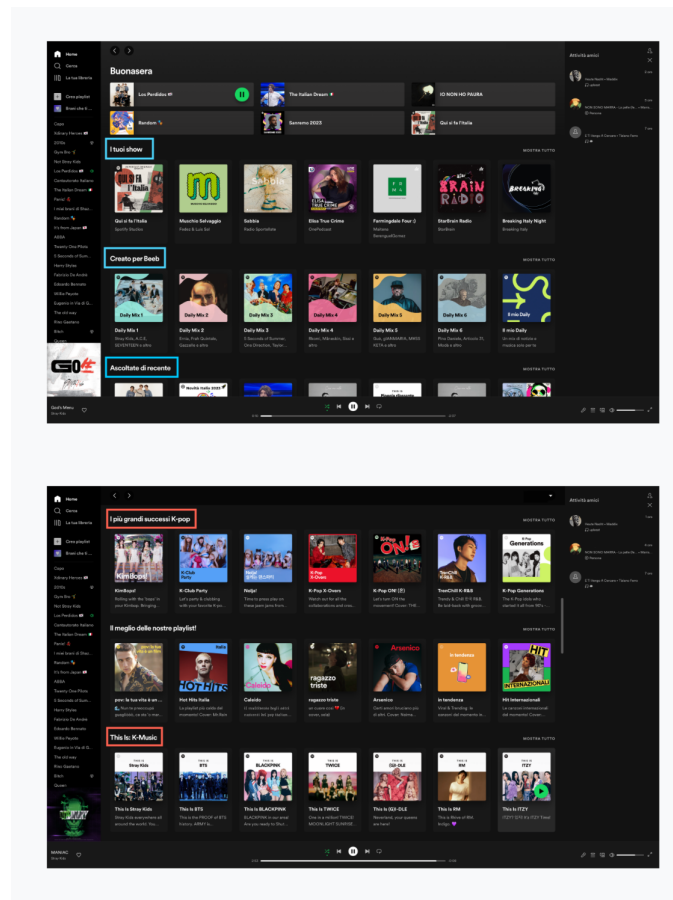
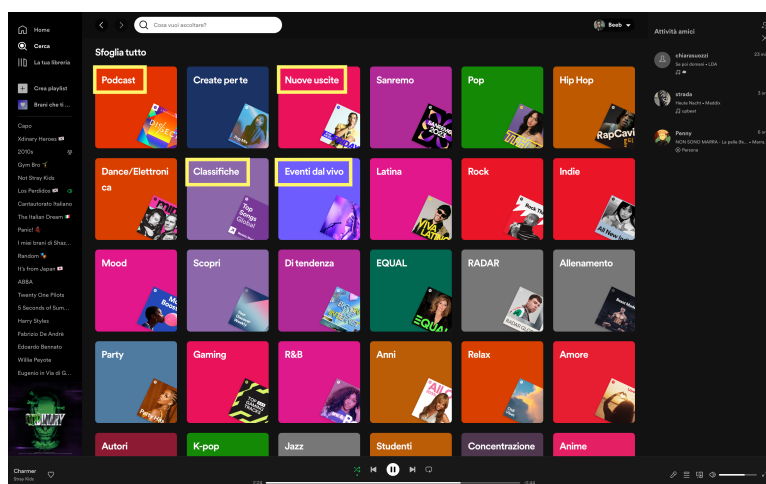


Figura 4.2: Parte dell'attuale *homepage* di Spotify

Stessa situazione si ripropone nella sezione *cerca* dove le playlist sono divise in un numero esagerato di generi, indistinguibili tra loro a primo impatto. I generi sono organizzati in un ordine apparentemente casuale, ad ognuno di essi è stata data la stessa grandezza e un colore tinta unita di sfondo, rendendo in questo modo l'interfaccia piatta e la ricerca molto confusionaria.

Inoltre, in mezzo ai generi, è possibile notare la presenza di sezioni che non sono pertinenti (evidenziate in giallo nell'immagine) e in particolare si trovano: *podcast*, che non secondo gli intervistati non riceve abbastanza visibilità in questo modo, *classifiche*, *eventi dal vivo* e *nuove uscite*, rendendone così poco intuitiva la consultazione - "*Ho scoperto molto di recente la presenza della sezione eventi dal vivo e se l'avessi conosciuta prima l'avrei sicuramente utilizzata di più.*" (P01) e "*Consulto le classifiche ogni venerdì e ogni volta faccio fatica a trovare la sezione dedicata.*" (P04). In figura 4.3 è riportata la pagina *cerca*.

Figura 4.3: Pagina *cerca* di Spotify

La pagina dell'artista è un'interminabile pagina in cui si trovano mescolate informazioni disparate, da una panoramica sull'artista alle playlist in cui compare, dai concerti ad artisti simili consigliati. L'utente in mezzo a tutte queste informazioni così diverse fa fatica a trovare ciò che sta cercando.

Per concludere, l'intervistato P02 ha sottolineato la scomodità di avere la barra di ricerca in una sola pagina e che sarebbe molto più comodo averla sempre a disposizione, in qualsiasi pagina uno si trovi - *"potrebbe sembrare un'osservazione poco importante perché si tratta solo di un click ma spesso trovo scomodo il fatto di dover cambiare pagina per fare una ricerca"* (P02). Questo ci fa capire come l'esperienza utente sia formata anche da piccoli dettagli che potrebbero rimanere impressi all'utente e determinare la differenza tra un'esperienza piacevole e una sgradevole.

Per ottenere risultati complessivamente più accurati, la ricerca qualitativa condotta mediante gruppo di discussione è stata abbinata ad una ricerca di tipo quantitativo, realizzata mediante un questionario per la valutazione dell'esperienza utente. Il questionario scelto è il modello UEQ analizzato nella sezione 2.4.2. Le motivazioni che hanno portato a questa scelta sono molteplici, ma vale la pena evidenziare: alcuni punti:

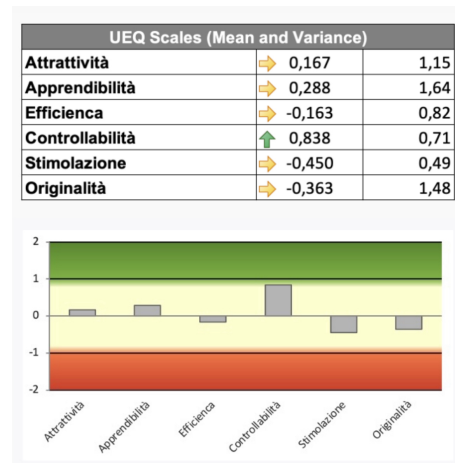
- Validità: dalla sua creazione, il modello UEQ è stato ampiamente utilizzato, testato e validato e ha sempre dimostrato di essere valido e affidabile nella valutazione.
- Copertura completa: il questionario è in grado di coprire tutti gli aspetti fondamentali dell'esperienza utente, fornendone un'analisi a completa.
- Disponibilità in varie lingue: solitamente i questionari per la valutazione dell'esperienza utente si basano su sfumature lessicali e se la lingua utilizzata non è compresa appieno dall'intervistato i risultati potrebbero risultare falsati. Per questo era importante trovare un questionario che fosse in italiano.

- **Rapidità:** uno dei punti di forza del questionario UEQ è la rapidità con cui può essere compreso e compilato.

Il questionario è stato compilato da 20 persone di età compresa tra i 16 e i 24 anni, di cui 15 di genere femminile e 5 di genere maschile. Per ogni domanda è stata calcolata la media, che può essere consultata nella seconda colonna della figura 4.4a e ogni domanda è stata assegnata ad un colore in base alla scala di appartenenza. Nella figura 4.4b è possibile esaminare la media calcolata per le 6 scale e la loro rappresentazione grafica.

Item	Mean	Variance	Std. Dev.	No.	Left	Right	Scale
1	0,3	2,1	1,4	20	increscioso	gradito	Attrattività
2	0,6	1,6	1,3	20	incomprensibile	comprensibile	Apprendibilità
3	-0,5	2,1	1,4	20	creativo	privo di fantasia	Originalità
4	0,7	2,2	1,5	20	facile da capire	difficile da capire	Apprendibilità
5	-0,1	2,2	1,5	20	di grande valore	di poco valore	Stimolazione
6	-0,8	1,3	1,1	20	noioso	appassionante	Stimolazione
7	-0,6	1,6	1,3	20	non interessante	interessante	Stimolazione
8	1,0	1,3	1,1	20	imprevedibile	prevedibile	Controllabilità
9	0,4	1,2	1,1	20	veloce	lento	Efficienza
10	-0,6	2,6	1,6	20	originale	convenzionale	Originalità
11	0,4	1,7	1,3	20	ostruttiva	di supporto	Controllabilità
12	0,4	1,9	1,4	20	bene	male	Attrattività
13	0,2	1,9	1,4	20	complicato	facile	Apprendibilità
14	0,1	1,8	1,4	20	repellente	attraente	Attrattività
15	-0,3	3,1	1,8	20	usuale	moderno	Originalità
16	0,3	1,3	1,1	20	sgradevole	piacevole	Attrattività
17	1,4	1,3	1,1	20	sicuro	insicuro	Controllabilità
18	-0,4	1,9	1,4	20	attivante	sonnifero	Stimolazione
19	0,6	2,5	1,6	20	aspettativo	non aspettativo	Controllabilità
20	0,5	1,2	1,1	20	inefficiente	efficiente	Efficienza
21	-0,4	3,0	1,7	20	chiaro	confuso	Apprendibilità
22	-0,2	1,8	1,3	20	non pragmatico	pragmatico	Efficienza
23	-1,3	2,7	1,7	20	ordinato	sovraccarico	Efficienza
24	0,1	2,2	1,5	20	attraente	non attraente	Attrattività
25	0,0	0,8	0,9	20	simpatico	antipatico	Attrattività
26	-0,2	1,6	1,3	20	conservativo	innovativo	Originalità

(a) Risultati calcolati per ogni domanda



(b) Risultati calcolati per ogni scala

In entrambi i casi se il valore medio è compreso tra $+0.8$ e -0.8 la valutazione di quel fattore è neutrale, sopra è positiva e sotto è negativa.

Gli elementi per cui è stata riscontrata una valutazione positiva sono imprevedibile/prevedibile (item 8) e insicuro/insicuro (item 17). Entrambi questi elementi appartengono alla scala che ha ricevuto la valutazione media più alta, ovvero la *Controllabilità*. L'utente sente di avere un buon controllo dell'applicazione, che funziona senza errori, si comporta come da aspettative e mantiene al sicuro le informazioni dell'utente.

Gli elementi che hanno avuto la valutazione media più bassa sono non interessante/interessante (item 7), noioso/appassionante (item 6) e ordinato/sovraccarico (item 23) che è anche l'unico elemento ad aver avuto una valutazione <-0.8 . Questi tre elementi fanno parte rispettivamente delle tre scale che hanno ottenuto le medie più basse, ovvero *Stimolazione*, *Originalità* e *Efficienza*.

La scala della *Stimolazione* rappresenta quanto l'utente sia stato stimolato e si sia divertito durante l'utilizzo dell'applicazione, quella dell'*Originalità* valuta la novità dell'interfaccia e la presenza o meno di elementi innovativi e caratteristici che riescano ad attirare l'attenzione. Infine, la scala dell'*Efficienza* stima il grado di produttività del-

l'applicazione e quanto risulti intuitivo lo svolgimento delle varie attività.

Valori così bassi per queste tre categorie sono perfettamente in linea con i problemi rilevati dal gruppo di discussione, ovvero la disorganizzazione e il sovraccaricamento delle pagine principali e un'interfaccia noiosa e monotona.

Mettendo insieme i dati qualitativi ottenuti dal gruppo di discussione e i dati quantitativi ottenuti dal questionario UEQ è stato valutato che le pagine che richiedessero il maggiore intervento fossero quelle *Homepage*, *Scopri* e *Artista*. Sembrerebbe infatti che siano queste le pagine che rimangano maggiormente impresse e che influiscono di più sull'esperienza utente.

L'obiettivo generale della riprogettazione sarà quello di ottenere un'interfaccia chiara, organizzata e più coinvolgente, oltre che risolvere i problemi più specifici segnalati dal gruppo di discussione.

4.2 Definizione del problema

Per la fase di definizione del problema sono state create tre user persona, con lo scopo di riassumere i problemi rilevati. Le user persona definite sono: *appassionato di musica*, *ascoltatore di podcast seriale* e *utente medio*

<i>Come appassionato di musica</i>	
<i>Voglio</i>	voglio scoprire nuove canzoni e rimanere aggiornato sulle ultime uscite
<i>In modo da</i>	poter espandere la mia cultura musicale ma allo stesso tempo seguire i miei artisti preferiti e non perdermi nessuna uscita
<i>Scenario</i>	In quanto grande appassionato di musica, utilizzo Spotify ogni giorno. Tuttavia, trovo le pagine principali (Artista, Home e Scopri) confuse e disordinate, di conseguenza fatico a trovare ed esplorare nuove playlist e mi perdo le nuove uscite. Inoltre amo consultare le playlist settimanali e partecipare ai concerti ma le sezioni dedicate sono poco in evidenza e difficili da individuare.

<i>Come ascoltatore di podcast seriale</i>	
<i>Voglio</i>	voglio trovare facilmente e accedere ai miei podcast preferiti, così da rimanere informato e non perdermi neanche un episodio
<i>In modo da</i>	poter espandere la mia cultura musicale ma allo stesso tempo seguire i miei artisti preferiti e non perdermi nessuna uscita
<i>Scenario</i>	In quanto grande amante dei podcast, utilizzo Spotify ogni giorno. Tuttavia, essendo la grafica utilizzata per i podcast nella homepage la stessa utilizzata anche per album e playlist, fatico a trovare quello che sto cercando. Inoltre trovo difficile individuare la sezione dedicata ai podcast per scoprire nuovi programmi.

<i>Come utente medio</i>	
<i>Voglio</i>	voglio trovare facilmente musica che si adatta al contesto o a come mi sento.
<i>In modo da</i>	poter poter rapidamente trovare intrattenimento senza dover venire sopraffatto da centinaia di consigli diversi.
<i>Scenario</i>	In quanto utente medio utilizzo Spotify occasionalmente e quando lo uso trovo le pagine molto caotiche e difficili da navigare. Di conseguenza faccio fatica a trovare musica che faccia al mio caso e sono portato a utilizzare piattaforme diverse. Vorrei un'interfaccia più chiara e immediata.

4.3 Ideazione

Conclusa le fasi di *Raccolta dei dati e definizione del problema* si è passati alle fasi di *ideazione e prototipazione*, ossia quelle più cruciali dell'intero processo.

Per **trovare idee e sviluppare ipotetiche soluzioni** sono stati consultati documenti, ricerche e manuali dedicati al web design e alla progettazione dell'esperienza utente. In particolare è stato molto utile il manuale *Research-Based Web Design & Usability Guidelines* scritto da Michael O. Leavitt [LS+06]. Nonostante sia stato pubblicato nel 2006, è ancora considerato un validissimo strumento per la progettazione di interfacce utente, in quanto raccoglie tutte le principali linee guida da seguire durante la creazione di un'interfaccia utente per l'ottenimento di una esperienza utente piacevole.

Per quanto riguarda l'organizzazione delle pagine, è consigliabile ottimizzare la densità di elementi sul display ed evitare di creare pagine troppo piene, che finiscono per risultare caotiche e disordinate all'occhio dell'utente. La pagina ha, infatti, il compito di guidare e aiutare l'utente a trovare gli elementi più rilevanti. Per gli elementi più importanti dovrebbero essere riservata la parte superiore-centrale dello schermo, mentre gli elementi meno importanti dovrebbero essere posizionati nella parte inferiore.

Parliamo in questo caso di gerarchia visiva, ovvero la pratica di organizzare i contenuti in modo che gli utenti possano trovare rapidamente ciò che stanno cercando. Questa organizzazione visiva può essere ottenuta attraverso l'uso di elementi quali dimensione, colore, posizione, spazi bianchi e tipografia 4.4 come afferma Ratri et al.[RPC+22]. Nella stessa ricerca viene menzionato, in abbinamento alla gerarchia visuale, il Call To Action (CTA), ovvero un qualsiasi elemento che può portare l'utente a compiere una certa azione, come nel nostro caso cliccare su una nuova playlist, album o podcast.

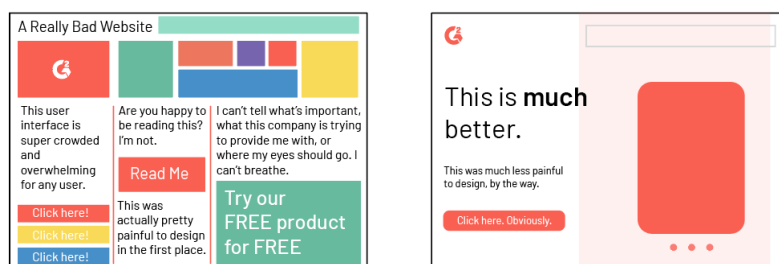


Figura 4.4: Esempio di riorganizzazione grazie ai principi di *gerarchia visuale*

Altro consiglio che viene dato nella guida di Leavitt relativamente alla struttura delle pagine è quello di evitare di creare pagine troppo lunghe che contengono molte informazioni diverse e prediligere piuttosto la paginazione, con le informazioni raccolte logicamente in pagine corte e di facile accesso.

Bisogna infatti sempre considerare che la prima cosa che fa un utente quando arriva su una pagina è quella di scannerizzare con gli occhi il contenuto fino a che non trova ciò che necessita, per poi concentrarsi e consultarne il contenuto, per ciò bisogna facilitare questa attività con soluzioni come quelle appena citate.

4.4 Prototipazione

Dopo aver raccolto spunti e idee, anche osservando l'offerta dei competitor, è iniziata la fase di prototipazione, che, come già accennato in precedenza, è divisa in due sotto-fasi, ovvero quella di prototipazione a bassa fedeltà e di prototipazione ad alta fedeltà.

La **prototipazione a bassa fedeltà** è stata fatta prima realizzando dei rapidi disegni a mano. L'obiettivo era quello di concretizzare tutte le idee emerse nella fasi precedenti. Dopo varie prove si è giunti ad una soluzione appagante che sembrava riuscisse ad affrontare tutte le problematiche rilevate ed è visionabile in figura 4.5 .

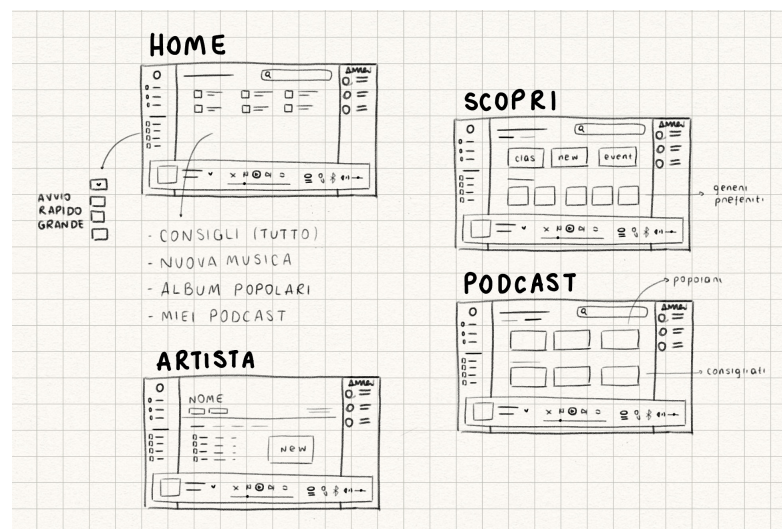
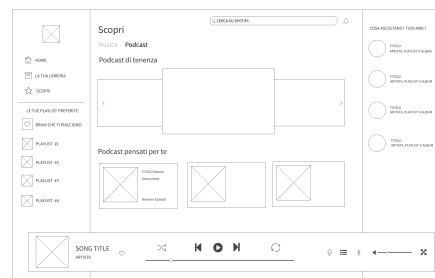
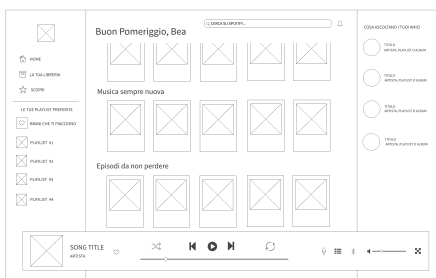
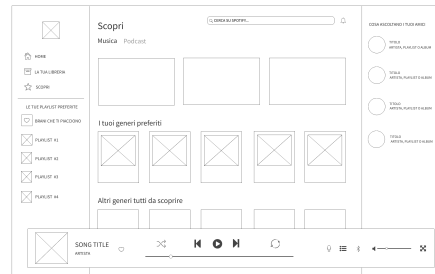
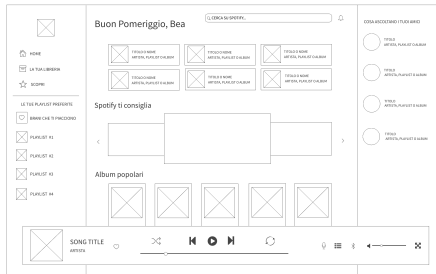


Figura 4.5: Prototipo realizzato a mano

In questa fase è stata valutata una importante modifica nella struttura delle pagine: rendendo la barra di ricerca disponibile in ogni pagina, la pagina chiamata *cerca* poteva essere rinominata con un nome più evocativo, che descrivesse meglio cosa si può trovare al suo interno, ovvero *scopri*. La pagina non conteneva solo la barra di ricerca ma anche molti strumenti per la scoperta di nuovi contenuti e con l'etichetta precedente questo aspetto veniva totalmente nascosto.

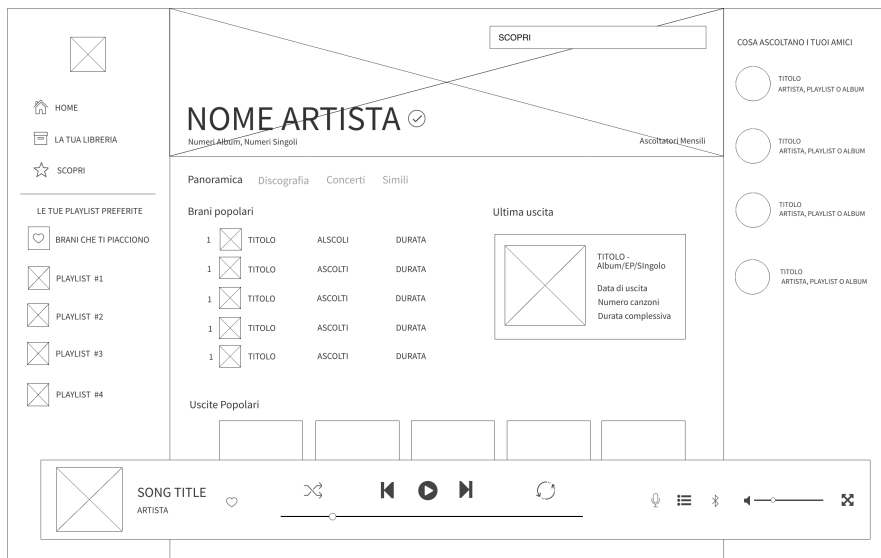
A seguire le bozze realizzate a mano sono state rifinite attraverso un software per la creazione di wireframe online chiamato MockFlow. Questo passaggio è servito per avere

una prima resa concreta dell'interfaccia con le proporzioni corrette e una visione generale dell'organizzazione degli elementi. Di seguito sono riportati i wireframe realizzati per le pagine *Homepage* 4.6a, *Scopri Musica*, *Scopri Podcast* 4.6b e *Artista* 4.6c.



(a) Homepage

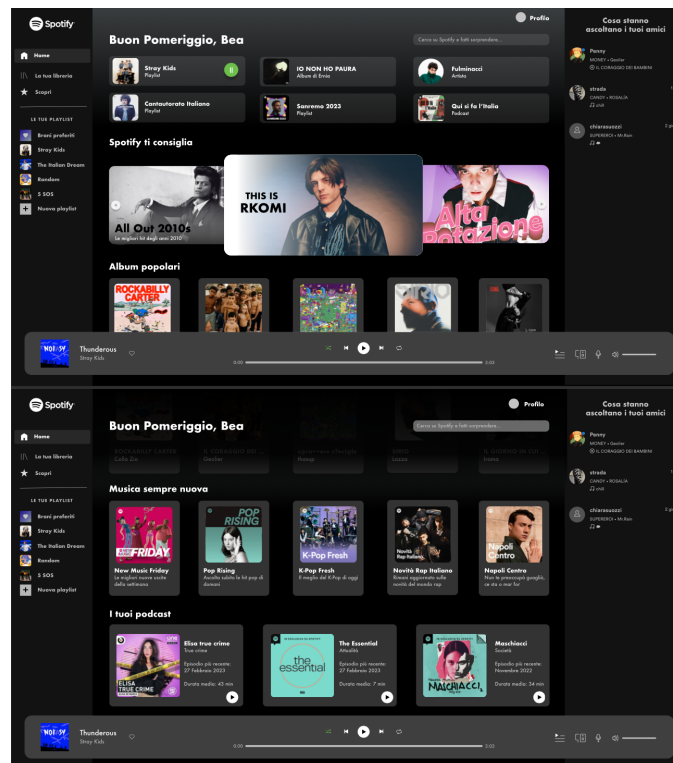
(b) Scopri



(c) Artista

La **prototipazione ad alta fedeltà** è stata realizzata con uno dei software maggiormente utilizzati nell'ambito della progettazione di interfacce utente, ovvero Figma. L'obiettivo di questo tipo di prototipazione è quello di creare un prototipo dell'interfaccia utente che rappresenta il design finale del prodotto. Risulta essere più dettagliato e completo di un wireframe a bassa fedeltà, dato che comprende elementi grafici, colori, font e immagini. Di seguito i risultati ottenuti con alcuni commenti sulle scelte prese.

Figura 4.6: *Homepage completa*



L'obiettivo della homepage è quello di creare una piacevole prima impressione. Un utente poco interessato potrebbe abbandonare subito la piattaforma e rivolgersi a un competitor. Non ci sarà una seconda possibilità di fare una bella impressione sull'utente.

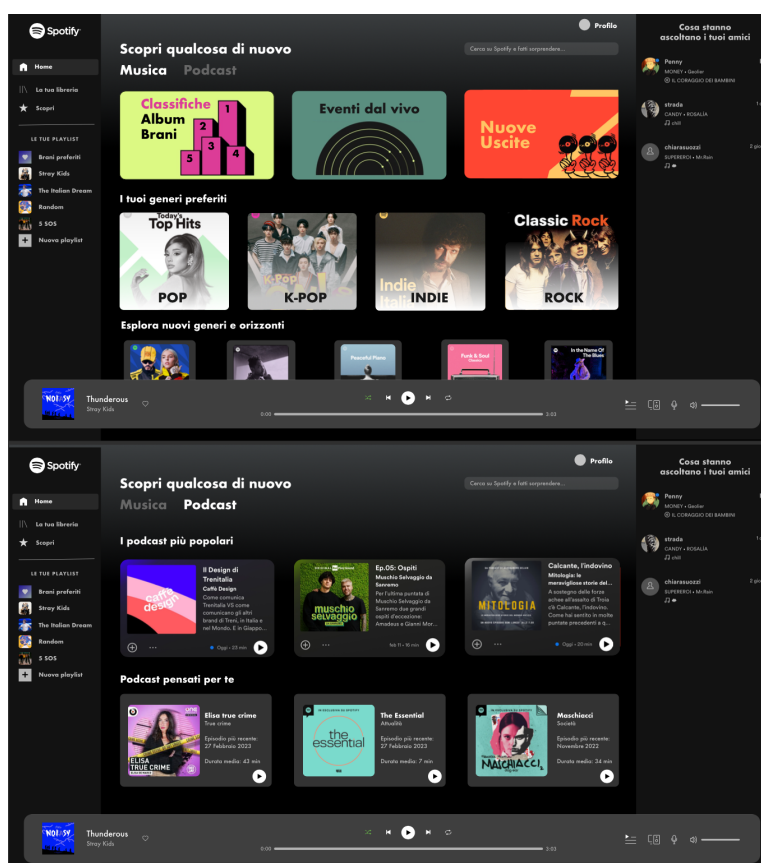
Ciò che consiglia Leavitt nella sua guida è di presentare nella homepage le principali opzioni offerte dalla piattaforma. Bisogna, per questo, attuare una selezione delle più importanti, per non sovraccaricare la pagina con informazioni superflue. Nella mia proposta, in alto, si trovano gli ultimi contenuti ascoltati di facile accesso. A seguire vi è un carosello con alcune proposte della piattaforma che potrebbero essere nuove uscite di artisti seguiti, playlist, album o podcast. La dimensione scelta per le card e la dinamicità del carosello servono per attirare l'attenzione dell'utente, che potrebbe essere maggiormente invogliato a cliccare.

Sotto si trovano due righe di consigli musicali: album popolari e proposte di playlist, che dal gruppo di discussione sono risultate le categorie più interessanti, entrambi con l'obiettivo di far conoscere all'utente nuova musica o nuovi artisti. In fondo si trova la sezione dedicata ai podcast che l'utente segue. Queste tre sezioni finali possono essere ricombinate a seconda delle preferenze dell'utente.

Altre modifiche che sono state apportate riguardano la navbar sinistra, in cui non si trovano più *tutte* le playlist dell'utente, ma solo le più importanti in numero limitato. Ipoteticamente potrebbero essere scelte dall'utente stesso, come se fossero dei preferiti. Anche questo accorgimento aiuta a rendere la pagina più chiara e meno caotica. Inoltre la copertina dell'album è stata rimossa dalla navbar sinistra e posizionata all'interno del player, dove ci si aspetta di trovarla logicamente. Infine, ai podcast è stata data una grafica diversa in cui è possibile inserire più dettagli e che permette di distinguerli da album e playlist. L'utente deve riuscire a distinguere a colpo d'occhio musica e podcast.

La prossima analisi riguarda la pagina *scopri* che, come già detto, è stata rinominata dalla precedente pagina *cerca*.

Figura 4.7: Pagina *Scopri* completa



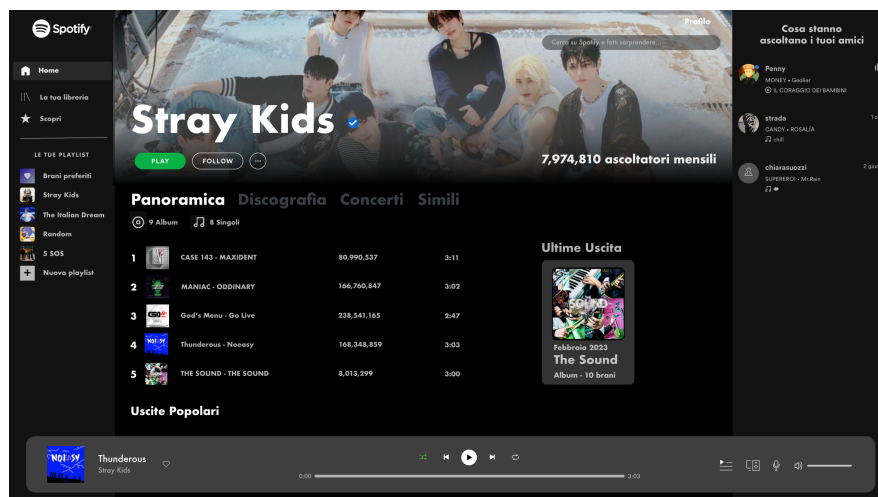
La prima modifica realizzata è l'inserimento di una navbar con due elementi: Musica e Podcast con l'idea di dare equo spazio e visibilità alle due pagine.

Nella pagina dedicata alla musica, in alto si trovano tre card dedicate rispettivamente alle *Classifiche*, *Eventi dal vivo* e *Nuove uscite* che nell'organizzazione piatta precedente non erano messe in risalto. A seguire si trovano quattro tra i generi preferiti dall'utente, a cui è stata data anche una grafica più accattivante di un semplice sfondo tinta-unita e in fondo si trovano altri generi. Si può notare come in questa riorganizzazione siano stati applicati i vari principi della gerarchia visiva, ovvero il posizionamento e l'utilizzo di forme, dimensioni e colori diversi per mettere in risalto gli elementi più importanti.

La pagina dedicata ai podcast è stata alleggerita con un numero più limitato di consigli più mirati e in particolare in alto si trovano *podcast popolari* e *podcast consigliati*.

Per concludere, l'analisi della pagina dell'artista.

Figura 4.8: Panoramica Artista



La pagina dell'artista aveva il problema di essere molto lunga e contenere al suo interno moltissime informazioni disparate, rendendo difficile trovare ciò che si sta cercando. Per questo motivo si è preferito dividere l'intera pagina in quattro sotto-pagine più corte in cui distribuire le informazioni. Le sezioni individuate sono: *Panoramica* in cui troviamo le informazioni principali dell'artista (numero di album e singoli, canzoni e uscite più popolari e ultima uscita con alcune informazioni al riguardo.), *Discografia*, *Concerti* e *Contenuti simili*.

Spotify ha la come punto di forza un branding molto ben riconoscibile e accattivante ma nella sua applicazione desktop non viene sfruttato il suo vero potenziale.

La fase di progettazione appena descritta, che si compone di empatia, definizione, ideazione e prototipazione, è stata un momento cruciale del processo di ricerca. Durante queste fasi, è stato possibile esplorare il punto di vista degli utenti, cogliendo esigenze e dubbi, e su queste informazioni generare idee e di soluzioni, per poi trasformarle in prototipi tangibili. Tuttavia, la verifica della corrispondenza tra i risultati attesi e quelli ottenuti avverrà nella fase successiva, ovvero quella del *test* e dell'analisi dei dati.

Capitolo 5

Valutazione dei risultati ottenuti

Una volta completata la fase di progettazione, si può passare all'ultima fase, ovvero quella del *Testing* per poi trarre le conclusioni. È importante ricordare che il processo del Design Thinking è molto flessibile, quindi da un qualsiasi punto è possibile tornare indietro ad una fase precedente per apportare delle modifiche. Inoltre si tratta di un modello iterativo e il ciclo può essere ripetuto tutte le volte che si ritiene necessario fino a che non si ottiene un risultato soddisfacente che può passare alla fase di implementazione.

A seconda delle conclusioni ottenute dalla fase di *Testing* è possibile interrompere l'iterazione o ripeterla.

In questo capitolo verranno analizzati i dati raccolti a seguito della riprogettazione in modo da trarre delle conclusioni sull'impatto del Design Thinking come strumento per la riprogettazione dell'esperienza utente e dell'interfaccia utente.

Esistono diverse modalità per la raccolta dei dati nella fase di testing, ma è scelto di utilizzare nuovamente il questionario UEQ, proponendolo allo stesso gruppo di intervistati coinvolti nella fase di define. Questa scelta ci ha permesso non solo di analizzare in modo indipendente il prodotto ottenuto dalla riprogettazione, ma anche di condurre un'analisi longitudinale pre-post.

Partendo dalla **somministrazione del questionario**, esso è stato riproposto ad una settimana di distanza dal primo questionario, allo stesso gruppo di individui, quindi un totale di 20 persone di età compresa tra i 16 e i 24 anni, di cui 15 di genere femminile e 5 di genere maschile.

In questo caso però prima del questionario è stato mostrato agli intervistati il risultato ottenuto dalla riprogettazione con alcune osservazioni e commenti sulle modifiche fatte e sulle motivazioni dietro a quelle scelte.

Tra gli strumenti messi a disposizione per l'analisi dei risultati dei questionari UEQ, vi è anche un analizzatore di coerenza dei questionari. C'è infatti la possibilità che gli intervistati non rispondano seriamente alle domande e, specialmente per la rapidità del questionario, potrebbero essere date risposte poco precise. Per questo motivo viene

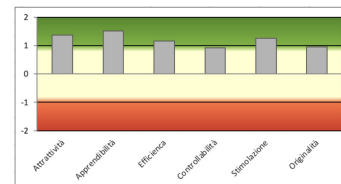
offerto questo strumento che ha l'obiettivo di rilevare questionari inconsistenti; per la precisione, considera per ogni questionario e per ogni scala, l'elemento che ha ricevuto la valutazione peggiore e la valutazione migliore; in una scala coerente questi due valori dovrebbero essere vicini. Se la differenza tra i due è >3 la scala è considerata incoerente. se un questionario presenta più di 3 scale incoerenti dovrebbe essere eliminato dall'analisi. Sia i questionari precedenti alla riprogettazione che quelli successivi sono risultati tutti coerenti.

A questo punto è stato chiesto di compilare il questionario e i risultati sono stati molto positivi. Nella figura 5.1a sono stati riportati i punteggi medi ottenuti da ogni domanda, mentre nella figura 5.1b è possibile esaminare la media calcolata per le 6 scale e la loro rappresentazione grafica.

Item	Mean	Variance	Std. Dev.	No.	Left	Right	Scale
1	1,6	2,9	1,7	20	incresoso	gradito	Attrattività
2	1,2	3,7	1,9	20	incomprensibile	comprensibile	Apprendibilità
3	1,9	0,2	0,4	20	creativo	povero di fantasia	Originalità
4	1,6	3,5	1,9	20	facile da capire	difficile da capire	Apprendibilità
5	1,2	2,9	1,7	20	di grande valore	di poco valore	Stimolazione
6	1,3	2,1	1,5	20	noioso	appassionante	Stimolazione
7	1,4	2,6	1,6	20	non interessante	interessante	Stimolazione
8	0,0	3,8	1,3	20	imprevedibile	prevedibile	Controllabilità
9	1,0	3,1	1,8	20	veloce	lento	Efficienza
10	0,8	3,9	2,0	20	originale	convenzionale	Originalità
11	1,7	3,5	1,9	20	colluttiva	di supporto	Controllabilità
12	1,6	3,4	1,8	20	buena	male	Attrattività
13	1,6	3,2	1,8	20	complicato	facile	Apprendibilità
14	1,3	3,3	1,8	20	repellente	attraente	Attrattività
15	0,8	4,1	2,0	20	moderno	originale	Originalità
16	1,5	3,6	1,9	20	sggradevole	piacevole	Attrattività
17	1,3	3,1	1,8	20	sicuro	insicuro	Controllabilità
18	1,2	3,3	1,8	20	attivante	soporifero	Stimolazione
19	0,7	3,2	1,8	20	aspettativo	non aspettativo	Controllabilità
20	1,5	4,1	2,0	20	inefficiente	efficiente	Efficienza
21	1,8	3,6	1,9	20	chiaro	confuso	Apprendibilità
22	0,7	2,5	1,6	20	non pragmatico	pragmatico	Efficienza
23	1,6	4,8	2,2	20	ordinario	sovranicario	Efficienza
24	1,2	2,7	1,6	20	attraivo	non attraivo	Attrattività
25	1,1	3,6	1,9	20	simpatico	antipatico	Attrattività
26	0,4	2,5	1,6	20	conservativo	innovativo	Originalità

(a) Risultati calcolati per ogni domanda

UEQ Scales (Mean and Variance)		
Attrattività	↑ 1,375	3,03
Apprendibilità	↑ 1,513	3,25
Efficienza	↑ 1,163	3,08
Controllabilità	↑ 0,913	2,15
Stimolazione	↑ 1,263	2,35
Originalità	↑ 0,950	1,52



(b) Risultati calcolati per ogni scala

Si ricorda che, in entrambi i casi, se il valore medio è compreso tra $+0.8$ e -0.8 la valutazione di quel fattore è neutrale, sopra è positiva e sotto è negativa.

Per quanto riguarda le singole domande è possibile notare come quasi tutte le risposte abbiano ottenuto un valore medio >1 , quindi positivo. Le uniche domande che fanno eccezione sono gli item 8, 10, 15, 19, 22 e 26 che hanno ricevuto i punteggi più bassi, rientrando nella zona neutrale, ma risultano comunque superiori a 0.

Passando invece all'analisi delle scale che valutano i vari aspetti dell'esperienza utente, risultano uniformemente positivi con valori medi coerenti tra loro, che vanno dal $+0,9$ al $+1,5$. Il migliore è l'*Apprendibilità* mentre il peggiore è la *Controllabilità*.

Più interessante dell'analisi indipendente del prodotto riprogettato è sicuramente il confronto longitudinale pre-post. Il confronto longitudinale è una metodologia di analisi dei dati molto potente che prevede il confronto dei dati ottenuti in momenti temporali diversi, con lo scopo di stimare eventuali miglioramenti o peggioramenti nel tempo. Questo approccio è molto utile per la valutazione di un intervento, come in questo caso

specifico la riprogettazione dell'esperienza utente di un software. L'analisi longitudinale può essere di due tipi: pre-post o multi-gruppo.

- Analisi longitudinale pre-post: si confrontano i risultati ottenuti dallo stesso gruppo prima e dopo l'intervento ed è utile quando l'obiettivo della ricerca è valutare gli effetti di un intervento.
- Analisi longitudinale multi-gruppo: prevede la raccolta dei dati da due o più gruppi diversi e viene utilizzato nel caso si vogliano sottolineare le differenze tra i gruppi in relazione a un qualche fattore.

Per questa situazione, per analizzare al meglio i risultati della riprogettazione, si è optato per un confronto pre-post soprattutto perché, in un confronto multi-gruppo, si potrebbero avere gruppi di partecipanti con caratteristiche molto diverse, rendendo così difficile l'interpretazione dei dati.

Tra i tool offerti per l'analisi dei dati dei questionari UEQ, vi è anche uno strumento per l'analisi comparativa tra prodotti. I prodotti in questione potrebbero essere diversi oppure lo stesso prima e dopo un intervento come in questo caso. Di seguito in figura 5.1 è possibile visualizzare un confronto diretto tra le 6 scale. In blu i risultati ottenuti prima della riprogettazione e in rosso i risultati ottenuti dopo la riprogettazione:

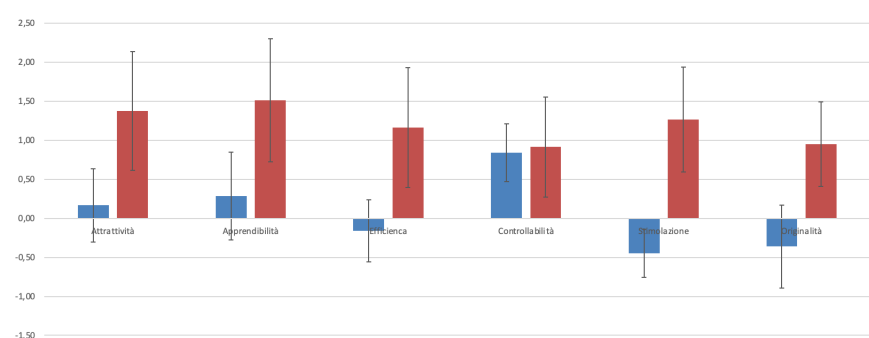


Figura 5.1: Confronto tra i risultati pre e post progettazione

I risultati sono perfettamente in linea con gli obiettivi iniziali, ovvero migliorare le scale *Stimolazione*, *Originalità* e *Efficienza*.

La scala dell'*Efficienza* si occupa di valutare quanto sia efficiente portare a termine le proprie attività. Le attività che, durante la fase di *Define*, sono risultate quelle più problematiche e faticose da completare sono la scoperta di nuovi contenuti, ricerca, aggiornamento su nuove uscite, e consultazione di podcast e playlist consigliate. Focalizzandosi su questi aspetti durante la riprogettazione è stato possibile ottenere un notevole miglioramento nella scala in questione. Alcune soluzioni adottate sono la riorganizzazione delle pagine con accurata selezione dei loro contenuti, applicazione dei principi della

gerarchia visuale e paginazione.

Per migliorare la scala dell'*Originalità* si è cercato di rendere l'interfaccia più accattivante, in modo che potesse attirare l'attenzione dell'utente senza annoiarlo. Per fare questo si è ricorsi a espedienti di web design come l'utilizzo di forme e dimensioni diverse, caroselli dinamici e colori accesi che facessero contrasto con lo sfondo scuro.

Tutti questi fattori hanno permesso anche un miglioramento della scala della *Stimolazione*, infatti il risultato è un'interfaccia e, in generale, un'esperienza utente, molto più stimolante e divertente. L'utente è quindi più invogliato ad esplorare la piattaforma e passarci più tempo, obiettivo cardine di qualsiasi piattaforma.

La scala che ha avuto il miglioramento meno evidente è quella della *Controllabilità* ma anche questo dato è coerente con il lavoro svolto, infatti era stata considerata già positivamente durante la fase di raccolta dei dati e di conseguenza non era stata valutata l'esigenza di apportarvi delle migliorie.

Bisogna considerare il fatto che questi risultati non rappresentano una valutazione assoluta della qualità del prodotto, ma indicano una valutazione assoluta del miglioramento percepito dagli utenti.

Si può quindi concludere che individuando i problemi e applicando delle semplici modifiche è possibile incrementare notevolmente l'esperienza utente e avere un ottimo feedback dagli utenti. Il processo di Design Thinking, in particolare, si è dimostrato molto potente per una corretta e precisa definizione dei problemi e per la ricerca di soluzioni soddisfacenti e innovative. Oltretutto, data la sua natura iterativa ed essendo basato sulla prototipazione veloce, il processo del Design Thinking, ha un impatto relativamente basso in termini di tempo e risorse economiche e quelle che vengono impiegate permetteranno di risparmiarne altrettante durante il processo molto più rigido e delicato di implementazione.

Capitolo 6

Conclusione e sviluppi finali

In questo lavoro, è stato esplorato il potenziale del Design Thinking nel processo di sviluppo software. L'obiettivo della ricerca è stato quello di capire se l'implementazione del Design Thinking come metodologia per problem solving potesse migliorare la qualità dei prodotti software e quali sono gli ostacoli che impediscono la sua adozione da parte delle aziende.

Sono stati infatti portati esempi di successo di applicazione di Design Thinking a processi di progettazione e riprogettazione nel Capitolo 2 e attraverso un'attenta applicazione della metodologia, nel Capitolo 3, è stato dimostrato che è possibile ottenere ottimi risultati e miglioramenti in tutti gli aspetti dell'esperienza utente grazie a questa metodologia. Oltretutto, se l'applicazione della metodologia è stata portata avanti con successo da una sola persona con pochi strumenti e poche risorse a disposizione, allora è ragionevole supporre che anche le organizzazioni possano integrare questo approccio nei loro processi di sviluppo software.

I dati raccolti e analizzati nei capitoli precedenti hanno dimostrato che il Design Thinking può essere effettivamente una metodologia efficace per sviluppare prodotti software più innovativi e incentrati sull'utente. Infatti si tratta di un approccio costituito da poche fasi poco dispendiose, molto flessibile, ben adattabile al contesto e con a disposizione moltissimi strumenti per ognuna delle sue fasi, come è stato mostrato. Tuttavia, la sua implementazione pratica può essere difficoltosa a causa di una serie di ostacoli che sono stati analizzati nel corso del lavoro.

In particolare, per una buona integrazione del Design Thinking è necessario un cambiamento culturale e organizzativo significativo da parte dei leader dell'organizzazione.

In secondo luogo è importante coinvolgere gli stakeholder interni ed esterni nell'adozione del Design Thinking. Con stakeholder interni si intende i team di sviluppo software mentre con stakeholder esterni si fa riferimento ai clienti e gli utenti finali che devono essere coinvolti attivamente nel processo di progettazione e sviluppo per garantire che il prodotto finale soddisfi le esigenze degli utenti e siano al passo con le tendenze del mercato. Nella ricerca è stata sottolineata, infatti, l'importanza di considerare l'aspetto

umano nella progettazione software; la partecipazione attiva degli utenti e il coinvolgimento delle parti interessate sono fondamentali per il successo del prodotto. L'empatia e la comprensione delle esigenze degli utenti sono cruciali per identificare i problemi e creare soluzioni efficaci. Questo aspetto potrebbe essere il più costoso in termini di tempo e risorse economiche ma garantirà un'implementazione del software molto più lineare, rapida e con risultati nettamente migliori.

Infine, l'integrazione del Design Thinking richiede l'adozione di strumenti e tecniche specifiche. Ciò richiede la formazione dei professionisti in modo che possano acquisire le competenze necessarie per applicare di tale metodologia correttamente. Questo aspetto potrebbe essere risolto, con il tempo, inserendo questo tipo di insegnamenti durante la formazione dei professionisti e incentivandoli a partecipare a corsi e workshop.

Se implementato correttamente, l'integrazione del Design Thinking può portare a prodotti software di alta qualità, di successo e soddisfacenti dal punto di vista delle esigenze degli utenti e del mercato.

Bisogna, inoltre, riconoscere che il Design Thinking è un approccio in continua evoluzione e miglioramento. Nuove tecniche vengono sviluppate costantemente per affrontare le nuove sfide della progettazione. Per questo motivo, è importante avere un atteggiamento aperto ed essere propensi all'apprendimento e all'adozione di nuovi approcci e strumenti.

Il Design Thinking è una metodologia innovativa che si basa sulla progettazione centrata sull'utente con lo scopo di sviluppare soluzioni creative e funzionali a problemi complessi. Negli ultimi anni, il Design Thinking si è diffuso in molte aree, dalla tecnologia alla sanità, ma ci sono ancora molti ambiti in cui può essere applicato per creare soluzioni ancora più avanzate ed efficaci. Due di questi ambiti sono l'accessibilità e l'intelligenza artificiale.

Nel primo caso, l'accessibilità è diventata un argomento sempre più importante in quanto le tecnologie e i servizi digitali sono sempre più diffusi nella vita quotidiana. L'intelligenza artificiale, d'altra parte, sta rapidamente diventando una delle tecnologie più importanti del nostro tempo e potrebbe avere un enorme impatto sulla progettazione di prodotti e servizi.

Per quanto riguarda più nello specifico l'accessibilità negli ultimi anni sta diventando più comune per i professionisti porsi domande come *"Stiamo coinvolgendo le persone giuste?"* o *"Chi potrebbe essere trascurato in questa soluzione e come potremmo risolvere?"*. Si nota quindi una tendenza generale a cercare soluzioni innovative non solo chiedendosi *"come"* ma anche concentrandosi su *"chi"* [Con19]. Ad esempio, si potrebbe utilizzare il Design Thinking per creare interfacce più intuitive e accessibili per le persone con disabilità, sfruttando la vera forza del Design Thinking, ovvero il fatto di essere un modello centrato sull'utente, adattabile a qualsiasi situazione e gruppo di individui.

Passando invece all'intelligenza artificiale, il Design Thinking può essere utilizzato per sviluppare soluzioni che migliorino l'esperienza utente nell'ambito dell'intelligenza artificiale. La progettazione di sistemi di intelligenza artificiale richiede competenze diverse da quelle necessarie per lo sviluppo di altri sistemi software, questo perché l'IA non segue regole, schemi e comportamenti prevedibili. Questo significa che è necessario sviluppare soluzioni che tengano piena considerazione del comportamento umano e capire in modo approfondito le emozioni, le necessità e i pensieri degli utenti che interagiranno con il sistema [Owc21]. Il Design Thinking risulta di conseguenza perfetto per queste esigenze e potrebbe facilitare, ad esempio, la creazione di sistemi più trasparenti o etici, due degli elementi più discussi quando si parla di intelligenza artificiale.

In breve, l'accessibilità e l'intelligenza artificiale sono due ambiti in cui il Design Thinking può essere utilizzato per creare soluzioni avanzate e adattabili alle esigenze degli utenti, migliorando così l'esperienza complessiva degli utenti e la loro interazione con le tecnologie moderne.

Bibliografia

- [Ard+14] Carmelo Ardito et al. «Investigating and promoting UX practice in industry: An experimental study». In: *International Journal of Human-Computer Studies* 72.6 (2014), pp. 542–551.
- [Bho+15] Tanmay Bhowmik et al. «Leveraging topic modeling and part-of-speech tagging to support combinational creativity in requirements engineering». In: *Requirements Engineering* 20 (2015), pp. 253–280.
- [Bor+15] Simone Borsci et al. «Assessing user satisfaction in the era of user experience: Comparison of the SUS, UMUX, and UMUX-LITE as a function of product experience». In: *International journal of human-computer interaction* 31.8 (2015), pp. 484–495.
- [Bou+22] Hamid Bouraghi et al. «Evaluating the usability of a national health information system with heuristic method». In: *Journal of Education and Health Promotion* 11 (2022).
- [Bro96] John Brooke. «SUS – a quick and dirty usability scale». In: gen. 1996, pp. 189–194.
- [Con19] Taylor Cone. *What’s Next for Design Thinking*. Set. 2019. URL: <https://modus.medium.com/whats-next-for-design-thinking-d44bebbb7649>.
- [Dar+22] Irfan Darmawan et al. «Design Thinking Approach for User Interface Design and User Experience on Campus Academic Information Systems». In: *JOIV: International Journal on Informatics Visualization* 6.2 (2022), pp. 327–334.
- [Fin06] Kraig Finstad. «The system usability scale and non-native English speakers». In: *Journal of usability studies* 1.4 (2006), pp. 185–188.
- [LS+06] Michael O Leavitt, Ben Shneiderman et al. «Based web design & usability guidelines». In: *Background and Methodology* (2006).
- [LT04] Talia Lavie e Noam Tractinsky. «Assessing dimensions of perceived visual aesthetics of web sites». In: *International journal of human-computer studies* 60.3 (2004), pp. 269–298.
- [NMH95] Don Norman, Jim Miller e Austin Henderson. «What you see, some of what’s in the future, and how we go about doing it: HI at Apple Computer». In: *Conference companion on Human factors in computing systems*. 1995, p. 155.

- [NN21a] Winda Suci Lestari Nasution e Patriot Nusa. «UI/UX design web-based learning application using design thinking method». In: *ARRUS Journal of Engineering and Technology* 1.1 (2021), pp. 18–27.
- [NN21b] Don Norman e Jakob Nielsen. *The Definition of User Experience (UX)*. Nielsen Norman Group, 2021. URL: <https://www.nngroup.com/articles/definition-user-experience/>.
- [Owc21] Dorota Owczarek. *Applying Design Thinking to Artificial Intelligence. Why Should You Use It in Your AI-Based Projects?* Set. 2021. URL: <https://nexocode.com/blog/posts/applying-design-thinking-to-ai/>.
- [Per+21] Lauriane Pereira et al. «Towards an understanding of benefits and challenges in the use of design thinking in requirements engineering». In: *Proceedings of the 36th Annual ACM Symposium on Applied Computing*. 2021, pp. 1338–1345.
- [Rau+13] Maria Rauschenberger et al. «Efficient measurement of the user experience of interactive products. How to use the user experience questionnaire (UEQ). Example: Spanish language version». In: (2013).
- [RPC+22] Arroyan Cahyaning Ratri, Eli Pujastuti, Nuri Cahyono et al. «UI Pattern Analysis in Obtaining Highest Success Rate in University Admission Website». In: *2022 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS)*. IEEE. 2022, pp. 314–319.
- [Rus+15] Cristian Rusu et al. «Usability and user experience: What should we care about?». In: *International Journal of Information Technologies and Systems Approach (IJITSA)* 8.2 (2015), pp. 1–12.
- [Sau15] Jeff Sauro. «SUPR-Q: A comprehensive measure of the quality of the website user experience.» In: *Journal of usability studies* 10.2 (2015).
- [Sch19] Martin Schrepp. *User Experience Questionnaire Handbook: All you need to know to apply the UEQ successfully in your projects*. Dic. 2019. URL: <https://www.ueq-online.org>.
- [SHT14] Martin Schrepp, Andreas Hinderks e Jörg Thomaschewski. «Applying the user experience questionnaire (UEQ) in different evaluation scenarios». In: *Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience: Third International Conference, DUXU 2014, Held as Part of HCI International 2014, Heraklion, Crete, Greece, June 22-27, 2014, Proceedings, Part I* 3. Springer. 2014, pp. 383–392.
- [SHT17] Martin Schrepp, Andreas Hinderks e Jörg Thomaschewski. «Design and evaluation of a short version of the user experience questionnaire (UEQ-S)». In: *International Journal of Interactive Multimedia and Artificial Intelligence*, 4 (6), 103-108. (2017).

- [SL11] Jeff Sauro e James R Lewis. «When designing usability questionnaires, does it hurt to be positive?» In: *Proceedings of the SIGCHI conference on human factors in computing systems*. 2011, pp. 2215–2224.
- [Sta22] Start.io. *Spotify Target Market Segmentation – User Demographics Audience Targeting Strategy for 2022*. Giu. 2022. URL: <https://www.start.io/blog/spotify-target-market-segmentation-user-demographics-audience-targeting-strategy/>.

Ringraziamenti

I miei ringraziamenti vanno alla mia famiglia, il mio punto di riferimento in ogni circostanza, nella luce e nel buio. Grazie per avermi sostenuta e per avermi insegnato a non arrendermi mai, ricordandomi sempre che "volere è potere".

Alle mie amiche Carlotta, Chiara, Elisa, Asia e Giorgia (in ordine di apparizione nella mia vita) che pazientemente mi avete accompagnata nel mio percorso. Forse amiche è riduttivo perché per me siete come sorelle. Grazie per gli anni trascorsi insieme, fatti di qualche lacrima, ma anche di tante risate.

Ai miei colleghi, voi siete riusciti a rendere speciali e indimenticabili questi anni universitari. Anche se la nostra amicizia è recente, mi sembra di conoscervi da tutta la vita, come se fossimo stati destinati a incontrarci.

Alle Pinne da Passeggio, siete state al mio fianco sia da vicino che da lontano, facendomi sempre sentire il vostro affetto. Mi auguro, ma sono certa che sarà così, che ci saremo sempre le une per le altre.

All'Università di Bologna e al mio relatore Fabio Vitali che mi hanno dato questa impagabile opportunità di apprendimento e di crescita.

E a tutte le belle persone che ho incontrato nel mio cammino e che mi hanno resa quella che sono oggi.

Hardship awaits me at every step, but I will run, I believe in myself