



UNIVERSITÀ
DEGLI STUDI
DI PADOVA

TECNOLOGIE WEB
A.A. 2024/2025



BookOverflow

Relazione progetto

Bazzan Matteo - matteo.bazzan.1@studenti.unipd.it - 2076422

Bernardello Alessandro - alessandro.bernardello.1@studenti.unipd.it - 2077674

Mathew Aleena - aleena.mathew@studenti.unipd.it - 2068242

Ribon Luca - luca.ribon@studenti.unipd.it - 2075516 - **Referente**

Indirizzo sito web

<https://tecweb.studenti.math.unipd.it/lribon>

Credenziali accessi al sito

Ruolo	Username	Password
User	user	user
Utente	utente	utente
Amministratore	admin	admin

Indice

1. Introduzione	1
1.1. Perché BookOverflow	1
2. Analisi	1
2.1. Descrizione del prodotto	1
2.2. Target di utenza	1
2.3. Funzionalità	2
2.3.1. Registrazione e accesso	2
2.3.2. Personalizzazione dell'esperienza	2
2.3.3. Ricerca e visualizzazione dei libri	2
2.3.4. Scambi	2
2.3.5. Recensioni	2
3. Progettazione	3
3.1. Layout	3
3.2. Tipologie di utente	3
3.3. API esterne	3
3.3.1. Google Books API	3
3.3.2. Picsum API	4
3.4. Base di dati	4
3.5. Struttura del sito	6
3.5.1. Header	6
3.5.2. Breadcrumb	7
3.5.3. Footer	7
3.5.4. Home	7
3.5.5. Esplora	7
3.5.5.1. Match per te	7
3.5.5.2. Potrebbe piacerti anche	7
3.5.5.3. Esplora tutti	7
3.5.5.4. Libri più scambiati	7
3.5.6. Come funziona	8
3.5.7. Libro	8
3.5.8. Accedi	8
3.5.9. Registrati	8
3.5.10. Profilo	8
3.5.10.1. Modifica generi	8
3.5.10.2. Modifica liste	9
3.5.10.3. I tuoi scambi	9
3.5.11. 404	9
3.6. BISAC	9

4. Implementazione	9
4.1. Organizzazione del lavoro	9
4.2. Frontend	10
4.2.1. Validazione lato client	10
4.2.2. Stampa	10
4.2.3. Ricerca libri	10
4.3. Backend	11
4.3.1. Routing	11
4.3.2. Connessione al database	11
4.3.3. Costruzione delle pagine	11
4.3.4. Validazione lato server	12
 5. Accessibilità e usabilità	12
5.1. Palette colori	13
5.2. Font	13
5.3. Compatibilità	14
 6. SEO	15
6.1. Scelta di keyword e description	15
6.2. Performance	15
6.3. Indicizzazione e alberatura	15
 7. Testing	15
7.1. Google Lighthouse	15
7.2. Total Validator	18
 8. Altre motivazioni progettuali	18
8.1. Account forniti	18
8.2. Warning nel console log	18
 9. Guida installazione progetto	18
9.1. .htaccess	18
9.2. .env	18



1. Introduzione

1.1. Perché BookOverflow

BookOverflow è il posto dove i libri non finiscono mai. Come *StackOverflow* per i programmatori, qui gli amanti della lettura trovano un «overflow» di storie, connessioni e possibilità, tutto senza spendere un euro.

La bellezza di BookOverflow risiede nel suo potere di trasformare libri dimenticati sugli scaffali in opportunità, inoltre si impegna a ridurre l'impatto ambientale dell'industria editoriale, incentivando pratiche sostenibili.

2. Analisi

2.1. Descrizione del prodotto

BookOverflow è una piattaforma web che permette agli utenti registrati di accedere a una vasta selezione di libri in modo gratuito e risolve il problema dell'accumulo di libri negli scaffali.

L'obiettivo principale è facilitare gli scambi tra utenti: chi cerca un libro specifico può trovare chi lo possiede e, viceversa, chi offre un libro può connettersi con chi è interessato a riceverlo. Per semplificare la ricerca, gli utenti possono:

- Indicare libri desiderati o generi preferiti (es. fantascienza, romanzi storici), evitando di selezionare manualmente ogni titolo.
- Esplorare liberamente tutti i libri disponibili sulla piattaforma, sia quelli offerti per lo scambio che quelli cercati da altri utenti.

In questo modo, la piattaforma unisce praticità, risparmio e personalizzazione, rendendo gli scambi di libri accessibili a tutti.

2.2. Target di utenza

La piattaforma si rivolge a tutti coloro che intendono aggiornare la propria raccolta di libri, per fare questo permette agli utenti di navigare ed eseguire delle ricerche nei seguenti casi:

- l'**utente sa di preciso cosa cerca** (*tiro perfetto*), quindi può:
 - eseguire una ricerca mirata tramite il form di ricerca della pagina «*esplora tutti*».
 - popolare la sua lista dei desideri e successivamente consultare la pagina «*match per te*», per visualizzare i possibili scambi per ottenere i libri che corrispondono ai suoi desideri
- l'**utente sa vagamente cosa cerca** (*trappola per aragoste*), quindi ha tre possibilità:
 - aggiungere i libri in suo possesso alla lista *libri offerti* e consultare la pagina «*potrebbe piacerti anche*» per visualizzare i libri offerti dagli utenti interessati ai propri libri;
 - aggiungere i suoi generi preferiti nella piattaforma, e successivamente accedere alla pagina «*esplora tutti*» dove è presente una funzione per filtrare i libri visualizzati usando i suoi generi preferiti;
 - accedere alla pagina «*esplora tutti*» e filtrare i risultati tramite una ricerca.
- l'**utente preferisce esplorare** (*pesca a strascico*) in modo ampio i libri disponibili nella piattaforma, quindi può accedere alla pagina «*esplora tutti*» e visualizzare tutti i libri disponibili



- l'**utente ha già individuato** (*boa di segnalazione*) i contenuti di suo interesse, quindi può:
 - salvare la pagina del libro tramite *bookmark* dato che l'ISBN del libro viene passato come parametro *GET*;
 - salvare i risultati di una ricerca tramite *bookmark* dato che sia l'input di ricerca che l'utilizzo del filtro *generi preferiti* vengono passati come parametri *GET*.

Si prevede un utilizzo tramite dispositivi con performance e dimensioni del display varie, quindi la piattaforma è stata progettata per essere responsive e performante sulla maggior parte dei dispositivi.

2.3. Funzionalità

2.3.1. Registrazione e accesso

La piattaforma offre agli utenti la possibilità di registrarsi per accedere a sezioni e funzionalità personalizzate. Una volta completata la registrazione, gli utenti potranno effettuare l'accesso utilizzando il proprio username o email, assieme alla propria password.

È inoltre prevista la possibilità di disconnettersi dal proprio account o di eliminarlo in modo permanente.

2.3.2. Personalizzazione dell'esperienza

La piattaforma consente agli utenti di personalizzare la propria esperienza attraverso la definizione di interessi specifici, tra cui l'inserimento di libri nella **lista dei desideri** e l'indicazione dei **generi letterari preferiti**. Gli utenti possono inoltre inserire i volumi che intendono mettere a disposizione per lo scambio.

2.3.3. Ricerca e visualizzazione dei libri

La piattaforma adotta un sistema di suggerimenti personalizzati che, tenendo conto dei generi e libri desiderati dall'utente, nonché dell'offerta disponibile, propone una selezione di libri in linea con i suoi interessi.

Per una ricerca più mirata, gli utenti possono consultare l'elenco completo dei libri registrati, filtrando i risultati per titolo, autore, ISBN o generi preferiti.

2.3.4. Scambi

Se un utente è in possesso di un libro desiderato da un altro utente, può inviare una proposta di scambio.

L'utente ricevente potrà decidere se accettare o rifiutare la proposta attraverso un'apposita interfaccia dedicata.

2.3.5. Recensioni

Dopo l'accettazione di uno scambio, gli utenti possono lasciare una recensione sull'esperienza. Il sistema di feedback permette sia a chi offre che a chi accetta lo scambio di condividere la propria opinione, aiutando così gli altri utenti a scegliere con chi scambiare.



3. Progettazione

3.1. Layout

In fase di progettazione si è deciso di adottare un **layout responsive** che permette di variare le caratteristiche layout basandosi su degli intervalli di dimensione della finestra di visualizzazione; questo garantisce un'esperienza di navigazione ottimale su dispositivi con dimensioni e risoluzioni diverse.

Il sito è stato testato per essere fruibile su dispositivi mobili, tablet e desktop.

3.2. Tipologie di utente

Abbiamo individuato le seguenti tipologie di utenti:

- **Utente non registrato:**
 - Può conoscere la piattaforma tramite la pagina iniziale e le pagine di corredo che forniscono ulteriori dettagli su di essa;
 - Può visualizzare i libri presenti nella piattaforma tramite la pagina «*esplora*», ma non avrà accesso ai suggerimenti personalizzati in quanto senza un account non è possibile definire i propri interessi; di conseguenza non potrà accedere alle pagine «*match per te*» e «*potrebbe piacerti anche*»;
- **Utente registrato:**
 - Può compiere tutte le azioni di un utente non registrato;
 - Può definire i propri interessi, tramite lista dei libri desiderati e/o generi preferiti;
 - Può visualizzare i suggerimenti personalizzati, ovvero «*match per te*» e «*potrebbe piacerti anche*»;
 - Può accedere alla funzionalità di scambio, sia come offerente che come richiedente;
 - Può inserire delle recensioni sugli utenti in relazione ad uno scambio.
- **Amministratore:**
 - Può visualizzare gli utenti registrati nella piattaforma;
 - Può visualizzare gli scambi di ogni utente;
 - Può eliminare gli utenti dalla piattaforma;
 - Può accedere alla pagina «*esplora*» per verificare i libri presenti nella piattaforma.

3.3. API esterne

3.3.1. Google Books API

Utilizziamo le API di Google Books in varie parti del progetto.

Queste, durante il processo di aggiunta di un libro, ci permettono di fornire all'utente una funzionalità di ricerca del libro desiderato in base a vari parametri (titolo, autore, anno di pubblicazione ed ISBN).

Le informazioni necessarie al funzionamento del sito (come ad esempio l'immagine di copertina ed i generi) vengono recuperate automaticamente mediante l'API e non è necessario che un utente le compili manualmente.

Abbiamo inoltre deciso di far sì che le immagini di copertina dei libri nelle varie pagine vengano recuperate direttamente tramite URL esterni al servizio Google, invece di scaricarle e fornirle direttamente dal nostro sito.



Questo per evitare di appropriarci e distribuire materiale senza l'adeguato permesso andando ad infrangere eventuali leggi sul copyright.

L'utilizzo delle API di Google Books con questa modalità è consentito nella relativa documentazione; inoltre questa API è pubblicamente accessibile anche senza registrazione e presenta rate-limit molto elevati (~100k richieste/giorno).

3.3.2. Picsum API

Utilizziamo le API di Picsum per assegnare ad ogni utente registrato un'immagine di profilo unica e copyright-free poiché utilizza le immagini di Unsplash con licenza gratuita

Per assegnarle usiamo questa chiamata:

https://picsum.photos/seed/{email_utente}/200/300

il seed utilizzato per generare l'immagine è l'email dell'utente stesso, essendo la chiave primaria della tabella Utente nel DB l'immagine risulta univoca per ogni utente.

3.4. Base di dati

Lo schema ER della base di dati è il seguente:

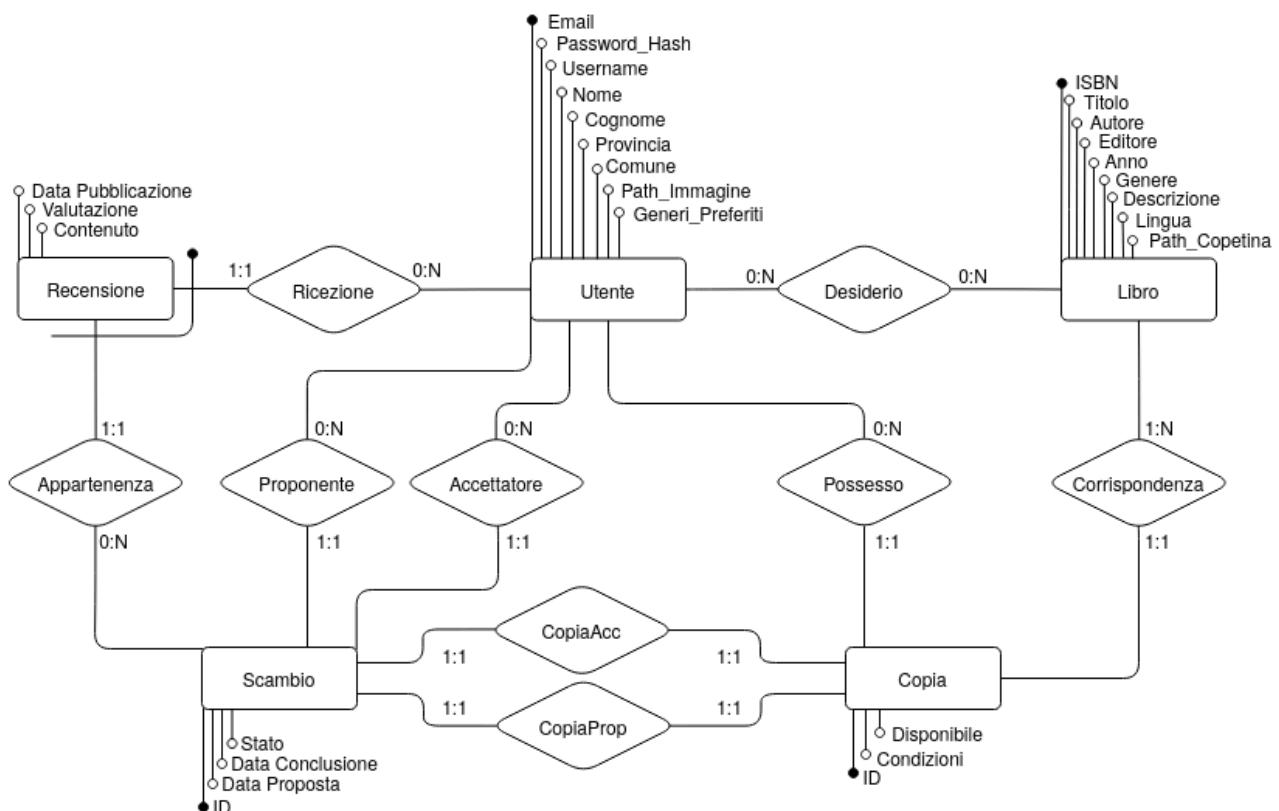


Figura 1: Schema ER della base di dati

La base di dati è stata implementata utilizzando MariaDB, ed è composta dalle seguenti tabelle:



- **Utente**

- email: chiave primaria
- password_hash: hash della password dell'utente, comprensivo di un salt univoco.
- username: nome utente univoco
- nome
- cognome
- provincia
- comune
- path_immagine: percorso all'immagine del profilo
- generi_preferiti: generi letterari preferiti dall'utente

- **Libro**

- ISBN: chiave primaria. È di tipo `VARCHAR(50)` poichè l'API Google Books certe volte restituisce identificativi più lunghi di un ISBN standard. Utilizzare una stringa a lunghezza variabile ed un limite meno restrittivo ci permette di evitare errori durante l'inserimento
- titolo
- autore
- editore
- anno: anno di pubblicazione
- genere
- descrizione
- lingua
- path_copertina: percorso all'immagine di copertina

- **Copia**

- ID: usato come chiave primaria in modo che un utente possa avere più copie dello stesso ISBN (cosa non possibile se usassimo (ISBN, proprietario) come chiave primaria); viene usato un `AUTO_INCREMENT` per garantire l'unicità
- ISBN: chiave esterna che fa riferimento a Libro
- proprietario: chiave esterna che fa riferimento a Utente
- disponibile: `BOOLEAN` che rappresenta lo stato di disponibilità allo scambio della copia, utile per indicare un libro che era nella lista dell'utente ma che è stato scambiato
- condizioni: `ENUM("nuovo", "come nuovo", "usato ma ben conservato", "usato", "danneggiato")` che rappresenta lo stato di usura della copia

- **Desiderio**

- email, ISBN: chiave primaria composta, chiavi esterne che si riferiscono rispettivamente Utente e Libro

- **Scambio**

- ID: chiave primaria, viene usato un `AUTO_INCREMENT` per garantire l'unicità
- emailProponente: identifica l'utente che propone lo scambio, chiave esterna che fa riferimento a Utente
- emailAccettatore: identifica l'utente che riceve la proposta di scambio, chiave esterna che fa riferimento a Utente
- idCopiaProp, idCopiaAcc: rappresentano rispettivamente la copia fisica offerta da chi propone lo scambio e la copia fisica offerta da chi riceve la proposta; chiavi esterne che fanno riferimento a Copia



- ▶ `dataProposta`, `dataConclusione`: date di proposta e conclusione (la conclusione avviene in caso di accettazione o rifiuto) dello scambio; `dataProposta` ha `CURRENT_DATE` come default per semplificare l'inserimento
- ▶ `stato`: `ENUM`("in attesa", "accettato", "rifiutato") che rappresenta lo stato dello scambio
- **Recensione**
 - ▶ `emailRecensito`: identificatore dell'utente che riceve la recensione; chiave esterna che fa riferimento a `Utente`
 - ▶ `idScambio`: identificatore dello scambio a cui si riferisce la recensione; chiave esterna che fa riferimento a `Scambio`
 - ▶ `dataPubblicazione`: data di pubblicazione della recensione, ha `CURRENT_DATE` come default per semplificare l'inserimento
 - ▶ `valutazione`: valore intero (usiamo `TINYINT` per ottimizzare lo spazio occupato) compreso tra 1 e 5
 - ▶ `contenuto`: testo della recensione; colonna di tipo `TEXT` per permettere la scrittura di recensioni di lunghezza variabile fino a 65,535 caratteri, questo non rappresenta un problema perché la documentazione di MariaDB indica che le stringhe di lunghezza variabile allocano solo lo spazio effettivamente occupato

C'è poi un secondo schema che è stato utilizzato per la gestione delle posizioni geografiche, che è composto dalle seguenti tabelle:

- `regioni`
- `province`
- `comuni`

Abbiamo deciso di non integrare quest'ultimo nello schema principale perché viene utilizzato solamente per ottenere un elenco di province e comuni da usare nel form di registrazione quindi abbiamo preferito non complicare ulteriormente la struttura dello schema principale.

3.5. Struttura del sito

Nel sito tutte le pagine utilizzano lo schema a tre pannelli che risponde alle seguenti domande:

- **Dove sono?** Informazione ottenibile tramite il `title` o la `breadcrumb`;
- **Dove posso andare?** Informazione ottenibile tramite la `navbar` o il `footer`;
- **Di cosa si tratta?** Informazione ottenibile tramite il `main`, ovvero il contenuto principale della pagina;

Inoltre si trova risposta anche alle seguenti domande:

- **Da chi è gestita questa pagina?** Informazione ottenibile tramite `footer`;
- **Dove posso trovare informazioni più approfondite?** Informazione ottenibile tramite la pagina `come funziona`, pagina sempre raggiungibile tramite `navbar`.

3.5.1. Header

L'header è composto da:

- **Logo**: il logo del sito;
- **Titolo**: il nome del sito;
- **Navbar**: contiene i collegamenti alle pagine principali del sito, ovvero `home`, `esplora` e `come funziona`;
- **Switch tema**: permette di cambiare il tema del sito tra chiaro e scuro;



- **Accesso:** contiene il collegamento alla pagina *accedi* se l'utente non è autenticato, altrimenti contiene il collegamento al *profilo*;

3.5.2. Breadcrumb

La breadcrumb è presente in tutte le pagine del sito e permette all'utente di capire in che punto della gerarchia del sito si trova e quali sono le pagine di livello superiore rispetto a dove si trova.

3.5.3. Footer

Il footer è composto da dei link alle altre pagine principali del sito, logo e nome del sito, una breve descrizione e gli autori. Inoltre è presente una **dichiarazione di non responsabilità** relativa all'utilizzo delle immagini ottenute tramite *Google Books API*.

3.5.4. Home

È la **landing page** del sito, contiene una breve descrizione del servizio offerto, una lista che mostra alcuni dei libri più scambiati ed un collegamento ulteriore alla pagina di accesso; il tutto viene completato da degli elementi decorativi.

3.5.5. Esplora

Questa pagina mostra le diverse opzioni di ricerca e visualizzazione dei libri presenti nella piattaforma; in particolare permette di accedere alle pagine «*esplora tutti*» e «*libri più scambiati*» a utenti registrati e non, e alle pagine «*match per te*» e «*potrebbe piacerti anche*» solo agli utenti registrati.

3.5.5.1. Match per te

Questa pagina permette di visualizzare i libri che corrispondono ai desideri dell'utente e che sono offerti da altri utenti a cui interessano i libri offerti dall'utente che esegue la ricerca.

3.5.5.2. Potrebbe piacerti anche

Questa pagina permette di visualizzare i libri offerti dagli utenti che sono interessati ai libri proposti dall'utente che esegue la ricerca; non sono presenti solo i libri che corrispondono ai desideri dell'utente ma tutti quelli che rispettano la condizione precedentemente definita in modo che l'utente possa considerare anche libri che non conosce e che quindi non ha inserito nella lista dei desideri.

3.5.5.3. Esplora tutti

Questa pagina permette di visualizzare tutti i libri presenti nella piattaforma, e di filtrarli tramite la funzione di ricerca che cercherà corrispondenze per titolo, autore e ISBN. Inoltre gli utenti registrati possono filtrare i libri usando i loro generi preferiti tramite un apposito bottone.

3.5.5.4. Libri più scambiati

Questa pagina permette di visualizzare i libri più scambiati nella piattaforma, è accessibile sia come utenti registrati che non.



3.5.6. Come funziona

Pagina che fornisce ulteriori dettagli riguardo ad obbiettivi, vantaggi e una breve spiegazione del funzionamento della piattaforma.

Include anche un collegamento alla pagina di accesso.

3.5.7. Libro

Questa pagina permette di visualizzare i dettagli di uno specifico libro, inoltre, se l'utente ha eseguito l'accesso, permette di visualizzare e di proporre uno scambio con gli utenti che possiedono una copia del libro.

3.5.8. Accedi

Questa pagina permette agli utenti di accedere alla piattaforma tramite username o email e password; per poter accedere l'utente deve prima aver completato la registrazione.

3.5.9. Registrati

Questa pagina permette agli utenti di registrarsi alla piattaforma, inserendo i dati richiesti ovvero: nome, cognome, provincia e comune dove ci si trova, username, email, password e conferma della password.

3.5.10. Profilo

Se il profilo visualizzato non è quello dell'utente autenticato allora la pagina permette di:

- visualizzare nome, cognome, posizione geografica, media delle valutazioni e immagine del profilo;
- visualizzare i generi preferiti;
- visualizzare i libri offerti e desiderati dall'utente;
- contattare l'utente tramite email;

Altrimenti la pagina permette di:

- visualizzare nome, cognome, posizione geografica, media delle valutazioni e immagine del profilo;
- visualizzare i generi preferiti;
- visualizzare i libri offerti e desiderati dall'utente;
- accedere alla pagina di modifica del profilo;
- accedere alla pagina di modifica dei generi;
- accedere alla pagina «*i tuoi scambi*»;
- accedere alla pagina «*recensioni ricevute*»;
- accedere alla pagina di modifica delle liste;
- uscire dall'account;
- eliminare l'account;

A fine pagina è presente anche una call to action che invita ad esplorare le possibilità di scambio.

3.5.10.1. Modifica generi

Questa pagina permette di visualizzare e scegliere i propri generi preferiti.



3.5.10.2. Modifica liste

Le pagine di modifica delle liste permettono di visualizzare tutti i libri di quella lista, mostrando anche disponibilità e condizione. Inoltre permette di aggiungere o eliminare libri dalla lista.

3.5.10.3. I tuoi scambi

Questa pagina permette di visualizzare tutti gli scambi in cui l'utente è coinvolto, sia come proponente che come ricevente; è possibile anche aggiungere le recensioni relative agli scambi accettati.

Se l'utente è il proponente può annullare gli scambi in corso, se l'utente è il ricevente può accettare o rifiutare gli scambi in corso.

3.5.11. 404

Questa pagina viene visualizzata quando l'utente tenta di accedere ad una pagina che non esiste, contiene un messaggio di errore e un collegamento alla pagina iniziale.

3.6. BISAC

La gestione dei generi dei libri è basata sullo standard BISAC, ovvero lo standard internazionale utilizzato per catalogare i libri.

4. Implementazione

4.1. Organizzazione del lavoro

Per la realizzazione del progetto è stato utilizzato un **repository su GitHub**, in cui sono presenti la **branch** main e diverse feature branch. Per la gestione dell'infrastruttura utilizzata per testare il progetto in locale è stato utilizzato un file **Docker Compose** assieme ad un **Dockerfile**, il secondo è stato usato per configurare alcuni parametri non gestibili tramite il Docker Compose.

Il lavoro è stato svolto in modo misto sia sincrono che asincrono, ed è stato suddiviso nel seguente modo:

- **Aleena Mathew:**
 - Progettazione dell'interfaccia grafica;
 - Realizzazione dell'interfaccia grafica tramite HTML e CSS;
 - Realizzazione dei layout di stampa;
 - Ideazione dei contenuti testuali e delle immagini;
 - Stesura della relazione;
- **Alessandro Bernardello:**
 - Progettazione del database;
 - Implementazione del database;
 - Implementazione delle API del database;
 - Implementazione del routing delle pagine;
 - Implementazione del backend delle pagine riguardante l'interazione con il database e relativi controlli;
 - Ottimizzazione delle immagini;
 - Stesura della relazione;



- **Luca Ribon:**

- Progettazione dell'interfaccia grafica;
- Progettazione del database;
- Implementazione del database;
- Implementazione delle API del database;
- Implementazione del backend delle pagine riguardante l'interazione con il database e relativi controlli;
- Stesura della relazione;

- **Matteo Bazzan:**

- Progettazione dell'interfaccia grafica;
- Realizzazione dell'interfaccia grafica tramite HTML e CSS;
- Implementazione delle API del database;
- Implementazione del routing delle pagine;
- Implementazione del backend delle pagine riguardante l'interazione con il database e relativi controlli;
- Implementazione degli script JavaScript per la gestione dei componenti grafici;
- Stesura della relazione;

Tutti i membri del gruppo hanno contribuito all'implementazione e/o alla verifica dell'accessibilità del sito.

4.2. Frontend

4.2.1. Validazione lato client

La validazione lato client viene effettuata tramite HTML, in particolare vengono eseguiti controlli tramite **espressioni regolari** per verificare che i dati inseriti dall'utente rispettino il formato richiesto; viene anche verificato che i campi obbligatori siano stati compilati. Lo scopo di questa validazione è quello di dare un **feedback** anticipato all'utente in modo che possa correggere eventuali errori prima di inviare i dati al server.

L'implementazione è stata svolta in HTML per questioni di accessibilità, in particolare per favorire una gestione migliore della navigazione tramite screen reader; oltretutto è stata ridotta la dipendenza da JavaScript.

Questa scelta non ha portato indebolimenti dal punto di vista della sicurezza dato che i controlli vengono comunque eseguiti lato server, inoltre JavaScript è manipolabile lato client tanto quanto l'HTML.

4.2.2. Stampa

Per ogni pagina viene incluso un layout di stampa, che riorganizza le informazioni principali dalla pagina sistemandole in un formato adeguato, principalmente rimuovendo i colori di background non necessari per la stampa.

4.2.3. Ricerca libri

La ricerca dei libri da aggiungere alle liste, utilizza le Google Books API; per fornire all'utente una **ricerca automatica** e immediata abbiamo deciso di implementare un meccanismo di *auto-fetching* basato su un **timer** che viene fatto partire nel momento in cui l'utente smette di digitare per più di 600 millisecondi.

In questo modo evitiamo un numero eccessivo di richieste alle API e soprattutto il contenuto non viene aggiornato troppo velocemente causando un **sovraccarico cognitivo** dell'utente.



Per migliorare l'usabilità e l'accessibilità di questa ricerca automatica per gli utenti che utilizzano screen reader è stato aggiunto un *alert* che viene che notifica all'utente la ricezione dei risultati della sua ricerca.

4.3. Backend

4.3.1. Routing

Tramite il file `.htaccess`, abbiamo impostato delle regole per indirizzare tutte le richieste al file `index.php`, creando un router centrale che gestisce dinamicamente il routing dell'applicazione. Questo approccio ci permette di:

- Nascondere le estensioni dei file (es. `.php`) garantendo URL puliti e user-friendly
- Gestire tutte le richieste GET attraverso un unico entry-point.
- Implementare errori 404 personalizzati che vengono visualizzati quando si tenta di accedere a pagine non definite nel sistema
- Bloccare l'accesso diretto ai file di configurazione e script PHP, aumentando la sicurezza

4.3.2. Connessione al database

Utilizziamo la classe `Database` presente in `src/dbAPI.php` per gestire tutte le query al database ed astrarre le operazioni di connessione e disconnessione.

La connessione al database viene effettuata durante la costruzione di un'istanza della classe `Database` e persiste fino alla sua distruzione.

Il meccanismo di query viene astratto tramite i metodi `query_to_array` e `void_query`, i quali chiamano internamente il metodo `prepare_sql_statement`.

In questa maniera abbiamo potuto forzare l'uso di **prepared statements** per prevenire attacchi di tipo SQL injection.

Abbiamo deciso di gestire eventuali errori durante l'esecuzione delle query stampando l'eccezione lanciata da `mysqli` nei log del server, in modo da evitare di mostrare informazioni sensibili all'utente ed al contempo avere messaggi di errore di facile comprensione.

La maggior parte degli errori mostrati all'utente sono quindi generici e generati altrove, generalmente nel file `php` che fa uso della classe.

Nel caso di errori specifici per condizioni che richiedono controlli lato database, come ad esempio la scelta di un username già in uso durante la creazione di un account, abbiamo creato una classe `CustomException` come estensione della classe `Exception`.

Questo approccio ci consente di distinguere le eccezioni con messaggi personalizzati e di stamparne il messaggio all'utente.

4.3.3. Costruzione delle pagine

Il rendering delle pagine avviene interamente in modo dinamico tramite PHP a partire da template HTML. I file template (posizionati in `src/templates/`) contengono ancora strutturali sotto forma di commenti HTML dedicati (ad esempio: `<!-- [navbar] -->`) i quali vengono sostituiti con il contenuto dinamico generato dal backend.

Questo metodo ha vari vantaggi:

- Separazione netta tra struttura e contenuto



- Possibilità di riutilizzare componenti comuni come header e footer
- Gestione corretta degli url tramite la sostituzione automatica di placeholder come `<!-- prefix -->` con il percorso relativo corretto. il funzionamento di asset statici (CSS, immagini) e link interni anche in percorsi nidificati.

La logica risolve problematiche di routing estremo: in caso di richieste a percorsi non validi con multiple sottodirectory (es. `/livello1/livello2/pagina-inesistente`), il sistema mantiene la corretta risoluzione dei percorsi nella pagina 404 personalizzata, evitando la rottura dei riferimenti ad asset statici tipica che avverrebbe con un link relativo statico.

4.3.4. Validazione lato server

La piattaforma implementa un sistema di validazione multi-livello per garantire l'integrità e la sicurezza dei dati inseriti dagli utenti durante la registrazione.

- Validazione lato server (PHP):
 - Controlli strutturali:
 - Verifica che la richiesta sia di tipo POST e che tutti i campi siano presenti.
 - Confronto tra password e conferma password per evitare errori di digitazione.
 - Sanitizzazione:
 - Uso di `htmlspecialchars()` su nome e cognome per prevenire attacchi XSS.
 - Regex per controllare il formato dei dati
- Gestione degli errori:
 - Reindirizzamento alla pagina precedente con messaggi specifici di errore (es. «Credenziali non valide»).
 - Utilizzo di prepared statements e classe DBAccess per prevenire SQL injection.
- Sicurezza aggiuntiva:
 - Hash della password con algoritmo moderno (*Bcrypt*) con aggiunta di salt randomizzato per prevenzione di attacchi mediante *Rainbow Tables* precalcolate

Questo approccio garantisce usabilità (feedback immediato all'utente sugli errori) e robustezza (protezione da input malevoli), allineandosi alle best practice per applicazioni web sicure odierne.

5. Accessibilità e usabilità

Si è tenuto conto dell'accessibilità e dell'usabilità del sito durante tutta la progettazione e l'implementazione, per garantire un'esperienza di navigazione ottimale a tutti gli utenti, indipendentemente dalle loro capacità.

Nei tre fogli CSS, dedicati alla versione desktop, mobile e stampa, sono state utilizzate misure relative per assicurare che il contenuto rimanga leggibile e accessibile su dispositivi con dimensioni diverse.

È stato scelto un linguaggio semplice e chiaro per facilitare la comprensione anche da parte di utenti con disabilità cognitive.

Per l'orientamento dell'utente all'interno del sito è stata inserita una *breadcrumb* in tutte le pagine, che permette di capire in che punto della gerarchia del sito ci si trova e di tornare alle pagine di livello superiore.



Per gli utenti che navigano tramite screen reader o altri dispositivi di supporto sono presenti aiuti alla navigazione che permettono di saltare direttamente ai contenuti principali della pagina, evitando di dover ascoltare tutto il contenuto come *salta al contenuto*, che permette di saltare direttamente al contenuto principale della pagina.

Inoltre, per facilitare l'utente a ritornare all'inizio della pagina è stato inserito un bottone, *scroll to top* che permette di tornare all'inizio della pagina evitando di dover scorrere nuovamente tutto il contenuto, questo è particolarmente utile per utenti con disabilità motorie e per utenti che navigano tramite dispositivi mobile.

Sempre tenendo conto degli utenti che navigano tramite screen reader, sono stati utilizzati gli attributi *lang* per le parole in lingue straniere per consentirne la pronuncia corretta, inoltre sono stati utilizzati gli attributi *aria-label* per facilitare la comprensione dei vari elementi della pagina, in particolare per la gestione dei messaggi di errore.

L'accessibilità è stata verificata sia manualmente che tramite strumenti automatici come ad esempio *Lighthouse* di Google Chrome, VoiceOver, Silktide, Wave Evaluation Tool e Stark Per garantire la compatibilità il sito è stato testato su diversi browser come Google Chrome, Mozilla Firefox, Microsoft Edge e Safari.

Non è stata realizzata una site map in quanto il sito presenta una gerarchia ampia e poco profonda che abbiamo ritenuto facilmente navigabile.

5.1. Palette colori

La palette colori è stata costruita per garantire un contrasto sufficiente tra i testi e lo sfondo, in modo da garantire una buona leggibilità anche a persone con disturbi visivi.

Per facilitare ulteriormente l'orientamento all'interno del sito e la creazione di una mappa mentale della struttura del sito, i colori dei link e delle pagine visitate sono stati scelti in modo che siano distinguibili tra loro; nella versione chiara i link sono di colore nero o marrone se visitati, mentre nella versione scura sono di colore bianco o marrone se visitati.

Siamo consapevoli che utilizzando la versione chiara potrebbe non esserci un sufficiente contrasto tra link visitato e non

Siamo consapevoli che la scelta dei colori potrebbe non garantire un sufficiente contrasto tra link visitato e non visitato quando viene utilizzato il tema chiaro ma abbiamo deciso di mantenere tale scelta dando priorità al contrasto tra testo e sfondo e alla coerenza tra i colori del sito.

Per costruire una prima palette di base abbiamo usato <https://coolors.co/>, successivamente è stata adattata per essere accessibile e in modo che i colori rispecchino il messaggio trasmesso dall'elemento a cui sono applicati.

Per verificare l'accessibilità dei colori è stata utilizzata l'estensione [WCAG Contrast Checker](#) che si basa sui requisiti di WCAG 2.2; grazie a questo strumento abbiamo constatato che la nostra palette rispetta il livello WCAG AAA.

5.2. Font

Abbiamo scelto i font nel catalogo di [Google Font](#), in particolare è stato scelto **Onest** come font *sans-serif*, e **Gentium Book Plus** come font *serif* per la pagina e per il logo.



Per garantire un'esperienza visiva armoniosa e accessibile, le dimensioni dei font sono state definite utilizzando GRTCalculator, uno strumento che applica il principio della sezione aurea ($\approx 1:1.618$). Questa scelta progettuale si basa su tre pilastri:

- **Armonia estetica:** è una proporzione matematicamente bilanciata, storicamente associata a forme piacevoli alla percezione umana. Crea una gerarchia tipografica coerente, dove titoli, sottotitoli e corpo del testo si relazionano in modo naturale, migliorando l'estetica generale del sito.
- **Leggibilità e accessibilità:** garantisce che le proporzioni mantengano un contrasto adeguato tra testo e sfondo, rispettando i criteri WCAG.
- **Responsività e coerenza:** il design rimane coerente anche su schermi con risoluzioni diverse. Questo approccio elimina la necessità di regolazioni manuali, ottimizzando il workflow di sviluppo.

Per salvaguardare la compatibilità è stata usata la regola :

```
@supports not (font-size: clamp(1rem, 1vi, 1rem))
```

nel caso in cui il browser non supporti la funzione clamp, la dimensione del font viene impostata in modo statico.

5.3. Compatibilità

Per verificare la compatibilità il sito è stato testato sui seguenti dispositivi e browser:

- Android 14
 - Chrome 132.0
 - Firefox 127.0
- Android 13
 - Chrome 132.0
- Android 12
 - Chrome 132.0
- iOS 18.2.1
 - Safari 17.4.1
- MacOS Sequoia 15.2
 - Chromium 132.0.6834.160
- Windows 11
 - Chrome 133.0.6943.35
 - Firefox 134.0.2
- Windows 10
 - Chrome 133.0.6943.35
 - Firefox 134.0.2
- Fedora 40
 - Chromium 130.0.6723.91
 - Firefox 133.0.3
 - Zen Browser 1.0.2-b.2
- ArchLinux (rolling, 2025-01-30)
 - Zen Browser 1.0.1-a.10



- Firefox 133.0.3
- Ungoogled-Chromium 128.0.6613.113
- Chrome 120.0.6099.129

6. SEO

6.1. Scelta di keyword e description

Lo scopo principale della piattaforma è quello di scambiare libri tra utenti, e attualmente, per questioni di implementazione, ci si rivolge all'utenza nel territorio italiano; per questo le **keyword scelte** sono: «scambio libri in Italia, libri, scambio, libri Italia, trova libri, BookOverflow».

Per il tag **description** invece abbiamo individuato «BookOverflow è un sito web che permette di scambiare libri con altri utenti in Italia, trova subito il tuo prossimo scambio.» che racchiude diverse keyword e una **call to action** stando al di sotto dei 150 caratteri.

6.2. Performance

La velocità di caricamento è un fattore fondamentale per migliorare la SERP, per questo abbiamo deciso di compiere le seguenti azioni per migliorare la performance del sito:

- ottimizzazione del caricamento delle immagini comprimendole in formato **.AVIF**, questo permette di ridurre il peso delle immagini mantenendo una qualità eccellente;
- utilizzo di font più leggeri in formato **.woff2**;
- **configurato il caricamento delle risorse** in HTML in modo che vengano caricati prima i font e le immagini necessarie per la visualizzazione della pagina;
- utilizzo della clausola **font-display: swap** per garantire che il testo sia visibile anche se il font non è ancora stato scaricato;

Non abbiamo minimizzato i file CSS e JavaScript perché la consegna richiede esplicitamente di non farlo.

6.3. Indicizzazione e alberatura

Abbiamo utilizzato il file **robots.txt** per evitare che vengano indicizzati file contenenti dati sensibili, configurazioni, script che vengono usati solo come utility e pagine dedicate ad un utilizzo più approfondito della piattaforma che non si vogliono rendere direttamente accessibili tramite indicizzazione.

Per quanto riguarda l'alberatura del sito è stata tenuta il più possibile piatta mantenendo un livello di profondità massimo di 3 livelli.

7. Testing

7.1. Google Lighthouse

Per verificare la qualità del progetto dal punto di vista della SEO abbiamo usato il tool **Google Lighthouse**. In questo modo abbiamo ottenuto una valutazione per le seguenti categorie:



- **Performance:** in ogni pagina otteniamo un punteggio maggiore o uguale a 90 sia nella modalità desktop che in quella mobile;
- **Accessibility:** in ogni pagina otteniamo un punteggio maggiore o uguale a 98 sia nella modalità desktop che in quella mobile;
- **Best Practice:** in ogni pagina otteniamo un punteggio maggiore o uguale a 100 sia nella modalità desktop che in quella mobile;

Facciamo notare che il punteggio visualizzato è più basso quando il sito viene ospitato nel server del dipartimento a causa dell'utilizzo del protocollo *HTTP* (e non *HTTPS*) su una porta diversa da 80

- **SEO:** in ogni pagina otteniamo un punteggio maggiore o uguale a 100 sia nella modalità desktop che in quella mobile;

Di seguito delle foto raffiguranti i punteggi ottenuti durante i test eseguiti in locale, dove l'unica differenza è che viene *bypassato* il controllo per l'utilizzo di *HTTPS*:

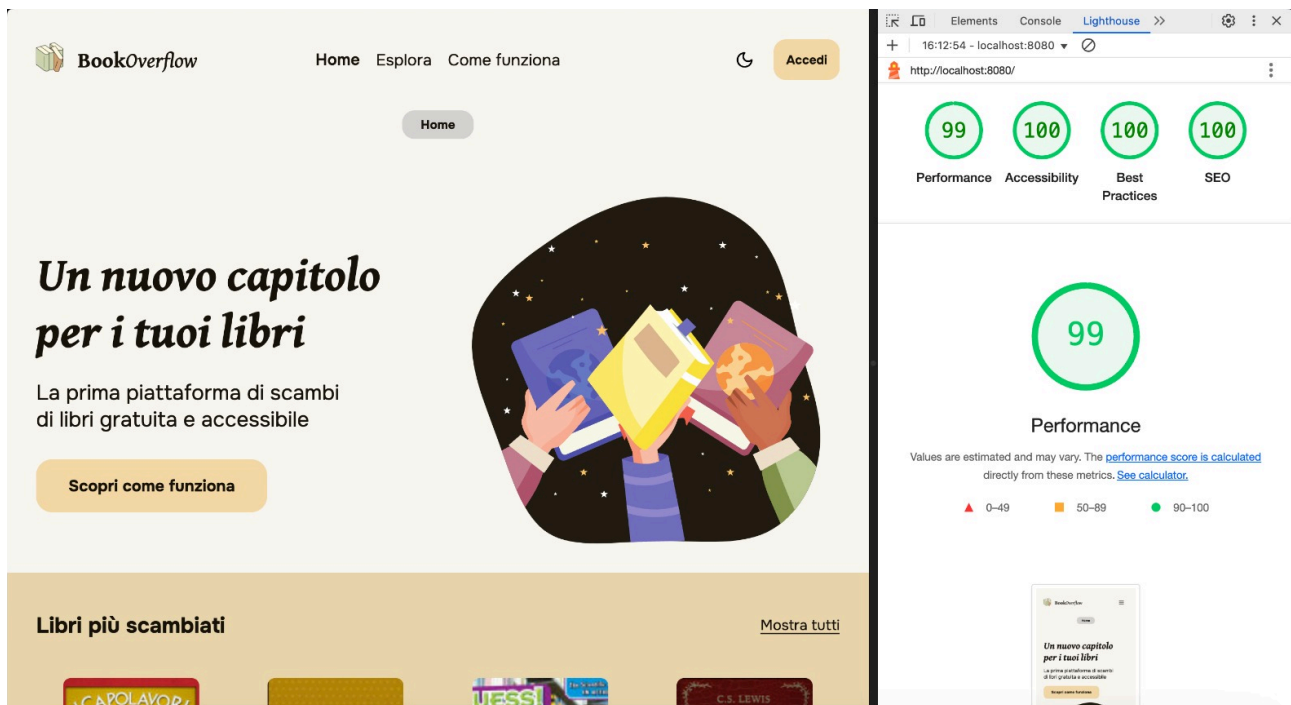


Figura 2: Risultati Google Lighthouse della pagina «Home»

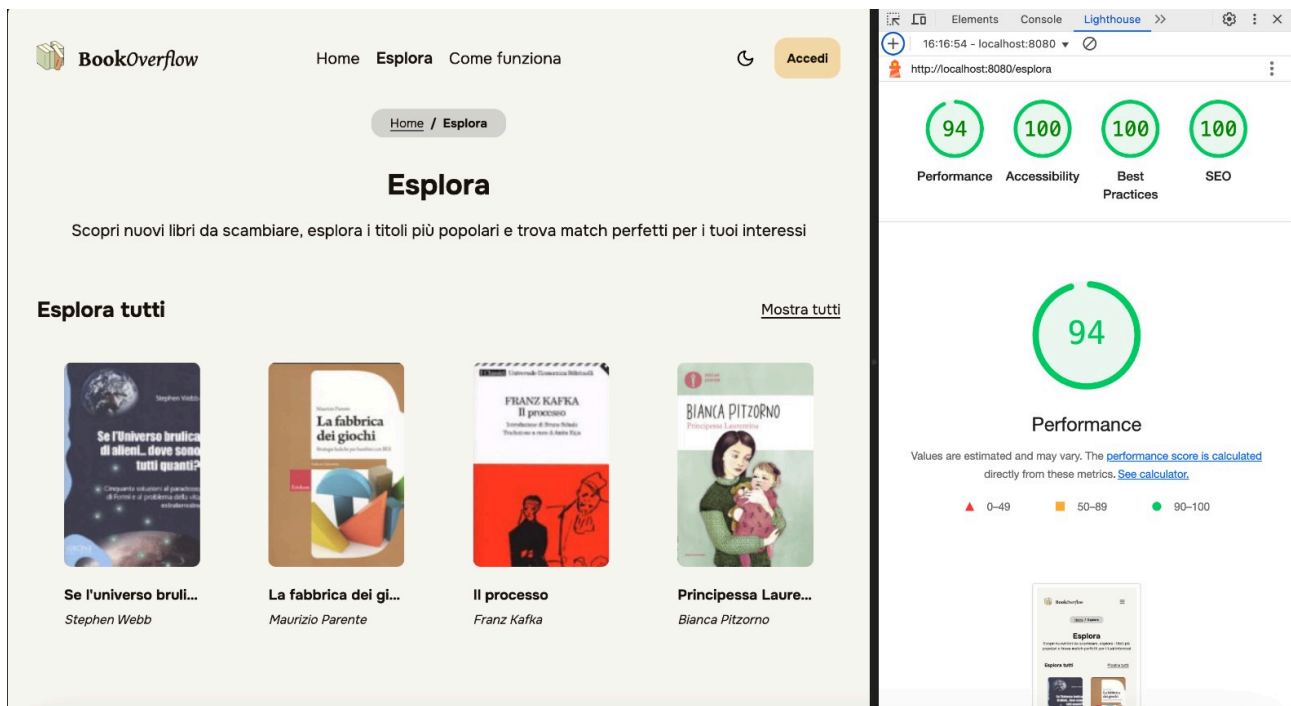


Figura 3: Risultati Google Lighthouse della pagina «Esplora»

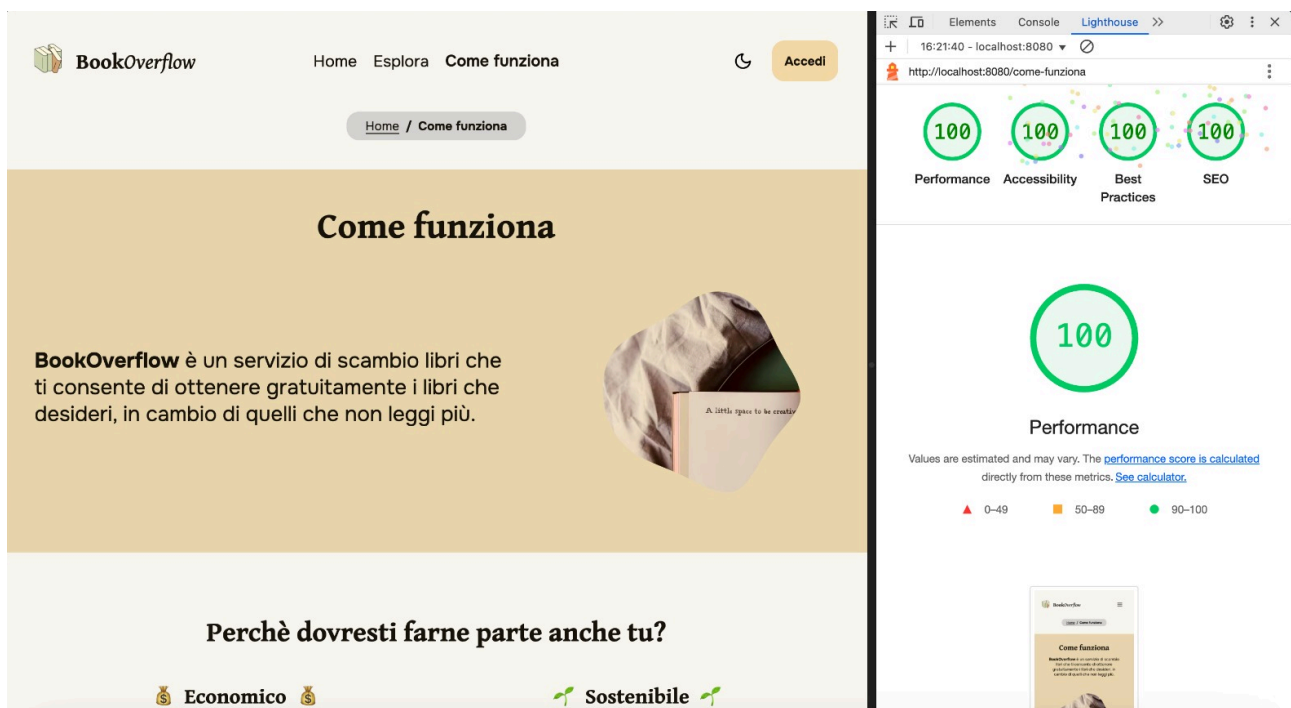


Figura 4: Risultati Google Lighthouse della pagina «Come funziona»



7.2. Total Validator

Per garantire un'esperienza inclusiva e conforme agli standard internazionali, tutte le pagine del sito sono state sottoposte a test automatizzati tramite Total Validator.

I controlli sono stati eseguiti seguendo le linee guida WCAG 2.2 al livello più elevato (AAA), e il programma ha non rilevato errori in nessuna delle pagine.

I risultati sono stati integrati con verifiche manuali tramite screen reader NVDA e VoiceOver.

8. Altre motivazioni progettuali

8.1. Account forniti

Vengono forniti due account, Utente e User, per poter testare tutte le funzionalità del sito.

8.2. Warning nel console log

Abbiamo verificato che, eseguendo il progetto nel server dell'università, vengono ritornati nel console log alcuni warning riguardanti la mancanza di HTTPS nelle richieste eseguite; questo accade perché le richieste vengono fatte tramite HTTP a causa della configurazione del server stesso.

9. Guida installazione progetto

9.1. .htaccess

Per il corretto funzionamento del progetto, è necessario impostare il reindirizzamento delle richieste al file `index.php`.

Affinchè ciò avvenga, è sufficiente utilizzare il file `.htaccess` fornito assieme ai file del progetto, il quale ha il seguente contenuto:

```
RewriteEngine On
RewriteCond %{REQUEST_URI} !\.(css|jpg|jpeg|png|gif|avif|js|ttf|svg)$ [NC]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ index.php [L]

RewriteRule ^\.env$ index.php [L]
```

9.2. .env

I parametri di configurazione al database vanno salvati in un file `.env`, da posizionare nella cartella radice del progetto.

Un esempio di configurazione è il seguente:

```
DB_HOST=localhost
DB_DATABASE=bookoverflow
DB_USERNAME=test
```



```
DB_PASSWORD=test  
PREFIX={{web_directory}}
```

Se il sito viene ospitato in una sottocartella invece che nella cartella radice di un certo dominio/sottodominio, è importante specificare il percorso nel quale questo viene ospitato mediante la chiave PREFIX.

Questo consente al codice php di riscrivere correttamente i vari link e riferimenti presenti nel sito in modo che puntino alle risorse corrette.

Se invece il sito viene ospitato nella cartella radice, tale valore può essere lasciata vuota o anche omessa dal file di configurazione.