

Lez 1 - 14/09/2021 + Lez 2 slide [1]

martedì 14 settembre 2021 16:13

Un **prodotto** è qualsiasi cosa che un utente usa.

Un prodotto che nessuno compra/usa/gli dedica attivamente tempo è un prodotto inutile.

Design

Per design si intende sia il progetto di progettazione e pianificazione che l'output stesso di questo processo

L'obiettivo principale del design di prodotto è quello di comprendere il problema nel suo insieme, piuttosto che trovare la soluzione al problema specifico.

Il primo passo è quello di capire **perché il problema esiste (e se esiste)**.

Se un problema non è percepito da un utente, allora quel problema non esiste.

Nel design dell'interazione l'obiettivo primo è quello di avere un utente soddisfatto, non un software teoricamente perfetto o super-ricco di funzionalità.

Interaction design

Interaction design, o progettazione dell'interazione, è l'attività di progettazione dell'interazione che avviene tra esseri umani e oggetti in generale.

- Design di prodotto
- Design dell'esperienza utente (UX design)
- Design dell'interfaccia (UI design)

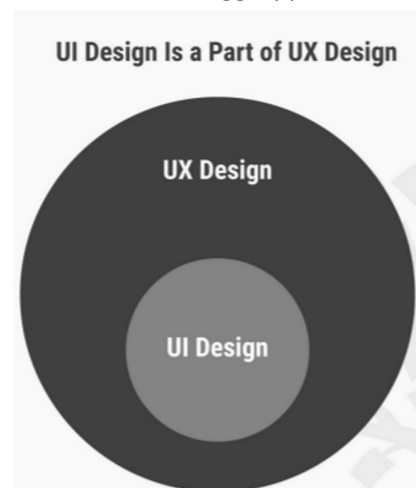
Design di prodotto

Nel design di prodotto si progettano beni e servizi il cui obiettivo principale è quello di essere utilizzati da quanti più utenti possibili migliorandone la vita

- Inventore: (designer) si focalizza su un problema da risolvere e inventa un prodotto che grazie alle sue caratteristiche tecniche permette di risolvere il problema
- User Experience Designer: (UX Designer) colui che studia come l'utente si deve "sentire" durante l'interazione con il prodotto. Ha l'obiettivo di far vivere all'utente del suo prodotto la miglior esperienza possibile. È impossibile progettare l'esperienza utente, ma **si progetta PER l'esperienza utente**, perché ogni utente si comporta in modo differente durante l'utilizzo e soprattutto si comporta in modo diverso da quello ipotizzato dal progettista
- User Interface Designer: (UI Designer)

Canvas: modo per rappresentare un prodotto/servizio a seconda di una serie di raggruppamenti standard.

Figma: power point on crack



Lez 3 - 21/07/2021

martedì 21 settembre 2021 16:24

Nel design antropocentrico si inverte il paradigma di progettazione mettendo l'utente al centro del processo. (soprattutto agli utenti che sbagliano).

E questa è una bella norma ISO

ISO 9241-210:2010(E)

L'utente non ragiona per funzioni e funzionalità, ragiona per problemi risolti (task, obiettivi).

Esempio di sviluppo tecno-centrico:

Blockchain

Sistema per la raccolta dei dati distribuito con un supporto per la certificazione del dato. Usata per le transazioni di cryptovalute.

L'applicazione della blockchain ad altri settori non sta funzionando perché stiamo cercando un problema per una tecnologia non una tecnologia per risolvere un problema.

Non si deve progettare per le persone come vorremmo che fossero, ma per come realmente sono.

Lez 4 - 22/09/2021

mercoledì 22 settembre 2021 09:06



Specificare il contesto d'uso: identificare gli utenti che utilizzeranno il prodotto, per cosa e sotto quali condizioni e vincoli

Specificare i requirements: identificare i business requirement e gli obiettivi utente che devono essere raggiunti grazie all'utilizzo di questo software

Progettare la soluzione: bozze -> prototipi -> soluzione fasi iterative

Testare e valutare

Alfa e beta testing per testare l'utilizzo che ne fanno gli utenti del software.

Alfa -> test software

Beta -> user experience

(opinione del mazzai)

Le caratteristiche più importanti di un buon design sono **discoverability** e **understanding**

Discoverability

Capacità di un sistema di veicolare e comunicare i propri possibili usi all'utente.

Per avere buona discoverability si usa la **visibilità**.

Sai cosa ci puoi fare, ma non come.

Understanding

Capacità del prodotto di farsi usare correttamente dall'utente. Quanto bene un prodotto dice *come* si usano le funzioni disponibili.

Evitare la frustrazione dell'utente. [Libro l'errore di Cartesio]

Incidente di three mile island

Migliorare la discoverability del nostro sistema

Ci sono 6 principi fondamentali dell'interazione uomo-macchina

- Affordance
- Significanti
- Constraints
- Mapping
- Feedback
- Conceptual model of the system

Affordance

Relazione tra le proprietà di un oggetto e le capacità dell'utente che vi interagisce.

L'affordance è un modo che abbiamo per incrementare l'understanding e la disc di un oggetto, ma è strettamente legata all'utente selezionato per usare il prodotto

Anti-affordance

Prevenzione dell'interazione

Significanti

Lez 5 - 28/09/2021

martedì 28 settembre 2021 18:39

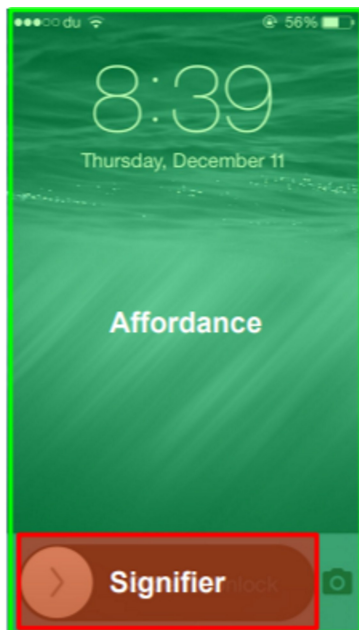
P 12 slide 4

Signifiers

Problema: come progettare le cose in modo da renderle capibili/intuitive

Lavorando con i design grafici per i dispositivi elettronici serviva un modo per capire cosa può essere toccato/slidato/quale interazione può essere eseguita con cosa

L'affordance determina quali azioni sono possibili, i significatori comunciano dove l'azione deve essere eseguita



I signifiers possono essere messi intentional (segnale push su una porta) o accidental (coda di gente al binario dice che il treno non è passato)

Nel design i significatori sono più importanti dell'affordance, comunicano come usare il design

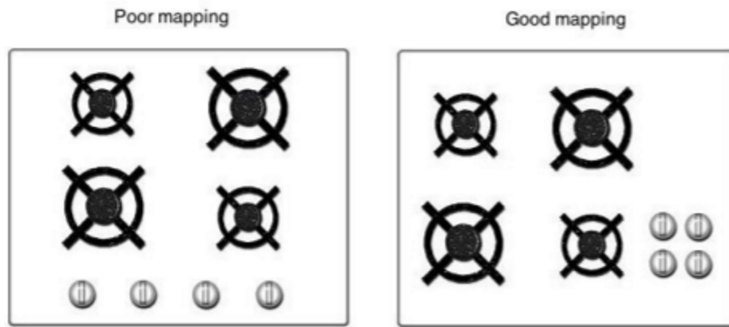
Come si va dalla percezione dell'affordance al capire l'azione che può essere eseguita?

In molti casi da convenzioni.

Ad esempio la maniglia di una porta ci comunica che può essere presa, ma il fatto che la usi per aprire la porta viene imparato. La stessa cosa su un muro può offrire supporto, ma ovvio che non ci apri il muro.

Mapping

Il mapping è la relazione tra elementi di due set di cose



..

Nel buon mapping si usa la corrispondenza del layout dei controlli con quella dei dispositivi che vengono controllati

Un mapping naturale porta a una migliore e immediata comprensione.

Alcuni mapping naturali sono culturali o biologici (muovere la mano verso l'alto per significare di più e verso il basso meno)

È appropriato usare posizioni verticali per rappresentare intensità o quantità.

Ci possono essere mapping diverse per determinate cose che feel natural, ma sono specifici a determinati ambienti

Feedback

Modo per farti sapere che il sistema sta processando la tua richiesta

Feedback: comunicare i risultati di un'azione.

Se il feedback avviene dopo i 100ms l'utente non lo percepisce

Quando c'è troppo feedback viene ignorato. Il feedback è essenziale, ma se intralcia le altre cose diviene inutile

Modello concettuale

Un modello concettuale è una spiegazione, generalmente molto semplificata di come qualcosa funziona, non deve necessariamente essere completo o preciso finché è utile.

I file, le cartelle, le icone sullo schermo del computer aiutagli gli utenti a creare un modello concettuale dei documenti e cartelle dentro il computer o delle app che devono essere aperte.

Ovviamente fisicamente non ci sono cartelle dentro il computer quindi quella è una rappresentazione efficace per rendere il sistema più facilmente utilizzabile

I modelli semplificati sono di valore finché le assunzioni che li supportano sono vere

Infatti nel Cloud Storage Sync i file appaiono come se fossero sul device, ma la maggior parte delle volte i file sono sul cloud.

Il modello concettuale è uno storage coerentemente disponibile su tutti i device di un utente.

Questo modello è vantaggioso per uso normale, ma se la connessione al servizio cloud è interrotta i risultati possono essere confusi; i file appaiono sul device, ma l'utente non può effettivamente aprirli.

Modello mentale

I modelli mentali sono i modelli concettuali in mente agli utenti che rappresentano come immaginano che le cose funzionino.

Diversi utenti possono avere due modelli mentali diversi di una stessa cosa, anche una sola persona può avere più modelli mentali della stessa cosa, ognuna per ciascun aspetto del suo uso. I modelli possono anche essere in conflitto.

I modelli concettuali sono spesso dedotti dal device stesso. Alcuni modelli sono passati da persona a persona, altri vengono da manuali.

Di solito il dispositivo stesso offre poca assistenza quindi il modello viene acquisito tramite esperienza. Spesso questi modelli sono sbagliati e risultano in difficoltà nell'uso del device.

Esempio del termostato comparato coi fornelli

Immagine di sistema/system image

Le persone creano modelli mentali di sé stesse, gli altri, l'ambiente e le cose con cui interagiscono. Questi sono modelli concettuali creati attraverso l'esperienza, allenamento e istruzione.

Come creiamo un appropriato modello concettuale per il device con cui interagiamo?

Non potendo parlare con il designer, ci affidiamo a qualsiasi informazione ci è disponibile: come appare il device, cosa sappiamo avendo usato cose simili in passato, cosa ci è stato detto del device da diverse fonti.

L'informazione combinata che ci è disponibile è **l'immagine di sistema (system image)**

Quando l'immagine di sistema è incoerente o sbagliata, l'utente non riesce a usare facilmente il device, se è incompleta contraddittoria ci potrebbero essere dei problemi.



Modello concettuale del designer -> oggetto venduto -> oggetto all'utente

L'immagine di sistema è quella che può essere percepita dalla struttura fisica che è stata costruita

Changing conventions

Le persone obiettano e si lamentano quando nuovi approcci vengono introdotti in un gruppo esistente di prodotti e sistemi

Le convenzioni vengono violate e quindi c'è bisogno di imparare di nuovo

Si preferisce la coerenza nel design

Lez 7 - 05/10/2021

martedì 5 ottobre 2021 16:19

Sempre per migliorare understanding e cose

Constraints

Come facciamo a fare cose che non abbiamo fatto prima d'ora?

conoscenza pregressa e vincoli, combinati

La conoscenza a priori è guidata dalle affordance e anche dai significanti a volte, ma questa conoscenza viene acquisita attraverso l'assimilazione di vari modelli concettuali che si vanno a acquisire nelle esperienze di vita

La conoscenza a priori è legata agli aspetti culturali

I vincoli più importanti sono quelli di tipo **fisico/geometrico** (lego, puzzle, tappo di bottiglia)

I vincoli di tipo **culturale** e **semantico** sono più deboli.

Una via di mezzo sono i vincoli **logici**. (limitati dalla conoscenza dell'utente)

- Fisici
- Culturali
- Semantici
- Logici

Servono per limitare lo "spazio di manovra" dell'utente, dargli una guida

Mancanza di vincoli e mapping:

Senza mapping e senza vincoli il modello concettuale non viene trasmesso

Per poter fare un design che sia utile e apprezzato dall'utente la prima cosa è capire quali sono i task + importanti che il nostro utente va ad effettuare.

Vincoli per obbligare un comportamento

FORCING FUNCTIONS - FUNZIONI OBBLIGANTI

Basati sui vincoli, hanno la finalità di obbligare un comportamento dell'utente

Vincoli forti che servono a evitare comportamenti indesiderati dagli utenti.

Interlock

Obbliga a fare una serie di azioni prima di raggiungere un obiettivo

Vincolo che forza le azioni in un certo ordine.

(frullatore che non si avvia senza coperchio) verifica dell'email

Non puoi arrivare al tuo obiettivo se non fai quello che è stato "chiesto"

Lock-ins

Salva prima di uscire

Blocca l'utente nello stato corrente per fargli prendere una decisione prima di passare allo stato successivo.

Lock out

Impedisce un'azione fino a che l'utente non prende una decisione

Preclude l'accesso a uno stato.

Activity-centered controls

Apoteosi della filosofia human-centered

Controlli incentrati sulle attività dell'utente.

Negli active centered controls ciò che è importante non è il controllo, ma il processo?

L'attività che l'utente ambisce a fare

Modo di interagire con il dispositivo basato sull'attività che l'utente vuole fare

Non interazioni atte a abilitare funzioni del sistema

Problema nella progettazione activity centered.

Mappare le attività sugli utenti è complesso. Infatti si trova in oggetti parecchio moderni (automazione casa ecc.)

Es routine google home richiede uno skill set dall'utente abbastanza nerd

Lez 8 - 11/10/2021

lunedì 11 ottobre 2021 10:09

How do people do things

Psicologia e piccolo modello concettuale di come le persone selezionano e valutano le loro azioni.

Questo porta alla discussione del ruolo dell' understanding e delle **emozioni: piacere** quando le cose funzionano senza problemi, **frustrazione** quando le cose non vanno come pianificato.

Quando le persone usano qualcosa, si trovano davanti a due "golfi"

- Gulf of Execution, dove cercano di capire come funziona
- Gulf of Evaluation, dove cercano di capire cosa è successo

Il ruolo del designer è aiutare le persone a attraversare tra i due golfi

Gulf of evaluation

Il gulf of evaluation è solitamente percepito come facile da attraversare

Riflette quanto sforzo deve fare un utente per interpretare lo stato fisico del device e determinare quanto bene le aspettative e intenzioni sono state soddisfatte

Il gulf è piccolo quando il device fornisce informazioni sul suo stato in un modo facile da interpretare e che corrisponde al modo in cui la persona pensa al sistema

02/11/2021

martedì 2 novembre 2021 16:20

Dispositivi di input

Tastiera:

Il modo in cui sono fatte le tastiere è normato ISO

3 layout da prendere in considerazione:

- Layout fisico: come sono posizionati i tasti sulla tastiera (normato)
- Visual layout: come le etichette sono poste sui tasti (es QWERTY)
- Functional layout: funzione che l'applicativo, SO, ecc, associa al tasto premuto o alla combinazione di tasti premuti (es spazio per pausa yt)

Il sistema operativo non sa che tastiera hai (layout visuale) sa solo la parte fisica

Tastiere multifunzione

Funzioni aggiuntive alla tastiera standard

Si ha bisogno in genere di un driver specifico per l'utilizzo delle funzionalità

Lettore di barcode

Un lettore di barcode è, dal punto di vista del SO, un serializzatore di caratteri, legge il barcode e invia la stringa.

I QR code sono l'"evoluzione" dei barcode oltre a mappare cifre mappano anche caratteri

RFID - identificatori a radiofrequenza

Gli RFID sono dei sistemi di input di stringhe di testo che funzionano attraverso un chip e un'antenna, negli RFID passivi, la potenza per accendere il chip, fare quella poca computazione che serve e emetterlo viene raccolta direttamente dall'impulso radio che l'antenna riceve del chip lo interroga.

Gli RFID attivi sono alimentati. Possono anche emettere impulsi, si usano anche per geolocalizzazione

Un barcode deve essere visibile tra il codice e il lettore (line of sight) un RFID no

NFC

Non è solo un dispositivo di input, è un dispositivo di comunicazione, sempre comunicazione radio a bassa distanza

Sistemi di puntamento

Un sistema di puntamento è un input device che consente di sottoporre input spaziali a un computer.

È un dispositivo che consente di replicare movimenti fisici dell'utente in movimenti di un elemento della GUI, tipicamente un cursore.

Ovviamente il più famoso è il mouse

Legge di Fitts

Quanto è il tempo richiesto dall'utente per muoversi da un punto di partenza al target

This scientific law predicts that the time required to rapidly move to a target area is a function of the ratio between the distance (D) to the target and the width of the target (W).

Movement Time:

- a = time to start/stop in seconds (empirically measured per device)
- b = inherent speed of the device (empirically measured per device)
- D is the distance from the starting point to the center of the target.
- W is the width of the target measured along the axis of motion.

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

Come noi ci interfacciamo con i dispositivi di input (sempre pointing d)

Possiamo avere input di tipo

- Diretto: un dispositivo che ci consente di agire laddove effettivamente viene perturbato il mondo (testuali parole) ovvero schermo touch
- Indiretto: interagisco su uno spazio che fisicamente è dislocato rispetto alla superficie che vado a modificare

Movimento:

- Assoluto: mapping consistente tra device e cosa succede nella gui
- Relativo
- Isotonico: si muove e misura il suo spostamento (mouse)
- Isometrico: un dispositivo isometrico non si muove e misura la forza che gli viene imposta (trackpoint)
- Elastico: la forza che applicate è proporzionale allo spostamento che fate fare al dispositivo (joystick)

Modo in cui i dispositivi fanno avanzare il puntatore

- Position control: la posizione del device influenza in maniera diretta la posizione assoluta o relativa del cursore sullo schermo
- Rate control: non controllo la posizione ma la velocità e la direzione di spostamento del puntatore (joystick)

16/11/2021

martedì 16 novembre 2021 16:22

NUI

L'immagine di sistema che io sviluppo per una NUI fa molto affidamento sulle skill pregresse e fa modo di introdurre sull'interfaccia stessa le veicolazioni all'azione

Ci sono tutta una serie di gesti touch che ormai abbiamo imparato, sono diventati elementi fondamentali del nostro background culturale:

- Tocco
- Scroll
- Pinch
- Swipe
- Rotate

Come sono stati scelti questi gesti:

Al fine di essere il più simile possibile a quello che fareste nell'azione fisica.

Come vengono resi intuitivi?

Feedback molto veloci, interazione naturale con qualcosa che si adatta come si adatterebbe qualcosa di fisico. Se questo feedback non è così veloce, l'utente non riesce ad associare il comportamento dell'interfaccia alla sua azione (disaccoppiamento)

Il fatto che si possa fare una determinata interfaccia vocale/touch.. Diretta, non fa di questa interfaccia un'interfaccia Naturale.

Naturale significa che l'utente capisce in maniera intuitiva come la deve usare

Tenere sempre in considerazione il contesto di utilizzo

Es se devo fare un'interfaccia che mappa un disco di scratch non deve saperla usare anche mia mamma, ma qualcuno che sa fare scratch

Two- and 3-year-old children could not follow any prompting technique and only a minority (27%) could tap the touchscreen at an intended place. Four- to 6-year-old children could perform simple gestures like a tap and slide (57%) and follow instructions provided through animation (63%). Seven- and 8-year-old children could perform more sophisticated gestures like dragging and dropping (30%) and follow instructions provided in audio and video formats (34%). [source here](#)

Perché una NUI sia definibile tale noi vogliamo che l'utente possa apprendere istantaneamente, questo apprendimento non deve essere completo, ma progressivo l'interazione è sempre diretta e si pretende un *basso carico cognitivo*. (ma continuo)

Progressive learning molto naturale

Una NUI ben progettata trae vantaggio dalle skill esistenti dell'utente (abilità innate)

È progettata considerando il percorso di apprendimento che l'utente deve fare

Esiste un passo successivo alle NUI

Introduzione della realtà virtuale

Nella realtà aumentata andiamo a ottenere una Reality Based Interface, rendo cliccabili degli oggetti reali. Non necessariamente deve essere in virtual reality della realtà intorno a te

Fare una RBI non vuol dire necessariamente fare una NUI

GUI design

- Interaction design
- Visual design
- Information architecture (architettura dell'informazione)

Struttura delle interfacce

La struttura dell'interfaccia non è il visual design

Mi dice come l'interfaccia a livello di navigabilità è strutturata

Come il contenuto è distribuito e organizzato

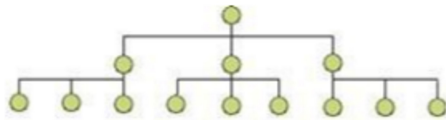
Come l'user experience è organizzata nel tempo e ricerca delle informazione

4 macro modi:

- Gerarchico
- Sequenziale
- A matrice
- A database

Gerarchica

Ad albero, l'utente comincia la navigazione attraverso una pagina generale e poi viene portato avanti su pagine di dettaglio, viste che vanno sempre più nel dettaglio



Struttura sequenziale

Tipicamente creazione account, configurare qualcosa..

Ha un inizio e ha una fine

Struttura a matrice

Sisti web

Io non posso vincolare a nessuna struttura, ho un tipo di navigazione molti a molti

Mis erve tipicamente una sitemap o un navigator

L'utente deve sapere dov'è

Modello a database

non è una struttura dell'interfaccia generalista per le GUI (non destinata a chiunque)

Modello excel o di qualsiasi altro tool che prevede che l'utente abbia dentro di sé presente la struttura del dato

A prescindere dal tipo di struttura che si vuole creare, la prima cosa che si deve prendere in considerazione è lasciarsi la possibilità di crescere o cancellare.

Quello che pensiamo sia necessario non è sicuramente quello che ci sarà nella versione finale del software, prediligere strutture che garantiscano la crescita: aggiungere sezioni a parti di dettaglio dell'interfaccia, ma anche creare sezioni nuove

Evitare sempre o le strutture troppo distribuite, larghe, ma anche troppo nested, profonde

Architettura dell'informazione

L'architettura dell'informazione ci dice come i contenuti sono organizzati

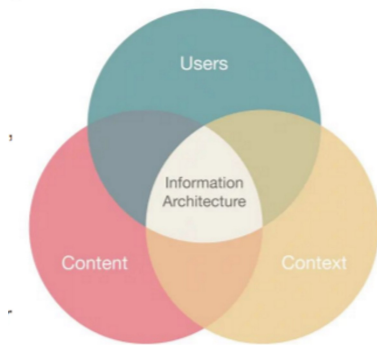
Serve a veicolare il modello concettuale.

Si compone di diversi elementi:

- Schema e struttura dell'informazione
- Sistemi di labeling: come si chiamano le cose già veicola una parte importante della struttura dell'informazione
- Sistemi di navigazione la struttura dell'interfaccia si rispecchia nell'architettura che scelgo
- Sistemi di ricerca

Come la scegliamo:

Valutiamo utenti, contesto e contenuto



Schema organizzativo

Esistono due tipologie di schemi:

Esatti e soggettivi

Esatto

Distribuisce e organizza le informazioni in modo oggettivo, universalmente accettato

Esempi:

Ordine alfabetico

Le rubriche sono organizzate con un architettura di tipo oggettivo, in particolare ordine alfabetico

Ordine cronologico:

Le foto sono rganzizzate in ordine cronologico (le foto sul rullino ad esempio sono ovviamente in ordine cronologico)

Organizzazione di tipo geografico

Schemi soggettivi

Sono legati alla percezione dell'utente, sono tipicamente molto utili per veicolare informazioni, perché si adattano all'utetne, ma più complessi da progettare e mantenere

Es:

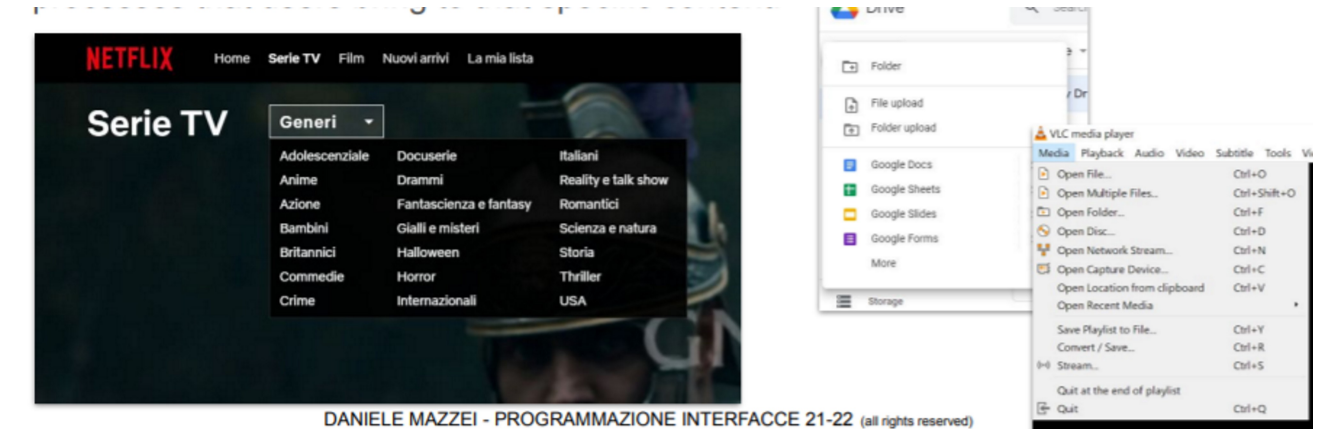
I suggerimenti di netflix

L'architettura dell'informazione serve all'utente

Metodi per schemi soggettivi:

Schemi per topic

Schema a task:



In diverse parti uso schemi diversi per veicolare il contenuto

Inventare uno schema è un po' una situazione

In casi particolari puoi mischiare gli schemi, ma l'utente è già abituato agli altri schemi quindi mappa quello che te hai fatto msito o su uno o su l'altro a seconda di quello con cui è più familiar

Document object model

Rappresentazione in alto livello della struttura della nostra app interpretata poi dal browser e whatever

Organizzato come documento gerarchico (dice non ci interessa)

Come sono organizzati e come sono chiamati le componenti che abbiamo per produrre

un'interfaccia grafica

- Input control

checkbox: consente all'utente di selezionare una o più opzioni da un set

dropdown list

list boxes (selezionare più di un item)

buttons/dropdown buttons

toggle

<https://www.usability.gov/how-to-and-tools/methods/user-interface-elements.html>