

# - Indice -

- Indice -	1
- Introduzione -	2
- In concreto -	2
- Progetti e Collezioni -	3
- Modello ER -	4
- WBS -	5
- Material Design -	6
- Layout -	7
- Dettagli che fanno la differenza -	8
- Tecnologie utilizzate -	9
HTML5, CSS, JavaScript, Materialize CSS	
jQuery, Ajax	
Apache, MySQL, MySQLi, PHP, AlterVista	
GitHub	
- Gestione dei file -	11
- Sicurezza e Privacy -	11
- SEO -	12
- Sitografia -	14

## - Introduzione -

Gli ultimi anni hanno portato a uno sviluppo sempre maggiore della stampa 3D, con stampanti sempre più economiche e alla portata di tutti. L'unico inconveniente è rappresentato dal fatto che per stampare un oggetto, è necessario progettare con appositi software che richiedono conoscenze semi-professionali, cosa che non tutti logicamente possiedono. In questo contesto si inserisce DDDParts, una piattaforma che consente ai propri utenti di condividere le proprie creazioni e renderle fruibili a tutti, in modo che chiunque, con conoscenze più o meno approfondite, possa dare forma ai propri desideri nel salotto di casa.

Inoltre le aziende possono mettere a disposizione dei clienti i modelli dei propri pezzi di ricambio, rendendoli disponibili a chiunque, in pochi istanti e a basso costo, riducendo i costi di magazzino e gli sprechi.

## - In concreto -

Di fatto un utente DDDParts, dopo essersi registrato, ha a disposizione:

- una sezione dedicata al proprio profilo, con le informazioni personali e sociali, come follower, likes, progetti e collezioni, e la relativa gestione;
- una parte dedicata alla creazione di un nuovo progetto, nella quale può caricare i file che desidera e aggiungere informazioni quali titolo, descrizione, categoria e tag;
- una dashboard, nella quale vengono visualizzati i progetti, le collezioni e gli aggiornamenti più recenti degli utenti che segue e ai quali è interessato;
- una Search Box per consentire ricerche rapide ed immediate.

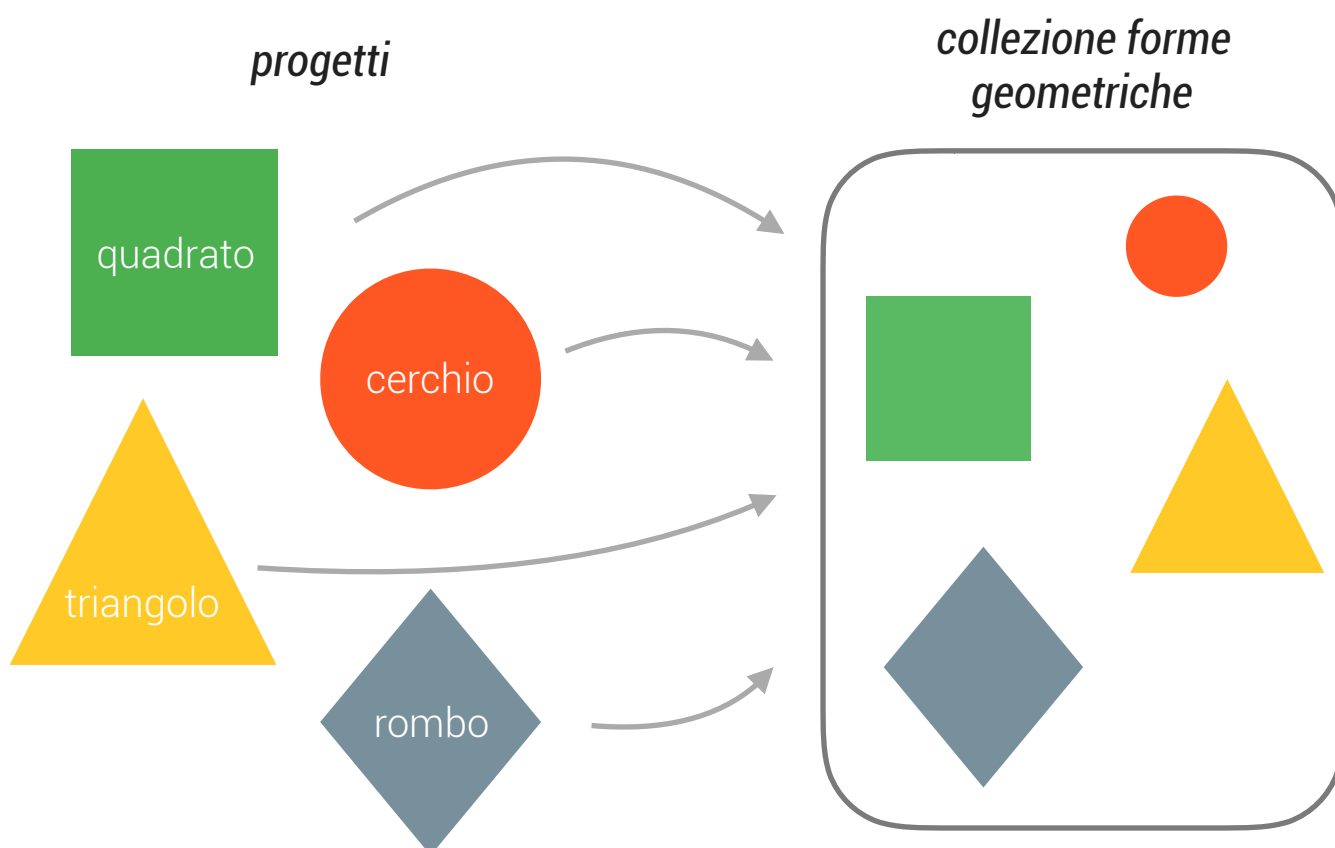
# - Progetti e Collezioni -

Progetti e Collezioni sono gli elementi alla base di DDDParts, ma cosa sono precisamente?

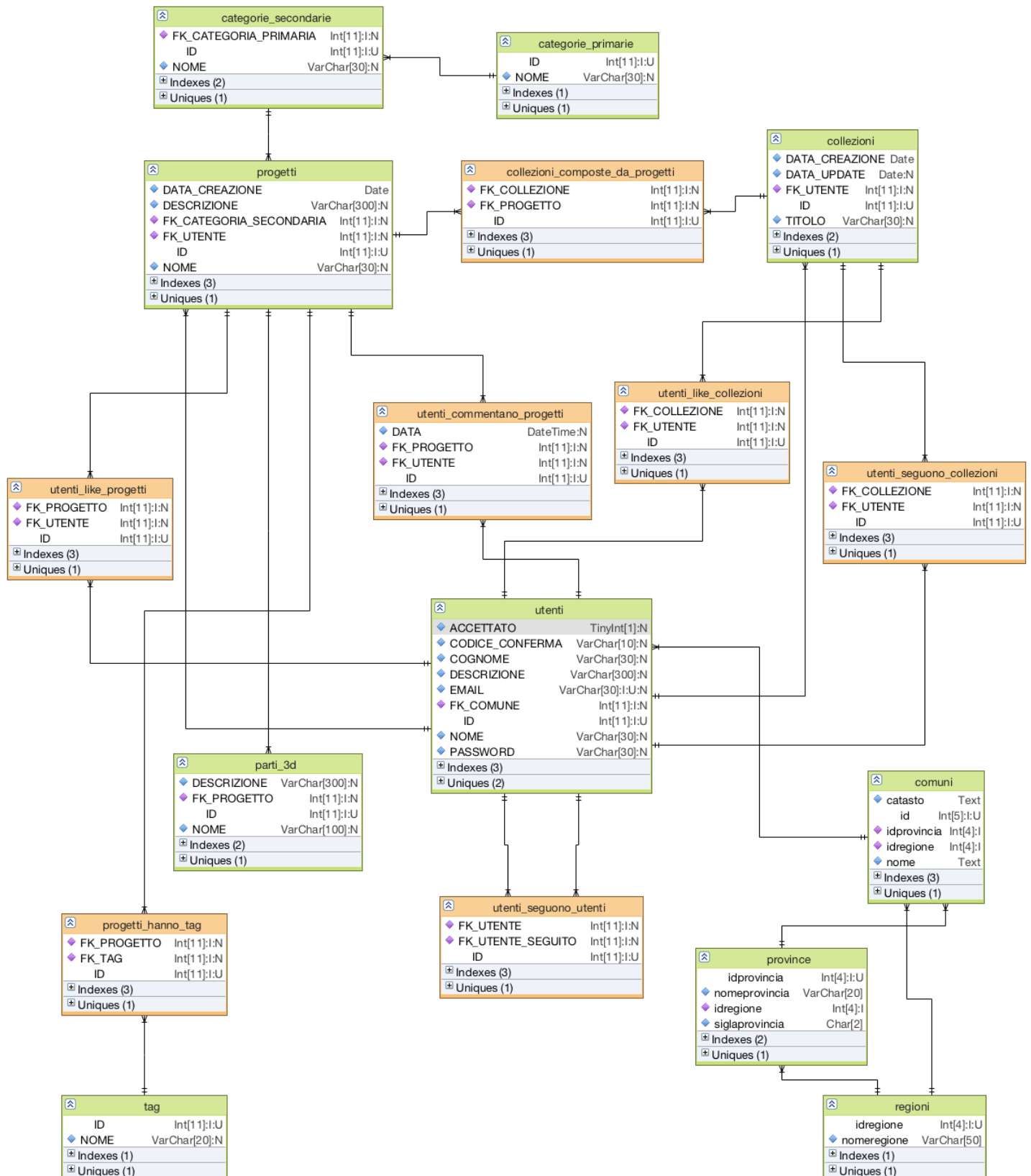
Un progetto è un insieme di file, quali immagini, testi, .STL o .OBJ, che un utente può caricare e condividere con gli altri. Questo viene poi contrassegnato dal creatore con un titolo, una descrizione, le categorie di appartenenza, i tag e l'immagine principale.

Una collezione invece è un semplice raggruppamento di progetti inerenti uno stesso argomento. L'utente può creare le proprie collezioni ed aggiungere ad esse i propri progetti o quelli di altri in modo del tutto libero e creativo. Il loro scopo è quello di avere sempre tutto ordinato e a portata di mano.

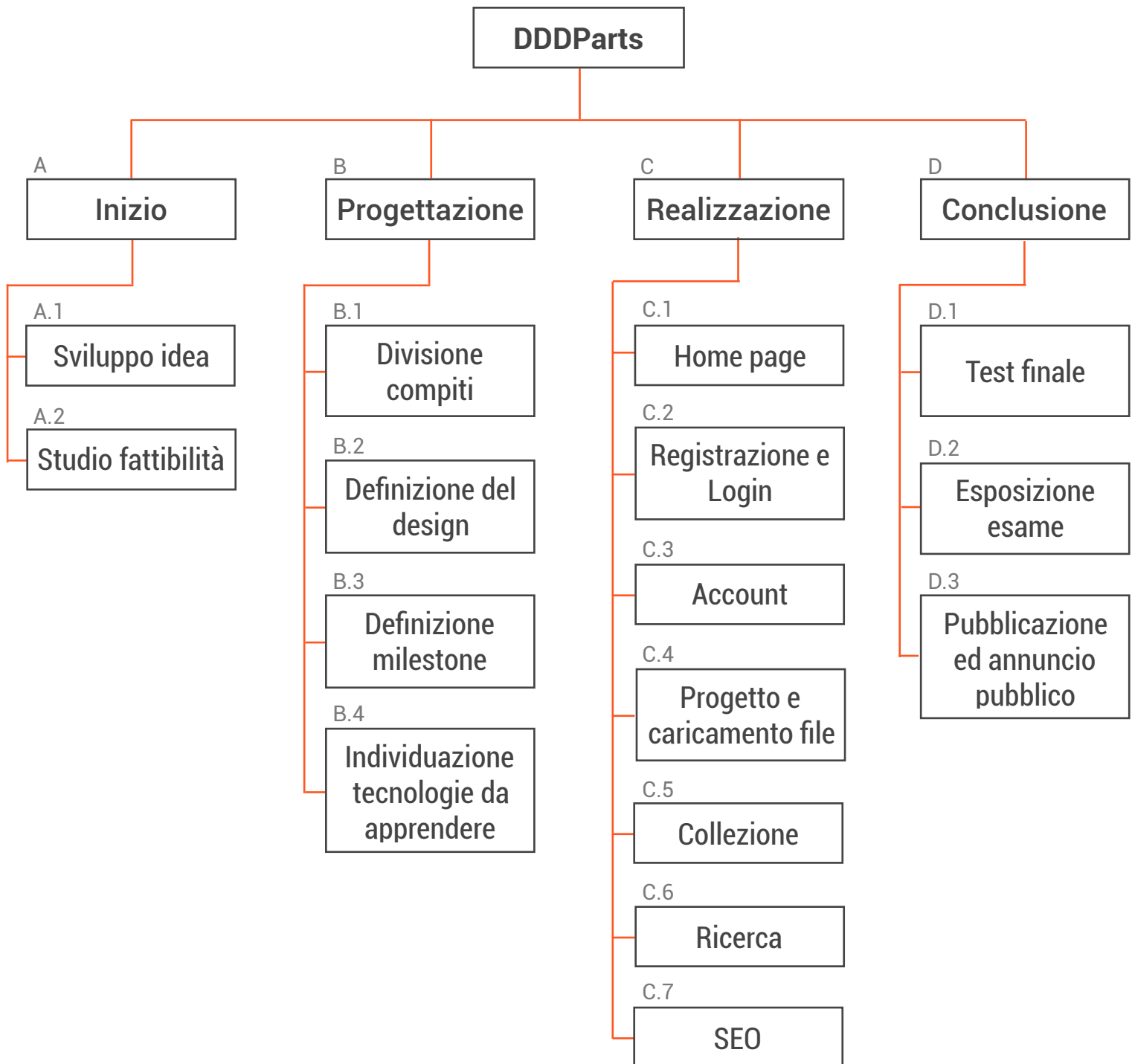
Esempio:



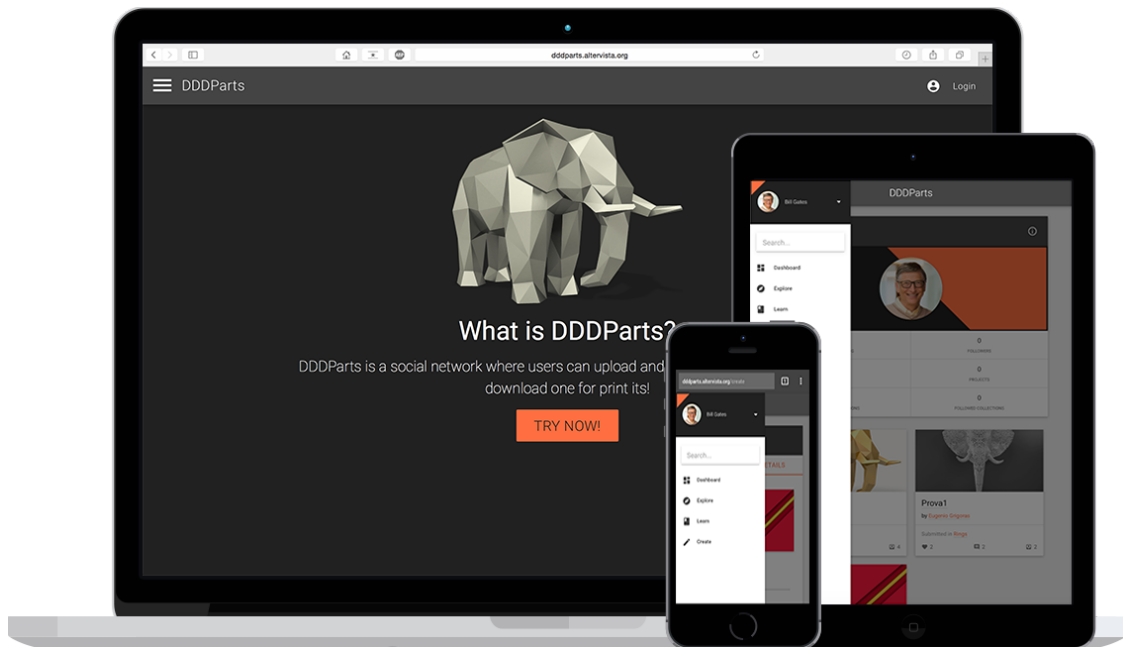
## - Modello ER -



# - WBS -



# - Material Design -



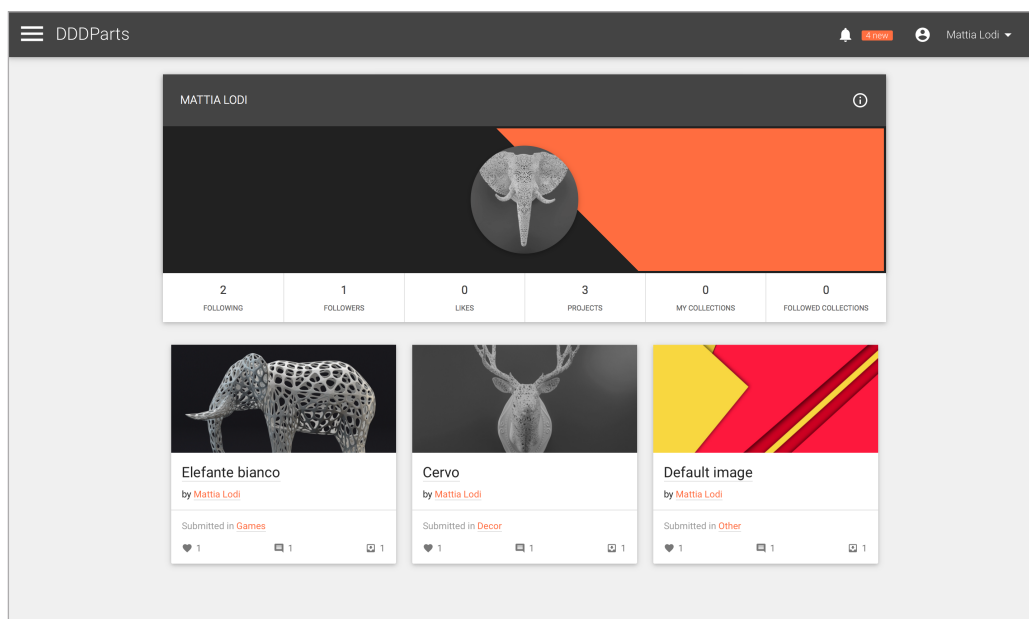
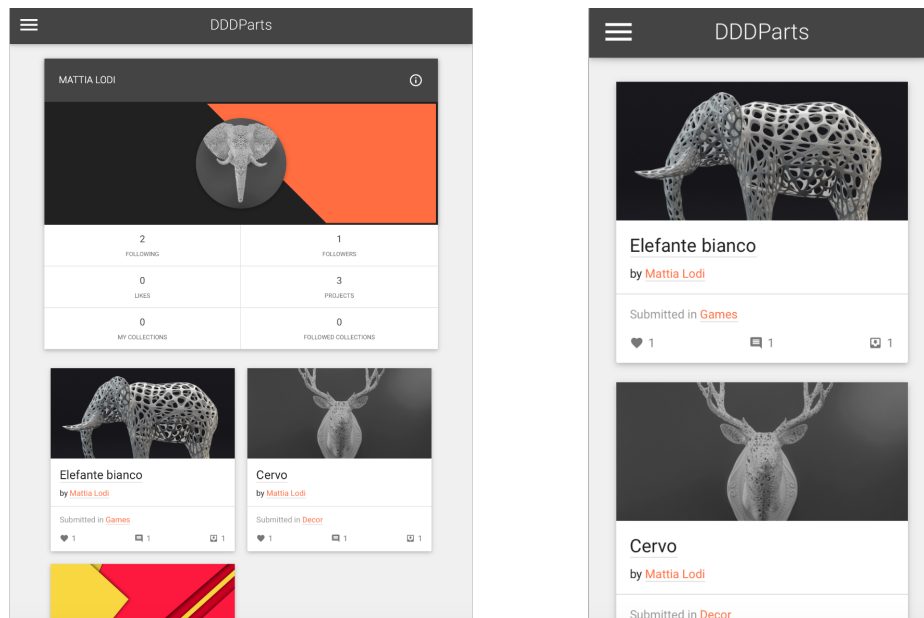
Il material design è uno stile, un codice, un linguaggio di design. È lo stile con cui Google ha deciso di rinnovare tutti i suoi prodotti e sotto il quale gestirli tutti quanti allo stesso modo, con gli stessi principi di esteriorità grafica.

Il concetto chiave del material design è la parola *material*. Dove material, cioè il materiale, è un oggetto. Qualcosa di molto più bello di un qualsiasi bottone o menu.

Abbiamo cercato di restare il più fedeli possibili a questa filosofia, seguendo le linee guida di Google e le regole necessarie. Il risultato è un sito family feeling con l'interfaccia di Android, in cui la maggior parte degli utenti si possono ritrovare, favorendo la navigazione e rendendo il tutto 'normale'. Tutto è proprio come ci si aspetta e dove ce lo si aspetta.

## - Layout -

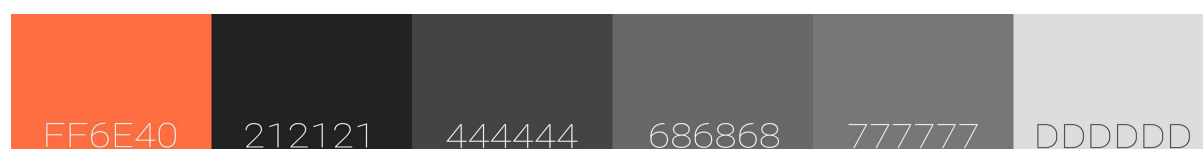
Grazie a Materialize CSS ed alla sua capacità di adattarsi alle dimensioni dello schermo, DDDParts risulta sempre proporzionato e con disposizioni diverse a seconda del tipo di dispositivo che si sta utilizzando. Per esempio su desktop vengono visualizzati tre progetti per riga, mentre su smartphone soltanto uno. In questo modo nessun elemento si trova in posti sbagliati e l'esperienza utente è sempre pulita e funzionale.



## - Dettagli che fanno la differenza -

Siamo fortemente convinti che ogni dettaglio, ogni piccolezza, sia parte integrante e soprattutto fondamentale di un sito. Sono le piccole cose che ti fanno affezionare a qualcosa, e proprio seguendo questa idea abbiamo inserito in DDDParts sfiziosi accorgimenti mirati a migliorare la chiarezza e l'esperienza utente. Alcuni esempi ne sono:

- il placeholder del campo password, che ad ogni secondo cambia, mostrando i criteri da rispettare per avere una password efficace ed accettata;
- i campi che richiedono una particolare sintassi; se l'utente non la rispetta si sottolineano di rosso mostrando l'errore commesso, altrimenti di verde;
- ad ogni aggiornamento o quando necessario, l'utente viene avvisato da un piccolo toast che compare per qualche secondo e poi se ne va, senza creare disturbo a nessuno;
- le proporzioni seguono quelle del Material Design, con padding, bordi e margini mai fuori luogo;
- al passaggio del puntatore su una scheda, questa viene evidenziata da una piccola sfumatura che, se cliccate, dà vita ad una onda, come se si stesse toccando uno specchio d'acqua;
- le immagini caricate non vengono mai storpiate, ma mantengono il rapporto originale, così ogni soggetto sarà proprio come nella realtà;
- tutti i colori seguono la palette che abbiamo creato in fase di progettazione, amalgamandosi in modo sinuoso ed elegante;





## - Tecnologie utilizzate -

DDDParts è stato realizzato utilizzando linguaggi e tecnologie che spaziano in tutto il campo del Web Development.

Oltre alle conoscenze apprese durante il corso di studi, abbiamo approfondito anche altri linguaggi e strumenti, quali Ajax, MySQLi e GitHub.

### HTML5, CSS, JavaScript, Materialize CSS

HTML5, CSS e JavaScript costituiscono la struttura portante di DDDParts, come di ogni altro sito web. Materialize CSS invece è un framework che fonde i tre linguaggi, fornendo elementi grafici e funzionalità in linea con il Material Design.

### jQuery, Ajax

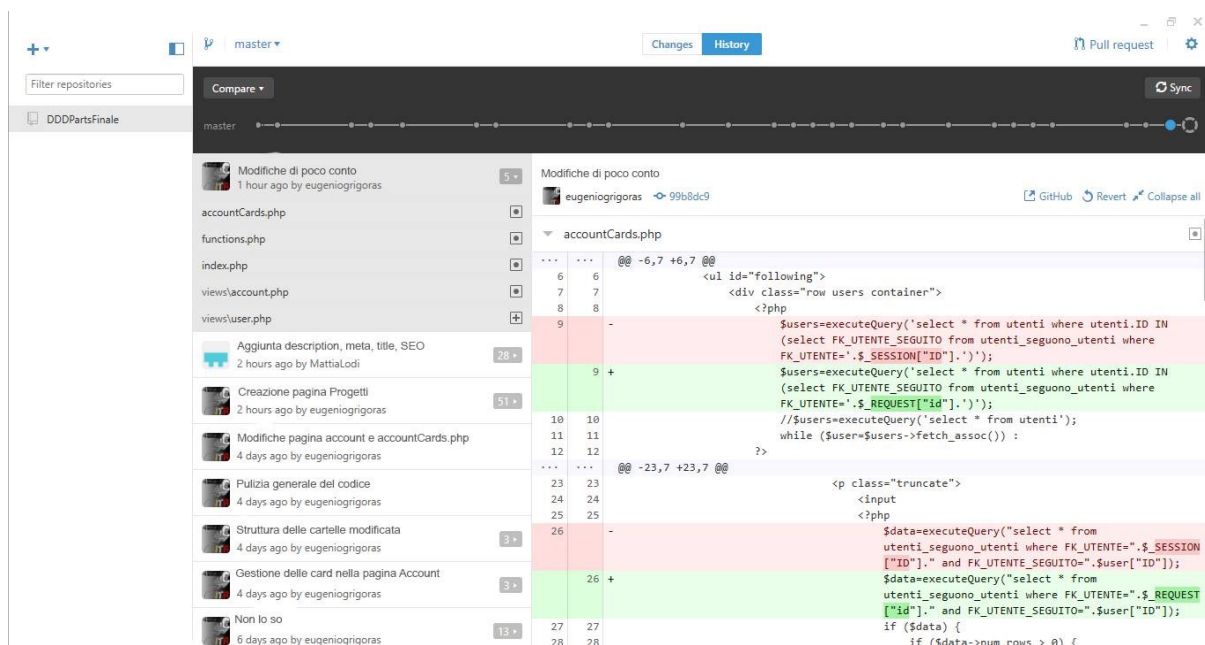
Le funzioni di jQuery, unite alla potenza di Ajax, ci permettono di fare chiamate PHP anche una volta che il caricamento della pagina è stato completato. Infatti in un qualunque momento è possibile aprire un canale di comunicazione con il server ed ottenere dati dinamicamente. Un esempio significativo si trova nella pagina account, nella quale, completato il caricamento della pagina, sono presenti delle schede informative. Una volta cliccata una di queste, il canale client-server viene aperto e vengono restituiti i dati desiderati, che appaiono in una sezione dedicata senza il refresh della pagina.

## Apache, MySQL, MySQLi, PHP, AlterVista

La gestione del database e la sua interrogazione vengono effettuate da PHP e MySQLi. Per lo sviluppo locale abbiamo inoltre utilizzato un server virtuale Apache con un database MySQL, mentre per la pubblicazione in rete ci siamo appoggiati ad AlterVista.

## GitHub

GitHub è un servizio web di hosting per lo sviluppo di progetti software, che usa il sistema di controllo di versione Git. Ci è stato utile per avere una cartella condivisa e sincronizzata dell'intero progetto e per tenere traccia di ogni modifica a ciascun file.



Interfaccia di GitHub Desktop:

- a sinistra troviamo la storia con le varie modifiche condivise
- a destra le singole modifiche, con in rosso il 'prima' e in verde il 'dopo'

## - Gestione dei file -

Per la gestione dei file abbiamo scelto di creare una cartella per ogni utente, nominandola con nome-cognome-mail. All'interno di essa è presente l'immagine del profilo, rinominata sempre come 'profile.jpg'. Oltre all'immagine, alla creazione di un nuovo progetto viene creata una nuova cartella nominata con l'ID che il progetto ha sul database e contenente tutti i file che lo compongono.

Questa gestione rende le operazioni di manutenzione molto semplici e tiene le cartelle ordinate e senza ridondanze.

## - Sicurezza e Privacy -

In termini di Privacy DDDParts chiede all'utente di accettare i termini e le condizioni di utilizzo in fase di registrazione; se non lo fa non può finalizzare la registrazione e quindi utilizzare la piattaforma.

Riguardo alla sicurezza invece tutte le password non vengono salvate sul database in chiaro, ma ne viene salvata l'impronta. In fase di login poi viene ricalcolata l'impronta di quella inserita e confrontata con l'impronta presente nel database.

Inoltre, subito dopo la registrazione, viene mandata una mail contenente un link di conferma dell'account, che non permette il login fino a che non è stato cliccato. Questa operazione evita la creazione di troppi account multipli e con false mail, salvaguardando DDDParts e il carico di lavoro del server.

## - SEO -

Al fine di garantire una più elevata rilevazione possibile da parte dei motori di ricerca e facilitare la indicizzazione del sito, ci siamo occupati anche di una parte relativa al SEO (Search Engine Optimization). Senza di essa infatti, anche cercando tutto il nome del sito su Internet, questo non risultava presente nei primi risultati.

Le operazioni che abbiamo svolto sono:

- aggiunta di descrizioni, titoli e keyword per ogni pagina e diversi in ciascuna di esse;

Esempio pagina login:

```
<title>Welcome back to DDDParts!</title>
<meta name="keywords" content="3D 3DParts
DDDParts printer stampante parti social
profile sharing 3d printer 3d printer for
sale 3d printer pen 3d printer reviews 3d
printer filament 3d printer price 3d printer
kit 3d printer designs 3d printer 3d printer
software login enter logon log in log on
account mail password remember me">
<meta name="description" content="Welcome
back to DDDParts! Please log in to explore
DDDDProjects.">
```

- integrazione di Flight, un veloce, semplice ed estensibile framework per PHP. La sua funzione è quella di associare ad ogni URL una funzione di callback, grazie alla quale i nostri link hanno una particolare costruzione SEO Friendly. Ad esempio la pagina account viene restituita inserendoo '[dddparts.altervista.org/account](http://dddparts.altervista.org/account)', senza chiamate a nessun file .php, in modo tale che gli utenti non possano accedere a pagine alle quali non sono

autorizzati. La gestione del routing viene svolta dalla pagina `index.php`, il cuore del sito, contenente tutte le relazioni URL-pagina.

Esempio account:

```
Flight::route('/account(/@message)',  
function($message){  
    if (logged()) {  
        Flight::render('account',  
array('message' => $message));  
    } else {  
        Flight::redirect('/login');  
    }  
});
```

- implementazione della Sitemap per l'indicizzazione, al fine di facilitare l'esperienza dell'utente e per 'far capire' al motore di ricerca quali e quante pagine ha il sito;
- registrazione sulla Google Search Console per monitorare e gestire la presenza del sito nei risultati della ricerca di Google. Questo ci è servito per capire meglio come viene visualizzato il sito su Google e come ottimizzarne le prestazioni;
- grazie a Google Analytics e all'implementazione del codice fornito da Google stessa, siamo in grado di ricevere dati aggiornati su come i visitatori utilizzano il sito, come sono arrivati su quest'ultimo e cosa possiamo fare per incentivarli a tornare.

## - Sitografia -

[thingiverse.com](https://thingiverse.com)

[secure.php.net/manual](https://secure.php.net/manual)

[stackoverflow.com](https://stackoverflow.com)

[flightphp.com](https://flightphp.com)

[materializecss.com](https://materializecss.com)

[github.com](https://github.com)

[w3school.com](https://w3school.com)

[materialup.com](https://materialup.com)

[site.uplabs.com](https://site.uplabs.com)

[codepen.io](https://codepen.io)

[design.google.com](https://design.google.com)

[caniuse.com](https://caniuse.com)