

# IUM - Interazione Uomo Macchina

Marco Ferrara

Ottobre 2024

# Contents

<b>1</b>	<b>Introduzione</b>	<b>2</b>
1.1	Interfaccia Utente . . . . .	2
1.2	HCI – Human-Computer Interaction . . . . .	2
<b>2</b>	<b>Accettabilità di un sistema software</b>	<b>5</b>
2.1	Definizione di Jacob Nielsen - 1993 . . . . .	5
2.2	Definizione di ISO 9241-11 - 1998 . . . . .	6
2.3	Confronto tra le due definizioni . . . . .	7
2.4	User eXperience (UX) nell'ISO 9241-210 . . . . .	7
2.5	Obiettivi dell'usabilità . . . . .	8
2.6	Dalla Usability alla User eXperience . . . . .	8
2.7	Migliorare l'interfaccia utente . . . . .	9
2.8	Come ottenere l'usabilità . . . . .	9
2.9	Human-Centred Design Process - ISO 9241-210 . . . . .	10
<b>3</b>	<b>Requisiti, tecniche di raccolta dati e analisi dei task</b>	<b>12</b>
3.1	Requisiti di prodotto . . . . .	12
3.2	Tecniche di raccolta dati . . . . .	14
3.2.1	Interviste . . . . .	14
3.2.2	Focus Group . . . . .	17
3.2.3	Osservazione sul campo . . . . .	18
3.2.4	Scenari d'uso . . . . .	19
3.2.5	Questionari . . . . .	20
3.2.6	Confronto di tecniche di raccolta dati . . . . .	20
3.3	Caso d'uso . . . . .	21
3.3.1	Analisi dei task . . . . .	21
3.3.2	Scenario vs Caso . . . . .	22
<b>4</b>	<b>UCD Sprint</b>	<b>23</b>
4.1	Discovery . . . . .	24

# Chapter 1

## Introduzione

L'obiettivo del corso è quello di fornire un primo orientamento sulle problematiche del **design** e della **valutazione dell'interazione uomo-macchina**, per la progettazione di sistemi interattivi **facili da usare**, discutendo le qualità del software di interesse degli **utenti finali**.

### 1.1 Interfaccia Utente

La **interfaccia utente** è il mezzo attraverso il quale l'utente interagisce con un sistema o prodotto per raggiungere obiettivi specifici. Essa funge da punto di contatto tra l'utente e il sistema, rendendo possibile la comunicazione e la gestione delle funzionalità offerte. Un'interfaccia utente efficace non solo facilita l'uso del sistema, ma si adatta ai bisogni e alle aspettative dell'utente, risultando essenziale per un'esperienza utente intuitiva e soddisfacente.

### 1.2 HCI – Human-Computer Interaction

E' la disciplina che si occupa della progettazione, valutazione e realizzazione di sistemi interattivi basati su computer destinati all'uso umano e dello studio dei principali fenomeni che li circondano.

Ha le sue origini in due aree disciplinari molto diverse:

- **Ergonomia**: si occupa dei problemi relativi al lavoro umano in rapporto alla progettazione delle macchine e degli ambienti di lavoro, cercando di ottimizzare il comfort, la sicurezza e l'efficienza dell'utente.
- **Scienza dei Computer**: si occupa della progettazione e realizzazione di sistemi software, includendo sia aspetti teorici che pratici come l'architettura dei sistemi e la programmazione.

I principali temi su cui si basa l'HCI sono:

- Metodologie e processi per la **progettazione** delle interfacce fra uomo e computer: studio e definizione delle migliori pratiche per creare interfacce intuitive e accessibili.
- Metodi e strumenti per la **realizzazione** delle interfacce: utilizzo di tecniche di sviluppo e software specifici per implementare le interfacce progettate.
- Tecniche per la **valutazione** e il confronto delle interfacce: approcci empirici e analitici per misurare l'efficacia e l'efficienza delle interfacce in contesti reali o simulati.
- Progettazione di nuove **tecniche di interazione**: esplorazione di modalità innovative per interagire con i sistemi, come interfacce vocali, touch, o realtà aumentata.
- Sviluppo di **teorie** e **modelli** descrittivi e predittivi dell'interazione uomo-computer: costruzione di modelli teorici per comprendere e prevedere i comportamenti degli utenti durante l'interazione con i sistemi.
- ...

L'Human-Computer Interaction (HCI) è una disciplina che nasce dall'integrazione di diverse aree del sapere, ognuna delle quali contribuisce alla sua complessità e alla sua applicazione pratica. In particolare, l'HCI trae ispirazione e strumenti da tre principali campi disciplinari: la **Scienza dei computer**, la **Scienza della progettazione** e le **Scienze dell'uomo**:

- **Scienza dei computer**: In quest'area si trovano i paradigmi di interazione, che definiscono il modo in cui gli utenti interagiscono con i dispositivi digitali. Comprende anche lo sviluppo di device di interazione e la programmazione delle interfacce utente (UI programming). Inoltre, si studiano i modelli di dialogo tra uomo e macchina, la computer graphics e le tecniche di visualizzazione delle informazioni, oltre all'intelligenza artificiale, che offre nuove possibilità di interazione e automazione.
- **Scienza della progettazione**: Qui rientrano l'interaction design e l'industrial design, che si concentrano sulla progettazione ergonomica e funzionale di sistemi e interfacce. Si studiano anche l'architettura dell'informazione, le tecniche di valutazione dell'usabilità e dell'esperienza utente, e il project management, che garantisce la corretta gestione dei progetti legati alla progettazione di interfacce.
- **Scienze dell'uomo**: Questa area include la psicologia, le scienze cognitive, la psicologia sociale, la linguistica e le scienze della comunicazione. Queste discipline studiano il comportamento umano, i processi mentali, le dinamiche sociali e linguistiche, e come queste influenzano l'interazione con i sistemi informatici.

L'HCI si posiziona dunque al centro di queste tre grandi aree, con l'obiettivo di creare interfacce utente che siano non solo funzionali e tecnologicamente avanzate, ma anche ottimizzate per l'uso umano, tenendo conto dei bisogni cognitivi, fisici e sociali dell'utente. Così facendo va a fondere due discipline complicate con background culturali molto diversi: la psicologia e la programmazione.

## Chapter 2

# Accettabilità di un sistema software

L'accessibilità di un sistema software è la capacità di essere utilizzato da quante più persone possibili, indipendentemente dalle loro capacità fisiche, cognitive o sensoriali.

La sua definizione si è evoluta nel tempo vedendo due principali accezioni riconosciute ufficialmente dalla comunità scientifica.

### 2.1 Definizione di Jacob Nielsen - 1993

L'accettabilità di un sistema software, secondo Jacob Nielsen, è la misura in cui un software viene considerato idoneo all'uso dagli utenti finali. Tale accettabilità è influenzata da diversi fattori, che possono essere suddivisi in due categorie principali: le caratteristiche generali del sistema e le proprietà legate all'usabilità.

- **Costo:** che deve essere commisurato alle risorse disponibili e al valore percepito del software.
- **Compatibilità:** con altri sistemi e dispositivi con cui si interfaccia, facilitando la sua integrazione nell'ambiente lavorativo o personale.
- **Affidabilità:** Il sistema deve essere stabile e funzionare senza problemi, garantendo un basso tasso di malfunzionamenti o errori.
- **Utilità:** Rappresenta la capacità del software di soddisfare i bisogni pratici dell'utente, fornendo le funzionalità necessarie per raggiungere gli obiettivi prefissati.
- **Utilizzabilità:** Questa componente riguarda il grado di facilità con cui gli utenti possono utilizzare il sistema. Si divide a sua volta in diversi fattori chiave legati all'usabilità.

L'usabilità si compone di più fattori:

- **Facilità di apprendimento:** Indica quanto rapidamente un nuovo utente può apprendere l'uso del software, riducendo il tempo necessario per diventare operativo.
- **Facilità d'uso:** Si riferisce alla semplicità di interazione quotidiana con il sistema, che non dovrebbe richiedere sforzi eccessivi per compiere operazioni comuni.
- **Facilità di memorizzazione:** Riguarda la capacità dell'utente di ricordare facilmente come utilizzare il sistema anche dopo lunghi periodi di inattività.
- **Numero ridotto di errori commessi dall'utente:** Un sistema ben progettato dovrebbe minimizzare la possibilità che l'utente commetta errori e, in caso di errore, offrire una facile via di recupero.
- **Soddisfazione nell'uso:** L'utente deve provare una sensazione positiva nell'uso del sistema, che dovrebbe risultare piacevole e gratificante.

## 2.2 Definizione di ISO 9241-11 - 1998

La nuova versione elabora i concetti di versioni precedenti e li espande, introducendo nuovi concetti e definizioni. In particolare, l'ISO 9241-11 definisce l'usabilità come

“ la misura in cui un **sistema**, prodotto o **servizio** può essere utilizzato da utenti *specifici* per raggiungere obiettivi *specifici* con efficacia, efficienza e soddisfazione in uno *specifico* contesto d'uso. ”

L'usabilità, in questo caso, non è più un concetto assoluto ma relativo, che dipende da vari fattori come gli utenti, gli obiettivi e il contesto d'uso.

Questa definizione introduce tre nuovi concetti chiave:

- **Efficacia:** Indica la capacità del sistema di consentire all'utente di raggiungere i propri obiettivi in modo accurato e completo senza conseguenze negative.
- **Efficienza:** Si riferisce alla capacità del sistema di consentire all'utente di raggiungere i propri obiettivi con un impiego minimo di risorse, come tempo, sforzo e costi.
- **Soddisfazione:** Rappresenta il grado di piacere e gratificazione che l'utente prova nell'utilizzare il sistema, derivante da una combinazione di fattori come l'usabilità, l'estetica e l'esperienza d'uso (UX).

## 2.3 Confronto tra le due definizioni

Le due definizioni di accettabilità hanno diverse caratteristiche, ma sono anche diverse in conseguenza. La prima definizione è basata su una visione più generale e pratica dell'accettabilità, che include aspetti come costo, compatibilità e affidabilità, oltre alle proprietà legate all'usabilità. La seconda definizione, invece, è più specifica e focalizzata sull'usabilità, definendo l'accettabilità come la capacità di un sistema di consentire agli utenti di raggiungere i propri obiettivi in modo efficace, efficiente e soddisfacente.

Il confronto tra le due definizioni può essere facilmente illustrata con un esempio. Supponiamo di considerare due sistemi software, A e B, che hanno le seguenti caratteristiche:

- Sistema A: è molto costoso ma altamente affidabile e compatibile con altri sistemi. Tuttavia, è difficile da imparare e usare, e gli utenti commettono frequentemente errori nell'interazione con il sistema.
- Sistema B: è economico, facile da imparare e usare, e gli utenti sono soddisfatti dell'esperienza d'uso. Tuttavia, è meno affidabile e compatibile rispetto al sistema A.

Secondo la definizione di Jacob Nielsen, il sistema A potrebbe essere considerato più accettabile rispetto al sistema B, poiché ha caratteristiche come costo, affidabilità e compatibilità che lo rendono idoneo all'uso. Tuttavia, secondo la definizione ISO 9241-11, il sistema B potrebbe essere considerato più accettabile, poiché consente agli utenti di raggiungere i propri obiettivi in modo efficace, efficiente e soddisfacente, nonostante le sue limitazioni in termini di affidabilità e compatibilità.

In conclusione, entrambe le definizioni forniscono un quadro utile per valutare l'accettabilità di un sistema software, ma si concentrano su aspetti diversi e possono portare a conclusioni diverse a seconda del contesto e degli obiettivi specifici dell'utente.

## 2.4 User eXperience (UX) nell'ISO 9241-210

L'ISO 9241-210 definisce l'esperienza utente (UX) come “le percezioni e le reazioni di un utente che derivano dall'uso o dall'aspettativa d'uso di un prodotto, sistema o servizio”. Questa definizione sottolinea l'importanza di considerare non solo l'usabilità di un sistema, ma anche l'esperienza complessiva dell'utente durante l'interazione con esso. L'esperienza utente include le emozioni dell'utente, le sue preferenze, le reazioni psicologiche e fisiche e i comportamenti che si verificano prima, durante e dopo l'uso del sistema.



## 2.5 Obiettivi dell'usabilità

L'usabilità è un concetto multidimensionale che può essere valutato attraverso diversi obiettivi, tra cui:

- **Efficacia:** Misura la capacità del sistema di consentire all'utente di raggiungere i propri obiettivi in modo accurato e completo senza conseguenze negative.
- **Efficienza:** Si riferisce alla capacità del sistema di consentire all'utente di raggiungere i propri obiettivi con un impiego minimo di risorse, come tempo, sforzo e costi.
- **Sicurezza nell'uso:** Indica la capacità del sistema di proteggere l'utente da situazioni di rischio o pericolo durante l'interazione, evitando errori o danni accidentali.
- **Soddisfazione:** Rappresenta il grado di piacere e gratificazione che l'utente prova nell'utilizzare il sistema, derivante da una combinazione di fattori come l'usabilità, l'estetica e l'esperienza d'uso (UX).
- **Facilità di apprendimento:** Indica quanto rapidamente un nuovo utente può apprendere l'uso del software, riducendo il tempo necessario per diventare operativo.
- **Facilità d'uso:** Si riferisce alla semplicità di interazione quotidiana con il sistema, che non dovrebbe richiedere sforzi eccessivi per compiere operazioni comuni.
- **Facilità di memorizzazione:** Riguarda la capacità dell'utente di ricordare facilmente come utilizzare il sistema anche dopo lunghi periodi di inattività.
- **Presenza di features:** Indica la presenza di funzionalità e caratteristiche che facilitano l'uso del sistema e migliorano l'esperienza dell'utente.

Sicuramente, la UX non deve portare l'utente a provare sentimenti negativi come noia, frustrazione, sensi di colpa, confusione e così via. Altresì, l'utente deve provare sentimenti positivi come divertimento, soddisfazione, orgoglio, eccitazione, ecc.

## 2.6 Dalla Usability alla User eXperience

La User eXperience (UX) è un concetto più ampio e complesso rispetto all'usabilità, poiché include non solo la facilità d'uso di un sistema, ma anche le emozioni, le preferenze e le reazioni dell'utente durante l'interazione con esso. L'usabilità è un aspetto fondamentale della UX, ma non è l'unico fattore che contribuisce alla qualità complessiva dell'esperienza utente.

Se l'efficienza mira a ridurre il tempo e lo sforzo necessari per raggiungere un obiettivo, in maniera semplice e senza errori, la UX coinvolge i sentimenti, i quali possono essere influenzati da attributi funzionali del sistema, come la portabilità o la robustezza, ma anche da attributi non funzionali, come l'usabilità e la privacy, fino ad arrivare a fattori specifici come il piacere nell'utilizzare il sistema, la motivazione o l'estetica.

Un prodotto capace di generare una buona UX è in grado di soddisfare le esigenze e le aspettative dell'utente e dovrebbe essere utile, usabile e desiderabile. Per realizzare prodotti di successo, la UX enfatizza sul piacere dell'utente, sull'estetica e sul divertimento. C'è una correlazione tra l'estetica e la percezione della qualità dell'interfaccia. Interfacce appetibili dal punto di vista estetico sono percepite come più utili anche se lo sono meno rispetto a interfacce con funzionalità simili ma meno attraenti.

## 2.7 Migliorare l'interfaccia utente

Migliorare l'interfaccia utente di un sistema software è un processo continuativo che ha come scopo quello di rendere il sistema più facile da usare, più efficace e più soddisfacente per l'utente. Se pensiamo a questo processo in ambito business, possiamo dire che un'interfaccia ben progettata può portare a una maggiore produttività, a una riduzione degli errori e dei costi, a un aumento della soddisfazione dell'utente e, di conseguenza, a una maggiore competitività sul mercato.

Di conseguenza, la progettazione e la valutazione dell'interfaccia utente dovrebbero essere parte integrante del processo di sviluppo del software, dall'inizio alla fine, coinvolgendo gli utenti fin dall'inizio per garantire che le loro esigenze e aspettative siano soddisfatte. L'usabilità e la user experience dovrebbero essere valutate in modo continuativo durante tutto il ciclo di vita del prodotto per identificare e risolvere i problemi di usabilità e migliorare l'esperienza dell'utente.

Basarsi su linee guida e best practice di design, condurre test di usabilità e coinvolgere gli utenti nel processo di progettazione sono solo alcune delle strategie che possono essere adottate per migliorare l'interfaccia utente di un sistema software.

## 2.8 Come ottenere l'usabilità

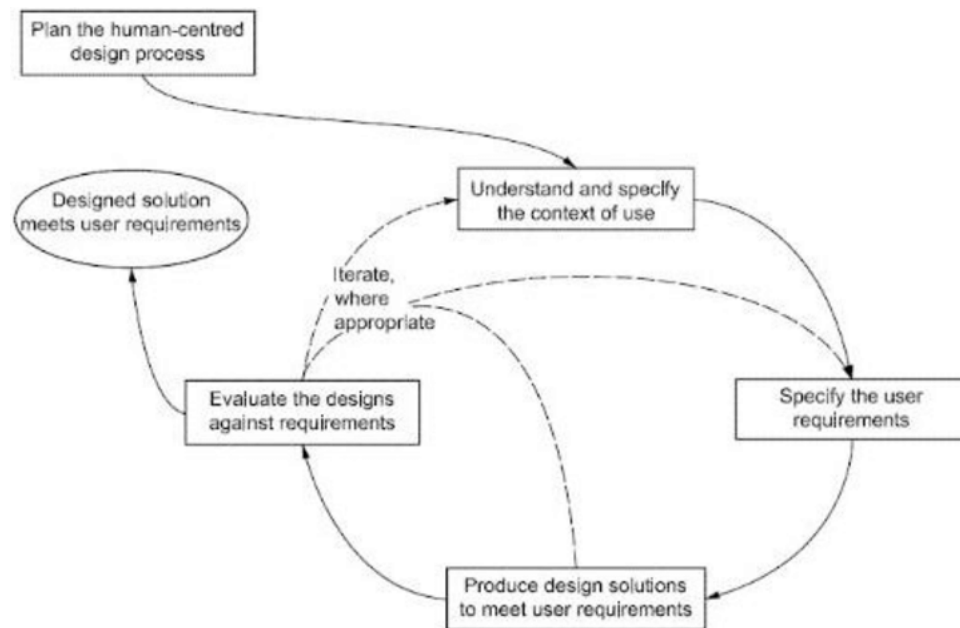
Per ottenere un'interfaccia utente usabile e una user experience soddisfacente, è necessario seguire alcune linee guida e best practice di design. Il punto di partenza è centrare il progetto sull'utente (User-Centred Design) andando a capire l'utente stesso, i suoi compiti e il contesto d'uso, sviluppando iterativamente prototipi e testandoli con gli utenti per valutare l'usabilità e l'esperienza

d'uso. Inoltre, è importante adottare un approccio multidisciplinare, coinvolgendo esperti di diverse discipline come la psicologia, il design, l'informatica e l'ingegneria, per garantire che l'interfaccia utente sia progettata in modo efficace e soddisfacente per l'utente.

Queste linee guida sono specificate in vari standard e raccomandazioni, come l'ISO 9241-210, che fornisce linee guida per la progettazione Human-Centred.

## 2.9 Human-Centred Design Process - ISO 9241-210

L'Human-Centred Design Process (HCD) è un approccio iterativo alla progettazione che mette l'utente al centro del processo, coinvolgendo gli utenti fin dall'inizio per garantire che le loro esigenze e aspettative siano soddisfatte.



Il processo HCD si divide in 6 fasi:

1. **Plan the human centred design process:** Definire gli obiettivi del progetto, identificare gli utenti e le loro esigenze, e pianificare le attività di progettazione e valutazione. Definendo il target di riferimento si possono identificare le caratteristiche necessarie per soddisfare le esigenze degli utenti, i metodi di coinvolgimento all'interno della progettazione e di contatto con gli stessi e il metodo di raccolta delle informazioni da utilizzare.

2. **Understand and specify the context of use:** Una volta definito il target e le caratteristiche, si procede con l'analisi del contesto d'uso, ovvero l'ambiente in cui il sistema verrà utilizzato, le attività che gli utenti dovranno svolgere e le caratteristiche degli utenti stessi. Questa fase è fondamentale per comprendere le esigenze e le aspettative degli utenti e per definire i requisiti del sistema.
3. **Specify the user requirements:** Definire i requisiti funzionali e non dell'utente in base all'analisi del contesto d'uso, identificando le funzionalità e le caratteristiche che il sistema dovrà avere per soddisfare le esigenze degli utenti. Questi requisiti dovrebbero essere specifici, misurabili e verificabili, e dovrebbero essere utilizzati come base per la progettazione del sistema.
4. **Produce design solutions to meet user requirements:** Sviluppare soluzioni di design che soddisfino i requisiti dell'utente, utilizzando tecniche di progettazione come la creazione di wireframes, mockup e prototipi. Queste soluzioni dovrebbero essere valutate con gli utenti per garantire che siano efficaci, efficienti e soddisfacenti. I prototipi non devono necessariamente essere completi, ma devono essere sufficientemente dettagliati per permettere agli utenti di valutare l'interfaccia, possono addirittura essere schizzi o storyboard.
5. **Evaluate designs against requirements:** Valutare le soluzioni di design rispetto ai requisiti dell'utente, utilizzando tecniche di valutazione come i test di usabilità, le valutazioni euristiche e le analisi cognitive. Queste valutazioni dovrebbero essere condotte con gli utenti per identificare i problemi di usabilità e migliorare l'esperienza dell'utente. I test di usabilità possono essere condotti in laboratorio, in campo o in remoto, e possono coinvolgere diversi tipi di utenti, come utenti esperti o utenti novizi. Da questo punto, in base agli esiti delle valutazioni, si può tornare alla fase 2 per rivedere il contesto d'uso e ripetere il processo; oppure alla fase 3 per rivedere i requisiti utente e ripetere il processo; oppure alla fase 4 per rivedere le soluzioni di design e ripetere il processo; oppure avanzare alla fase 6.
6. **Designed solutions meet user requirements:** Implementare le soluzioni di design finali che soddisfino i requisiti dell'utente, utilizzando tecniche di sviluppo come la programmazione e la progettazione dell'interfaccia utente. Queste soluzioni dovrebbero essere testate con gli utenti per garantire che siano efficaci, efficienti e soddisfacenti, e dovrebbero essere valutate in un contesto d'uso reale per verificare che soddisfino le esigenze degli utenti.

## Chapter 3

# Requisiti, tecniche di raccolta dati e analisi dei task

### 3.1 Requisiti di prodotto

I requisiti di prodotto sono le specifiche funzionali e non funzionali che definiscono le caratteristiche e le prestazioni del prodotto. Vengono raccolti in un documento strutturato che fornisce una descrizione dettagliata delle funzionalità, delle prestazioni e delle caratteristiche del prodotto, e che viene utilizzato come base per la progettazione e lo sviluppo del prodotto stesso. Essi vengono identificati attraverso analisi condotte con varie tecniche.

I requisiti di prodotto possono essere divisi in due categorie principali: requisiti funzionali e requisiti non funzionali. I requisiti funzionali definiscono le funzionalità che il sistema deve implementare per soddisfare le esigenze dell'utente, mentre i requisiti non funzionali definiscono le caratteristiche del sistema che non sono direttamente legate alle funzionalità, ma che influenzano la sua usabilità, la sua sicurezza e le sue prestazioni.

I requisiti di prodotto possono essere definiti mediante diversi tipi di analisi:

- **Analisi dell'utente:** qual è il target di utenza del prodotto.
- **Analisi dei bisogni:** quali sono i bisogni e le esigenze degli utenti.
- **Analisi del contesto d'uso:** in quali contesti il prodotto verrà utilizzato.
- **Analisi degli scenari d'uso:** in quali modi diversi gli utenti interagiranno con il prodotto.

- **Analisi della concorrenza:** quali sono i prodotti simili presenti sul mercato. Quali sono i punti di forza e di debolezza rispetto ai prodotti concorrenti.

Si precisa che individuare e specificare i requisiti non significa progettare il prodotto, ma definire cosa il prodotto deve fare e come deve comportarsi. I requisiti pongono dei vincoli all'attività di progettazione. Il documento dei requisiti indica che cosa si deve realizzare e perché, non come.

La definizione dei requisiti può essere sintetizzata come nello schema che segue:



Si parte dalle richieste del committente, che includono le aspettative e le necessità per il prodotto o servizio da sviluppare. Da qui, si avvia la fase di esplorazione, durante la quale si conducono interviste con gli stakeholder, si analizzano prodotti simili già esistenti e si valuta la concorrenza. Tutte queste informazioni raccolte diventano appunti e materiali di riferimento.

Successivamente, si passa alla organizzazione, dove questi dati vengono rielaborati per redigere un documento che descrive i requisiti necessari. Le linee guida servono come riferimento per questa stesura. Il risultato finale è un elenco di requisiti chiari e strutturati.

Infine, questi requisiti vengono sottoposti a una fase di revisione e approvazione, dove si valuta se soddisfano le esigenze iniziali e, una volta approvati, si può procedere alle attività di progettazione e sviluppo.

La raccolta dei requisiti è un'attività fondamentale e critica per il successo di un progetto. Durante questa fase, si possono presentare diversi problemi come:

- il rischio di ampliare/restringere eccessivamente il campo di applicazione del progetto o i temi da approfondire;
- incomprensioni con gli utenti che spesso hanno una conoscenza limitata dei loro bisogni; o con chi raccoglie i requisiti che può non avere una conoscenza approfondita del dominio applicativo e può utilizzare un linguaggio differente da quello degli utenti;
- conflitto tra gli interessi dei vari stakeholder che possono avere esigenze contrastanti;

- volatilità dei requisiti, che possono cambiare nel tempo a causa di nuove esigenze o di nuove tecnologie.

Gli utenti sono coloro che decidono se utilizzare il prodotto, che hanno preferenze, abitudini e aspettative ogni qualvolta si vada ad utilizzare un prodotto. Non supportare le loro esigenze può provocare alti costi in termini di produttività per l'azienda e di frustrazione per l'utente a tal punto da mettere a rischio la riuscita del progetto.

## 3.2 Tecniche di raccolta dati

La raccolta dei dati è un'attività fondamentale per la definizione dei requisiti di un prodotto. Esistono diverse tecniche di raccolta dati che possono essere utilizzate per raccogliere informazioni sui bisogni e le esigenze degli utenti, sul contesto d'uso del prodotto e sulle funzionalità richieste. Alcune delle tecniche più comuni sono:

- **Interviste:** Conversazioni strutturate o semi-strutturate con gli utenti per raccogliere informazioni sui loro bisogni, aspettative e comportamenti.
- **Questionari:** Questionari strutturati per raccogliere informazioni quantitative sui bisogni e le preferenze degli utenti.
- **Osservazioni:** Osservazioni dirette degli utenti durante l'utilizzo del prodotto per identificare problemi di usabilità e migliorare l'esperienza dell'utente.
- **Focus group:** Gruppi di discussione con un piccolo gruppo di utenti per raccogliere informazioni qualitative sui loro bisogni e aspettative.
- **Analisi documentale:** Analisi di documenti esistenti come report, studi di mercato e analisi della concorrenza per identificare le esigenze degli utenti e le funzionalità richieste.
- **Analisi della concorrenza:** Analisi dei prodotti simili già presenti sul mercato per identificare i punti di forza e di debolezza e le funzionalità richieste.
- **Suggerimenti e feedback degli utenti:** Raccolta di suggerimenti e feedback dagli utenti attraverso canali come i social media, i forum online e i servizi di assistenza clienti.

### 3.2.1 Interviste

L'intervista è una tecnica di raccolta dati che coinvolge una conversazione tra un intervistatore e un intervistato per raccogliere informazioni sui bisogni, le aspettative e i comportamenti dell'utente. Le interviste possono essere strutturate, semi-strutturate o flessibili.

Le interviste strutturate seguono un elenco di domande predefinite e sono utilizzate per raccogliere informazioni specifiche in modo coerente e ripetibile e sono spesso utilizzate per condurre indagini di mercato, studi statistici o confrontare opinioni di soggetti diversi. Le interviste flessibili si basano su un argomento conduttore ma non prevedono una sequenza fissa di domande, garantendo un flusso di conversazione più libero, ed effettuando domande di approfondimento in base alle risposte dell'intervistato. Le interviste semi-strutturate sono un compromesso tra le due, con un elenco di domande predefinite che l'intervistatore utilizza in caso di digressione della conversazione o di risposte vaghe.

Per preparare un'intervista occorre definire l'obiettivo dell'intervista e le informazioni che si vogliono reperire; selezionare il target dell'intervista; definire la struttura di essa; dove svolgerla; come raccogliere, memorizzare e analizzare i dati:

- **Obiettivo:** Definire l'obiettivo dell'intervista e le informazioni che si vogliono reperire significa:
  - identificare i bisogni dell'utente, identificando l'obiettivo che essi vogliono perseguire attraverso l'applicazione che si vuole sviluppare;
  - identificare il modello di comportamento dell'utente, ovvero quali azioni eseguono per raggiungere i goal, quali sono le priorità e le difficoltà che incontrano;
  - verificare il grado di soddisfazione degli utenti, il loro gradimento riguardo alcune feature rilevanti del sistema, quanto si sentono a loro agio nell'utilizzarlo.
- **Selezione del target:** Definire il target dell'intervista significa:
  - identificare l'utente finale del prodotto se lo scopo dell'intervista è determinare il profilo o il modello di comportamento dell'utente;
  - determinare classificazioni per ruolo, area geografica o età
  - stabilire se effettuare interviste singole o di gruppo.
- **Struttura dell'intervista:** Definire la struttura dell'intervista significa:
  - definire la struttura generale tramite un approccio top-down, da domande di carattere generale a domande dettagliate riguardanti particolari di interesse;
  - definire la struttura in dettaglio, se utilizzare domande chiuse o aperte, se utilizzare domande di controllo o di approfondimento;
  - intervista nel contesto (contextual inquiry) per osservare e intervistare l'utente mentre svolge i suoi tasks;
  - evitare domande la cui formulazione possa influenzare o preannunciare una risposta attesa;



- non chiedere valutazioni sull'importanza di un aspetto dell'applicazione, quanto cercare di capire quando tale aspetto è rilevante e perchè;
- **Luogo dell'intervista:** Definire il luogo dell'intervista significa:
  - identificare l'ambiente in cui l'utente opera, permettendo all'intervistatore di osservare direttamente il contesto d'uso e consente agli utenti di mostrare piuttosto che descrivere;
  - se l'ambiente di lavoro non è disponibile o non è rappresentativo, effettuare l'intervista in un luogo nel quale l'utente si senta a suo agio e prevedere domande per inquadrare il contesto lavorativo.
- **Raccolta, memorizzazione e analisi dei dati:** Definire come raccogliere, memorizzare e analizzare i dati significa registrare l'intervista per non perdere informazioni importanti tramite video, audio o appunti;

#### Esempio di esecuzione di un'intervista

1. *Introduction*
  - ◉ introduce yourself, explain the goals of the interview, reassure about the ethical issues, ask to record, present any informed consent form
2. *Warm-up session*
  - ◉ make first questions easy and non-threatening
3. *Main session*
  - ◉ present questions in a logical order
4. *A cool-off period*
  - ◉ include a few easy questions to defuse tension at the end
5. *Closing session*
  - thank interviewee, signal the end, e.g, switch recorder off

Le linee guida per una corretta esecuzione di un'intervista sono:

- creare un clima amichevole e informale
- non dare opinioni personali
- non anticipare le risposte
- non guidare gli utenti nelle risposte
- non presentare una serie di opzioni tra cui scegliere
- porre domande semplici e chiare
- formulare una domanda per volta
- assicurarsi che l'utente abbia compreso la domanda
- concedere all'utente il tempo necessario per rispondere

- se non si ottiene la risposta desiderata, riprovare parafrasando la domanda, senza mai condurre l'utente alla risposta da dare adattare le domande alle risposte precedenti, senza seguire rigorosamente il piano preparato per l'intervista
- evitare l'utilizzo di termini tecnici o ambigui
- raccogliere esempi
- mostrarsi spontanei, semplici e flessibili, lasciando che l'utente fornisca informazioni in un qualsiasi ordine
- ascoltare l'utente, anche quando si dilunga in altri discorsi, e cercare di riportarlo sull'argomento senza evidenziare l'accaduto

Una volta eseguita l'intervista, si procede con l'analisi dei dati raccolti strutturando gli stessi in tabelle (una per utente) con osservazioni e fatti principali ricavati da registrazioni e appunti. Successivamente si raggruppano le tabelle in base alle analogie nei dati dei vari utenti ed etichettare le categorie risultanti. Infine, si effettua un resoconto finale per ogni categoria deducendo solo ciò che si può dimostrare.

Questi risultati possono essere utilizzati nelle varie fasi di design e implementazione in modo da accertarsi che si stia procedendo secondo i bisogni dell'utente individuati tramite l'intervista. Inoltre possono usare aspetti evidenziati come spunto per l'ideazione e lo sviluppo di nuove caratteristiche dell'applicazione oppure come mezzo di risoluzione dei problemi evidenziati dall'utente durante l'intervista.

Nello specifico si possono utilizzare le interviste durante le fasi di:

- **Analisi dei requisiti:** per individuare i bisogni e le aspettative degli utenti e definire i requisiti del sistema.
- **Progetto:** per valutare le specifiche dei requisiti, le funzionalità del sistema e le soluzioni di design.
- **Implementazione:** per individuare i problemi evidenti durante l'utilizzo del sistema e valutare le modifiche necessarie.
- **Valutazione:** per valutare l'usabilità e l'esperienza d'uso del sistema e identificare i problemi di usabilità.

### 3.2.2 Focus Group

Il focus group è una tecnica di raccolta dati che coinvolge un gruppo di utenti in una discussione strutturata per raccogliere informazioni qualitative sui loro bisogni, aspettative e comportamenti sotto vari punti di vista. I focus group sono utilizzati per esplorare le opinioni e le percezioni degli utenti su un determinato argomento, per identificare i problemi e le sfide che affrontano e per generare idee e soluzioni innovative. I componenti di un focus group sono:

- **Guidatore:** guida la discussione, pone domande e stimola la conversazione.
- **Osservatore:** osserva e registra la discussione, prende appunti e raccoglie informazioni.
- **Partecipanti:** utenti che partecipano alla discussione, condividono le loro opinioni e le loro esperienze.

E' importante osservare direttamente il dialogo dei partecipanti e partecipare attivamente alla discussione in modo da mettere in relazione gli obiettivi degli utenti con i compiti e le azioni, capire come gli utenti scelgono i compiti per raggiungere i loro obiettivi, individuare i fattori ambientali e organizzativi che influenzano il loro lavoro e in che modo, e capire come gli utenti valutano il successo dei loro compiti.

Questa tecnica è ispirata dall'etnografia, ovvero la disciplina che si basa sull'osservazione diretta e la partecipazione attiva per comprendere le culture e le società umane, e può essere utilizzata per studiare i comportamenti e le pratiche degli utenti in un contesto specifico.

### 3.2.3 Osservazione sul campo

L'osservazione sul campo è una tecnica di raccolta dati che coinvolge l'osservazione diretta degli utenti durante l'utilizzo del prodotto per identificare problemi di usabilità e migliorare l'esperienza dell'utente. L'osservazione sul campo può essere condotta in vari contesti, come l'ambiente di lavoro dell'utente, la sua casa o il luogo in cui svolge le sue attività quotidiane. Per effettuare un'osservazione sul campo, è necessario:

- **Definire l'obiettivo dell'osservazione:** Identificare gli obiettivi dell'osservazione e le informazioni che si vogliono raccogliere.
- **Selezionare il contesto:** Identificare il contesto in cui si svolgerà l'osservazione e pianificare le attività da osservare.
- **Individuare i partecipanti e le sedi:** Identificare i partecipanti all'osservazione e le sedi in cui si svolgerà.
- **Programmare le visite:** Pianificare le visite sul campo e definire il tempo e la durata dell'osservazione.
- **Selezionare gli utenti da osservare:** Identificare gli utenti da osservare e ottenere il loro consenso per partecipare all'osservazione.
- **Condurre l'osservazione:** Osservare direttamente gli utenti durante l'utilizzo del prodotto e prendere appunti sui problemi di usabilità e le aree di miglioramento.

- **Raccogliere i dati:** Raccogliere i dati raccolti durante l'osservazione e organizzarli in modo da poterli analizzare e interpretare. Si potranno utilizzare media come videoregistratori, audioregistratori, fotocamere, carta e penna, moduli prestampati, ecc.

### 3.2.4 Scenari d'uso

Durante la progettazione e lo sviluppo di un prodotto potremmo tendere a concentrarci sull'oggetto della progettazione trascurando il contesto d'uso; tendiamo a vedere noi stessi come utenti tipici errando il punto di vista dell'utente o sottovalutiamo l'importanza di quest'ultimo rischiando di mancare la correttezza del prodotto.

Per evitare questi errori, è possibile utilizzare gli scenari d'uso, ovvero storie immaginarie ma tipiche di uso del sistema da parte di persone fittizie ma concrete, che rappresentano bisogni, contesti e modalità d'uso tipiche del sistema da progettare. In questo modo otteniamo contesto, concretezza e visione oggettiva mettendo in evidenza requisiti inespressi.

Gli scenari d'uso devono mettere in scena situazioni d'uso tipiche ma non ovvie, non devono contenere dettagli irrilevanti allo scopo, devono essere complete (indicando le motivazioni e le conseguenze dell'uso del prodotto nella particolare situazione) e possono essere realizzati con tecniche diverse.

**Esempio di scenario d'uso** Progettazione di un sistema di prenotazione via web per un albergo di prima categoria di Catania

- **Persona:** Luigi, ingegnere di 35 anni, sposato, lavora in una società edile. Viaggia spesso per lavoro o vacanza, in Italia e all'estero, e si tratta bene. Non è mai stato in Sicilia
- **Scenario d'uso:** Luigi deve andare a Catania per lavoro. Desidera prenotare una camera in un albergo di prima categoria vicino alla filiale della sua azienda, che si trova in centro ad un passo da Piazza del Duomo. Deve pagare con American Express intestata all'azienda per politica aziendale. Starà a Catania due notti, forse tre. Preferisce alberghi moderni e desidera una camera doppia per uso singolo.

Lo scenario contiene molti requisiti impliciti:

- visualizzare sulla mappa di Catania gli alberghi di prima categoria
- mostrare la mappa in modo che Piazza del Duomo sia facilmente individuabile
- mostrare delle fotografie dell'albergo
- permettere di prenotare camere doppie a uso singolo

- accettare pagamenti con carta American Express
- non addebitare subito l'intero importo del soggiorno per via della durata variabile.

### 3.2.5 Questionari

I questionari sono uno strumento di raccolta dati che coinvolge la somministrazione di un elenco di domande strutturate a un campione di utenti per raccogliere informazioni quantitative sui loro bisogni, aspettative e comportamenti. I questionari possono essere utilizzati per raccogliere informazioni su larga scala, per confrontare le opinioni di un gran numero di utenti e per identificare tendenze e pattern nei dati raccolti. Sono meno flessibili delle interviste poichè le domande sono predefinite e non possono essere modificate durante la somministrazione avendo minori possibilità di approfondimento.

Le domande non devono essere ambigue, devono essere chiare e semplici, il questionario deve essere breve. Nel caso contrario, si consiglia di adottare degli incentivi per la compilazione. Possono contenere domande aperte o chiuse. Sono sempre preferibili domande chiuse poichè più facili da analizzare e da somministrare.

Dopo la somministrazione del questionario, le risposte vengono convertite in dati e analizzate per identificare tendenze e pattern nei dati raccolti.

Ci sono vari tipi di domande a risposta chiusa:

- **Domande a scelta multipla:** l'utente deve scegliere una o più risposte tra diverse opzioni.
- **Scalari:** l'utente deve giudicare un'asserzione specifica in base ad una scala numerica.
- **Checklist:** semplici risposte alternative, in numero limitato.
- **Rating Scale:**
  - **Likert Scale:** l'utente deve esprimere il proprio grado di accordo o disaccordo con un'affermazione su una scala a 5 o 7 punti.
  - **Semantic Differential Scale:** l'utente deve esprimere il proprio giudizio su un concetto o un oggetto utilizzando coppie di aggettivi da incrociare.

### 3.2.6 Confronto di tecniche di raccolta dati

Le tecniche di raccolta dati presentano diversi utilizzi, vantaggi e svantaggi a seconda del metodo scelto. I questionari sono utili per rispondere a domande

specifiche e permettono di raggiungere molte persone con poco sforzo. Tuttavia, devono essere progettati con grande accuratezza, altrimenti le risposte potrebbero risultare poco informative, e il tasso di risposta può essere basso.

Le interviste individuali sono impiegate per esplorare diversi aspetti di un problema e i punti di vista correlati. Offrono il vantaggio di permettere all'intervistatore di guidare la conversazione verso i temi più rilevanti, ma richiedono molto tempo e possono non ottenere risposte schiette su argomenti delicati.

I focus group si concentrano su argomenti specifici, coinvolgendo più persone e punti di vista. Tra i vantaggi c'è la possibilità di far emergere il consenso e soluzioni condivise. D'altro canto, la loro gestione richiede esperienza e c'è il rischio che alcune persone monopolizzino la discussione.

Le osservazioni sul campo aiutano a comprendere il contesto in cui l'utente opera. Questo metodo fornisce una conoscenza diretta dell'uso reale di un prodotto, ma può risultare complesso da eseguire e richiede notevoli risorse in termini di tempo e impegno.

I suggerimenti spontanei degli utenti permettono di identificare bisogni specifici o miglioramenti per un prodotto. Questo approccio è caratterizzato da bassi costi di raccolta, ma tende a produrre dati limitati e sporadici.

Infine, l'analisi della concorrenza e delle best practices è utile per individuare le soluzioni migliori adottate in un determinato settore. Tale approccio consente di evitare di ripetere errori già noti e di ottenere un vantaggio competitivo, ma comporta costi elevati e l'impiego significativo di tempo e risorse.

### 3.3 Caso d'uso

Un caso d'uso è un insieme di interazioni finalizzate ad uno scopo utile per l'utente, fra l'utente (o più utenti) e il sistema. Ha un nome e una descrizione che ne definisce lo scopo, solitamente verbo + complemento oggetto.

Un caso d'uso può essere composto da un insieme di task, ciascuno dei quali sarà a sua volta composto da un insieme di azioni elementari. Ovviamente si parla delle azioni che compie l'utente, non di ciò che fa il sistema.

Un task è un insieme strutturato delle attività richieste, usate e ritenute necessarie per raggiungere un obiettivo. Spesso incluse sottotasks, ovvero attività che possono essere scomposte in attività più piccole.

Task e azioni possono cambiare in base alle persone che li svolgono.

#### 3.3.1 Analisi dei task

L'analisi dei task è utile per comprendere la natura del lavoro e identificare quali sono i task rilevanti considerando il coinvolgimento degli utenti, la loro analisi per capire come essi eseguono le varie attività nell'ambiente di lavoro. E' simile alla tecnica degli scenari, dalla quale sono stati eliminati il contesto ed altri dettagli.

### Tipi e Livelli

Ci sono vari tipi di analisi dei task:

- **Workflow Analysis:** analisi dei processi di lavoro per identificare i task e le attività coinvolte quando sono coinvolte più persone.
- **Job Analysis:** analisi dei processi di lavoro di un singolo individuo nell'arco di una giornata, settimana o mese lavorativo.
- **Task List:** analisi dei task che sono eseguiti da tutte le persone che potrebbero essere coinvolte nell'uso del sistema.
- **Task Sequence:** analisi dell'ordine di esecuzione dei task.
- **Procedure Analysis:** analisi delle procedure e delle scelte che sono seguite per eseguire un task o parte di esso.
- **Task Hierarchy:** analisi della struttura gerarchica dei task per comprendere come suddividere un task complesso in task più piccoli.

#### 3.3.2 Scenario vs Caso

Sono due cose diverse con obiettivi differenti.

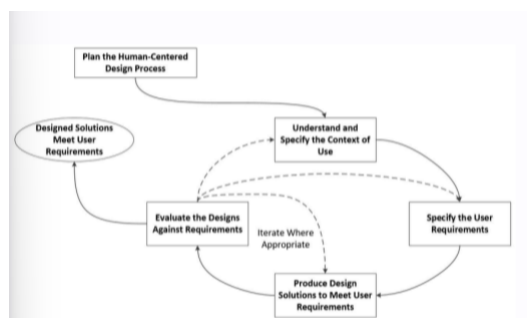
Gli scenari d'uso sono storie tipiche molto concrete di uso del sistema, raccontate con particolari per comprendere le motivazioni e il contesto e fare emergere eventuali requisiti impliciti.

I casi d'uso fanno riferimento a ruoli astratti dei vari attori e contiene solo informazioni sull'interazione che questi hanno con il sistema, senza alcuna informazione aggiuntiva sul contesto.

## Chapter 4

# UCD Sprint

L'UCD Sprint è un approccio iterativo e incrementale alla progettazione e allo sviluppo di prodotti e servizi che coinvolge gli utenti fin dall'inizio del processo di sviluppo. L'UCD Sprint si basa su un ciclo di sviluppo iterativo che prevede la realizzazione di prototipi, la valutazione con gli utenti e il miglioramento continuo del prodotto. Questo approccio consente di identificare i problemi di usabilità e migliorare l'esperienza dell'utente in modo rapido ed efficace. Si basa sul modello generale dello HCD (ISO 9241-210).



Nello specifico, ad ogni iterazione il modello suggerisce attività specifiche da fare utilizzando metodi di HCI per identificare i requisiti degli utenti, sviluppare prototipi che implementano soluzioni di progetto, valutare gli stessi prototipi e, a seconda della valutazione, ripetere il ciclo o passare alla fase di sviluppo.

Possiamo dividere il processo iterativo in 3 macroaree: discovery, design e reality check.

Discovery	Design	Reality Check
1. Initial mapping	1. Exploring webs suggested by users	1. Low-fi user testing
2. User group analysis	2. Defining users' tasks for testing	2. Prototype review
3. Interviewing users	3. Brainstorming designs	3. Realistic clickable prototype
4. Overall & UX goals	4. Making & selecting good designs	4. 5 users test
5. Mapping and selecting a target	5. Happy paths	5. Analysing testing results
6. Defining a design brief	6. Low-fi prototypes	6. Decide next steps & conclude



## 4.1 Discovery

La fase di discovery è la prima fase del processo di sviluppo e si concentra sulla comprensione delle esigenze e dei bisogni degli utenti. Durante questa fase, vengono condotte attività di ricerca per identificare i gruppi di utenti che potrebbero utilizzare il prodotto, i loro bisogni e le loro aspettative. Le attività di ricerca possono includere interviste, focus group, osservazioni sul campo e analisi documentale. L'obiettivo della fase di discovery è identificare i requisiti degli utenti e definire le specifiche del prodotto. Il risultato di questa fase sarà un'initial map che riassume i bisogni e le aspettative degli utenti e fornisce una visione generale del prodotto.

### Esempio: Initial Map dall'anno scorso

#### Blog/Forum per appassionati di cinema

