

- Design
- Human Centered Design (HCD)
- Progettare l'interazione
- Principi Fondamentali dell'interazione
- Constraints
- How people do things
- Human Error and Mitigation
- Personas, Requirements, User Stories, Scenarios, Use Cases
- Information Architecture
- Front-end Design and Wireframing
- Innovation Methods
- UX per IoT
- Pretotyping
- Prototyping and Mock-ups
- Human Interface Devices
- UI
- Natural UI (NUI)
- Usability Testing

Design

- *def:* sia il processo di sviluppo di un prodotto sia il risultato stesso
- *obiettivo* capire il problema, perché esiste e risolverlo **Interaction Design**
 - *def:* attività di progettazione dell'interazione che avviene tra esseri umani e oggetti
 - *obiettivo* rendere macchine, servizi e sistemi usabili dagli utenti

Product Design

- *def:* progettazione di beni e servizi con l'obiettivo di essere usati da più utenti possibile
- il *designer* è quello che inventa un nuovo modo/oggetto

User Experience (UX)

- come si sente l'utente?
- logica e struttura dietro agli elementi con cui si interagisce
- aumentare soddisfazione e fedeltà del cliente

User Interface (UI)

- interfaccia utente che renda possibile l'esperienza progettata
- si usano i *wireframe*

Human Centered Design (HCD)

- **Sviluppo Antropocentrico**
 - *obiettivo* sistemi interattivi incentrati su facilità e utilità
 - utente al centro del processo di sviluppo
 - buon design = buona comunicazione
 - soprattutto se va male
- *4 fasi*
 1. specificare contesto d'uso
 2. specificare requirements
 3. progettare soluzioni
 4. testare e valutare

Progettare l'interazione

- **Discoverability**
 - *cosa?*
 - *def:* capacità di un sistema di comunicare i propri possibili usi

Understanding

- *come?*
- *def:* capacità di un prodotto di farsi usare correttamente

Principi Fondamentali dell'interazione

- **Principi fondamentali della psicologia**
 - La discoverability è il risultato di 6 principi: **AFFORDANCES**
 - *def:* interazione tra le proprietà dell'oggetto e l'utente
 - interazione no proprietà
 - **Anti-Affordances:** prevenzione di un'azione
 - **SIGNIFIERS**
 - *def:* indicano dove compiere l'azione permessa dell'affordance
 - **Intenzionali:** cartello con "pull"
 - **Non-Intenzionali:** sentiero evidente
 - **CONSTRAINTS MAPPING**
 - *def:* relazione tra gli elementi di 2 insiemi
 - Mapping *naturale* porta più velocemente a capire il concetto, definito da cultura e standards
 - **FEEDBACK**
 - *def:* comunicazione del risultato di un'azione
 - deve essere *immediato*
 - devono essere informativi, non invadenti, del giusto numero
 - **MODELLO CONCETTUALE**
 - ciò che il designer pensa
 - *def:* spiegazione molto semplice di come funziona il sistema
 - **Modello Mentale:** idea che si fa l'utente
 - Idealmente modello mentale = modello concettuale

Constraints

- *def:* indicano quali azioni sono limitate
- 4 tipi:
 1. Fisici
 2. Culturali
 3. Semantici (significato azione)
 4. Logici
- Un buon mapping può essere un vincolo **Forcing Function**
 - 3 tipi:
 1. **Interlocks:** forza a seguire dei passi
 2. **Locks-ins:** previene l'arresto di un'azione
 3. **Locks-out:** previene l'esecuzione di un'azione

How people do things

- **Golfi**
 - **Golfo dell'esecuzione:** dove si capisce come fare le cose
 - Pianificare, Specificare, Performare
 - (signifiers, constraints, mapping, modello concettuale)
 - **Golfo della valutazione:** dove si valuta il risultato delle azioni
 - Percepire, Interpretare, Comparare
 - (feedbacks, modello concettuale)

I 7 STATI DELL'AZIONE

1. Goal
2. Pianificare
3. Specificare
4. Performare
5. Percepire
6. Interpretare
7. Comparare

Feedforwards

- aiuta nel rispondere alle domande su cosa fare

Azioni opportunistiche (istintive)

- scelte rispetto alla situazione, non pianificate/analizzate
- meno precise ma meno stressanti

Human Thought

- 3 livelli di processo:
 1. **Viscerale**
 - percezioni immediate
 - scelte estetiche
 2. **Comportamentale**
 - aspettative dell'utente
 - conoscenza e comprensione del prodotto
 3. **Riflessive**
 - riflessioni conscie

Human Error and Mitigation

- Le **interruzioni** sono una ragione comune di errore **Root Cause Analysis**
 - Bisogna investigare sull'origine del problema, lo facciamo con i **5 perché**

Tipi di errori

- **Lapsus**
 - fare una cosa anziché un'altra
 - **action-based**: si esegue un'azione sbagliata
 - **memory-based**: l'azione viene mancata
 - comuni nei professionisti
- **Errori Cognitivi**
 - goal o piano sbagliato
 - **rule-based**: errato flusso di azioni
 - **knowledge-based**: non si capisce il problema
 - **memory-lapse**: dimenticanze negli stadi del goal
 - comuni nei principianti

Prevenzione dell'errore

- **Capire** le cause dell'errore
- fare **test di sensibilità**
- fare in modo che si possa tornare indietro (**undo**)
- rendere **facile** la **scoperta** dell'errore
- non trattare l'azione come errore, ma **aiutare l'utente** a finirla
- si possono aggiungere **constraints**

Personas, Requirements, User Stories, Scenarios, Use Cases

- **Personas**
 - personaggio su un potenziale user del nostro prodotto
 - per sviluppare il design sulle necessità degli utenti
 - 3 tipi:
 - **Proto-personas**

- utenti standard senza dettagli
- studia casi più comuni
- **Qualitative personas**
 - si intervistano un numero di utenti
 - informazioni più dettagliate sui loro scopi
- **Statistical personas**
 - sondaggi estesi a un grande pubblico
 - raggruppare risposte simili in clusters
- **REGOLA 80/20**
 - concentrati sul 20% dei tuoi utenti che saranno l'80% delle tue entrate
- **ARCHETIPI**
 - tipi di personas più astratti
 - descrizione astratta del tipo di user
 - archetipi \subset personas

Requirements

- *servizi, funzioni, features*
- l'ambiente di business cambia nel tempo
- 2 tipi:
 - **Functional requirements**
 - funzioni o features
 - cosa è necessario ottenere, senza la soluzione
 - **Non-functional requirements**
 - quanto bene la soluzione deve attenersi
 - descrive sicurezza, affidabilità, mantenimento, reperibilità, ...

User Stories

- Riassunto che identifica l'utente e i suoi bisogni
- uno per persona
- documentano comportamenti e necessità dei vari users
- vari livelli di dettaglio, quelle più grandi e dettagliate sono dette **epic**
 - le **epic** sono divise in stories più piccole

Scenarios

- Situazione che raccoglie le info di come il nostro utente esegue una task
- possiamo determinare:
 - punti più importanti su cui concentrarsi
 - fasi del processo che richiedono più aiuto
 - bisogni e motivazioni degli utenti

Use Cases

- descrivono l'interazione che ha l'utente con il prodotto per raggiungere il goal
- User POV

Information Architecture

- **Strutture di interfaccia**
 - 4 modi per strutturare la UI
 1. **Gerarchica**: strutture ad albero
 2. **Sequenziale**: percorso step-by-step
 3. **Matrice**: user sceglie il percorso (tutto linkato)
 4. **Database**: es. Google

Architettura dell'Informazione (IA)

- organizzazione, struttura, etichettatura del contenuto in modo efficiente e sostenibile
- aiuta gli utenti a capire
- informa la strategia dei contenuti
- **Contesto**: obiettivi aziendali, finanziamenti, politica, ...

- *Contenuto*: obiettivi dei contenuti, tipi di dati, volume, ...
- *Utenti*: pubblico, compiti, bisogni, comportamenti di ricerca, ...
- Gli schemi organizzativi servono a decidere come organizzare le info
 - *Esatto*: schemi rigidi ed esatti (alfabetico, cronologico, ...)
 - *Soggettivo*: basato su scelte aziendali (task, audience,...) o metaforiche (icone tipo cestino)

NAVIGAZIONE

- orientamento dell'utente nell'interfaccia
- *findability*: trovare quello che si sta cercando
- *discoverability*: scoprire nuove features utili

User behavior patterns

1. *Quit*: fa una ricerca, vede e esce
2. *Narrow*: fa una ricerca, vede e filtra i contenuti
3. *Expand*: fa una ricerca, vede e espande i contenuti
4. *Pearl growing*: fa una ricerca, apre un risultato e apre un link al suo interno (wiki)
5. *Pogostick*: ripete fa una ricerca, vede e l'azione tornando alla pagina di ricerca
6. *Thrashing*: fa una ricerca, vede e fa una nuova ricerca con più dettagli
7. *Berry pick*: fa una ricerca, vede tutto e fa una query con più dettagli

Design patterns

- autocomplete
- autosuggest
- instant result
- did you mean
- autocorrect
- best first
- partial matches
- related searches
- federated search (più database contemporaneamente)
- advanced search
- scoped search
- personalization
- pagination (max risultati per pagina)
- structured result (visualizza risultato in base al tipo di dato)
- actionable result
- comparing result
- unified discovery (più tipi uniti in un'interfaccia)

Document Object Model (DOM)

- interfaccia che consente ad un linguaggio di programmazione di *manipolare struttura, contenuto e stile* di un sito web
- es JS

Front-end Design and Wireframing

- **Wireframing**
 - sono blueprint che aiutano la *comunicazione* tra designer e programmatore
 - creazione di sketch del prodotto
 - *Cosa fa*
 - prima forma ad una IA
 - Incoraggia discussioni nel team
 - determina funzioni della UI
 - *Cosa non fa*
 - non dà la sensazione di completezza
 - non provvede funzionalità
 - non garantisce la comprensione

- **Low-fidelity Wireframe**
 - si focalizzano su struttura e layout
- **High-fidelity Wireframe**
 - rappresentazione dettagliata

Adattabilità

- adattare il prodotto a necessità, contesto, abilità e preferenze dell'utente **RESPONSIVE DESIGN**
 - adattare dimensione e risoluzione del prodotto su diversi device
 - si può lavorare su una sola versione del sito
 - varie tecniche:
 - *griglia flessibile*: adattarsi in grandezza in base alla grandezza del device
 - *immagini flessibili*
 - *media queries & breakpoints*: stili applicati in base alle caratteristiche del device
- **Mobile First**
 - filosofia che inserisce tutte le features principali e le ottimizza su mobile
 - dopo (forse) le mette su desktop
- Good practices:
 - keep it simple
 - prioritize the content
 - design for touch
 - optimize for load time
 - test, test, test

Accessibilità

- Problematiche visive, uditive, motorie, disturbi (ex epilessia), cognitive/apprendimento
- L'adattabilità deve essere sviluppata insieme al processo di design

User flow

- determina le possibili azioni/step di un utente generico

Innovation Methods

- **Innovazione**
 - *def*: qualcosa di originale e più efficiente che "rompe" nel mercato
 - la maggior parte sono gradualisti su un prodotto già esistente **DISRUPTIVE INNOVATION**
 - crea un nuovo mercato e raggiunge un nuovo pubblico

Human Centered Design

- Osservare il comportamento dell'utente
- Mettersi nella situazione dell'utente **REGOLA DEL DOPPIO DIAMANTE**
 - Si raccolgono tutte le idee (*divergenza*)
 - Si scelgono le migliori (*convergenza*)
 - Si propongono alternative (*divergenza*)
 - Si sceglie come procedere (*convergenza*)

Design Thinking

- Processo iterativo per trovare strategie alternative non subito palesi
- riformula il problema in modo *antropo-centrico* con approccio pratico
- sperimentazione continua: schizzare, prototipare e testare

Agile

- capacità di creare e rispondere al cambiamento
- come capire ciò che succede, identificare incertezze, capire come adattarsi
- attenzione a chi lavora **SCRUM**
 - Framework agile con enfasi sullo sviluppo software
 - **Sprint**: obiettivi da completare entro un certo tempo (2 sett, max 1 mese)
 - **daily scrums**: incontri giornalieri di 15 min per monitorare i progressi
 - accetta che il problema possa non essere capito/definito pienamente all'inizio
 - 3 figure:
 - product owner

- development team
- scrum master

UX per IoT

- **Internet of Things (IoT)**

- è un sistema di dispositivi informatici interconnessi, dotati di identificatori univoci (*UID*)
- trasferire dati su una rete senza l'intervento dell'uomo **INDUSTRY 4.0**
 - organizzazione dei processi produttivi
 - device che comunicano autonomamente tra loro
 - i device creano una copia virtuale del mondo fisico
 - oggetti fisici integrati senza soluzione di continuità nella rete

Design of IoT

- Hardware e software ottimizzati per funzioni specifiche
- interazioni dell'utente progettate senza fare affidamento sugli standard di app e siti
- possibilità di interfaccia remota
- possibilità di funzionalità distribuite su device diversi
- importante coerenza nell'esperienza complessiva (*interusabilità*)

Pretotyping

- **Law of Market Failure**

- si assume che un prodotto è fallito fino a prova contraria
 - L'80/90% dei prodotti fallisce a prescindere dal livello di sviluppo

Thoughtland

- *Lost in Translation Problem*: cerchi di comunicare la tua idea di prodotto ad altri che non capiscono
- *Prediction Problem*: utenti scarsi a prevedere se useranno o se gli piacerà il prodotto

Pretotipi

- aiutano a "fallire" più velocemente
- via di mezzo tra l'idea del prodotto e il suo prototipo
- *mock-up* del prodotto creato in minuti/ore/giorni
- 7 pilastri del pretotyping
 1. obey the law of market failure
 2. make sure you're building the right it
 3. don't get lost in thoughtland
 4. trust only in your own data (YODA)
 5. prototype it
 6. say it with numbers
 7. think global, test local

TIPI DI PRETOTIPO

- **Fake door**: per pubblicizzare un servizio non ancora pronto, nuova entry del mercato
- **Mechanical turk**: risorse umane per simulare una tecnologia
- **impersonator**: usare un prodotto esistente per fingere la nuova offerta
- **Pinocchio**: un falso artefatto agisce da sostituto del vero prodotto
- **One Night Stand**: servizio completo senza l'infrastruttura per una soluzione definitiva
- **Facade**: affittare equipaggiamento costoso per simulare un'infrastruttura più stabile

Minimum Viable Product (MVP)

- Versione di prodotto con il minimo insieme di funzionalità per soddisfare le esigenze di base degli utenti
- testare e validare le ipotesi di business

Prototyping and Mock-ups

- **Mock-ups**

- rappresentazione high-fidelity del prodotto

- più dettagli rispetto a wireframe
- design visuale realistico
- mostrare idee a stack-holder e clienti

Prototypes

- campione sperimentale di un design per visualizzarlo e interagirci
- acquisisce feedbacks dagli utenti
- valutare usabilità, efficacia, esperienza dell'utente **PROTOTYPE LOW-FIDELITY**
 - carta e penna - poco realistica
 - cennette wireframes - risultati iniziali difficili da applicare
 - interazioni semplici/nulle - troppo semplice
 - fatto velocemente - over-semplifica i concetti
 - facile da modificare - poco interattivo
 - incoraggia Design Thinking - gli utenti non sanno come si usa

PROTOTYPE HIGH-FIDELITY

- può avere codice o simile - lungo e costoso da fare
- feel del prodotto - focalizza tanto dettagli poco contenuti
- test di più elementi - riluttanza verso i cambiamenti dai designers
- coinvolgente per stackholders - gli utenti pensano sia il prodotto
- stackholders valutano se è interessante
- test più accurati, risultati più concreti

PROTOTYPE MID-FIDELITY

- Mantengono vantaggi da entrambi (costo, interazioni, codice, ...)

SCOPE FIDELITY

- Livello al quale all'utente è consentito testare e vedere il prototipo
- **Horizontal**: intera prospettiva senza approfondire features
- **Vertical**: dettagli su una feature senza visione d'insieme

PROTOTYPING TECHNIQUE

- *Paper prototype*: low-fidelity su carta, mostra layout semplice e funzionalità
- *Wireframe Prototype*: simula usando input umano
- *Functional Prototype* features parzialmente o interamente funzionanti
- *Non-functional Prototype*: sembra il prodotto ma senza features

Human Interface Devices

- *Human Interface Design* = Interazione completa con gli umani **Software Specification**
 - 2 entità:
 - *host*: comunica con il device e riceve l'input, ritorna l'output al device
 - *device*: interagisce direttamente con l'utente (tastiera, mouse, ...)
 - I device definiscono pacchetti di dati
 - Presentano un **HID Descriptor**
 - array di bytes che descrive il pacchetto di dati al device
 - L'host deve recuperare l'HID descriptor e analizzarlo per comunicare

HID Devices

- 2 categorie:
 - **OUTPUT**
 - **INPUT**
 - tastiera
 - *Pointing Devices*
 - diretto (touch)
 - indiretto (mouse)
 - assoluto (penna dig, dito)
 - relativo (mouse, joystick)
 - isotonico (mouse)
 - isometrico (trackpad)

- elastico (joystick)
- position control (mouse)
- Rate control (joystick)
- barcode reader
- RFID (onde radio)
- NFC

Fitt's Law

- tempo per spostarsi fast = $\frac{\text{distanza dal centro del target}(L)}{\text{larghezza target}(W)}$
- a : tempo avvio/arresto(s), b : velocità intrinseca dispositivo, D : distanza partenza-arrivo, W : larghezza target su asse movimento

$$MT = a + b \cdot ID = a + b \cdot \log_2\left(\frac{2D}{W}\right)$$

- **Eye Tracking**: misura lo spostamento della pupilla tramite le tecniche bright-pupil e dark-pupil
- **Gaze Tracking**: misura tutti i movimenti della testa compresi occhi

HAPTIC DEVICE

- toccare, percepire e manipolare gli oggetti tridimensionali
- registrano manipolazioni
- forniscono sensazioni tattili realistiche

Interfacce utente basate su immagini 3D DEVICE DI CATTURA 3D

- **scansione 3D**: analisi di un oggetto/ambiente per raccogliere dati su forma/aspetto
- 2 tipi di device:
 - **Passivi**
 - **Stereoscopic sys**: 2 camere che guardano stessa scena
 - **Photometric sys**: singola camera con diverse luci
 - **Silhouette**: sequenza di foto intorno con sfondo a contrasto
 - **Attivi**:
 - emette luce o radiazioni e rileva la riflessione o la radiazione
 - luce, ultrasuoni, raggi X

Novel UI

- **Unità di Misurazione Inerziale (IMU)**: misura forza specifica, velocità angolare e orientamento di un corpo
- **Dispositivi a 9 assi**: giroscopio (3 assi) + accelerometro (3 assi) + bussola (3 assi)

UI

- **def**: spazio di un sistema dove avviene l'interazione fra uomo-macchina
- deve massimizzare la UX
- Un prodotto è un **Human Interface Device (HID)** se implementa un **Human Machine Interface (HMI)**
- 6 tipi di interfacce = 5 sensi + equilibrio
- **Composite User Interface (CUI)**: interfacce che usano più di un senso
- **Graphical User Interface (GUI)**: interfacce grafiche e tattili
- **Multimedia User Interface (MUI)** = GUI + audio
- 3 macrotipi:
 - **Standard**: interfaccia da sito web
 - **Virtual**: Realtà Virtuale (VR)
 - **Augmented**: realtà aumentata

Natural UI (NUI)

- interfaccia invisibile per permettere all'utente di imparare
- gli utenti passano rapidamente da principianti ad esperti
- apprendimento agevolato dal design **Intuitiva**
 - senza necessità di apprendimento
 - rende l'interazione immediata e comprensibile da subito

- convenzioni familiari e comprensibili per l'utente

Usability Testing

- testing con users per trovare problemi e migliorare, come si comporta l'user **Usability**
 - quanto è facile da utilizzare
 - 5 componenti
 1. *Learnability*
 2. *Efficiency*
 3. *Memorability*: quanto è facile riprendere la competenza
 4. *Errors*: quanti errori si commettono
 5. *Satisfaction*
 - Utility + Usability = Useful

Heuristic Evaluation

- valutare le interfacce basandosi su principi di usabilità stabiliti (euristiche) **NIELSEN'S 10 USABILITY HEURISTIC**
 1. Visibilità dello stato di sistema
 2. match fra sistema e mondo
 3. controllo dell'user e libertà
 4. consistenza e standard
 5. prevenzione dell'errore
 6. recognition rather than recall
 7. flessibilità ed efficienza
 8. design estetico e minimal
 9. riconosce diagnostica e recupera dall'errore
 10. help e documentazione

Elementi del test di usabilità

- **FACILITATOR**
 - guida il partecipante
 - da istruzione, risponde alle e fa domande
 - garantisce qualità e validità dei risultati senza influenzare
- **TASK**
 - attività realistiche che fa il partecipante
 - formulazione importante
- **PARTECIPANTE**
 - utente realistico del prodotto o con un background simile al target
 - idealmente, si parte dalle *personas*

Tipi di test di usabilità

- **Dati qualitativi (qual)**
 - osservazioni sulle caratteristiche del design
 - comprensione esperienze/sfide degli utenti
- **Dati quantitativi (quant)**
 - una o più metriche
 - completamente
 - misurazione numerica delle prestazioni