**BOOK STORE**

**Obiettivo del sito**

Il sito è stato pensato per un Book Shop/ book store online. Il target del sito pertanto è esteso a persone di qualsiasi età che hanno la necessità di acquistare uno o più libri.

**Scopo del sito**

Lo scopo del sito è quello di fornire ai clienti un modo semplice, veloce e intuitivo per acquistare libri online.

**Struttura logica**

**Front-end**

Il sito si apre con la schermata di login che consente ad un qualunque utente che già si è registrato di accedere con il proprio account. Un nuovo utente dovrà effettuare la registrazione cliccando “registrati ora”.

Il form di registrazione richiede all’utente i dati necessari per creare un nuovo account: nome, email e password (verrà richiesta la conferma della password) e tipo utente. Completata la fase di login a seconda del tipo di user ci si troverà di fronte a due schermate differenti.

Un user normale verrà indirizzato verso la home del sito con la possibilità di visualizzare i libri presenti nello shop, effettuare ordini, vedere gli ordini fatti e il loro stato e inviare commenti/recensioni agli admin. Nella sezione shop è possibile fare l’ordine degli articoli.

Il cliente può effettuare l’ordine di uno o più articoli presenti nello shop. Verranno richiesti come dati obbligatori: nome, email, telefono, possibilità di pagare subito o alla consegna, tipologia di pagamento (paypal, carta di credito etc.), stato, città e PIN della carta utilizzata.

Un admin viene indirizzato alla sua pagina admin. Nella home troverà una dashboard con tutti i dati aggiornati del sito. L’admin può modificare gli ordini e il loro stato. L’admin in più può caricare nuovi prodotti nello shop e visualizzare i messaggi inviati dai vari utenti.

**Back-end**

Il backend è stato sviluppato in php facendo utilizzo di MAMP e per il database phpmyadmin. Per accedere al sito è sufficiente caricare i file in htdocs e avviare la pagina php “login.php”. Ogni pagina del sito inizializza una connessione con il database shop\_db (attraverso la pagina config.php).

*Lato User:*

**Login.php**

Login prende in input mail e password dell’utente e tramite una query verifica prima nel database se è presente il profilo dell’utente. Se esiste nel database allora viene effettuato l’accesso all’account. Se nel database non viene trovato nessun profilo significa che le credenziali sono sbagliate oppure che è necessario creare un nuovo account. Il sito risponderà in ogni caso facendo visualizzare a schermo “Email o password non corrette!”.

**Register.php**

Vengono presi in input nome, email, password e la password di conferma. Inizialmente tramite una query viene verificato che non sia già presente nel database. Se è già nel database viene visualizzato a schermo che l’utente esiste già. Se non è presente nel database viene effettuato il confronto tra la password e la password di conferma. Se coincidono viene creato il nuovo account e viene effettuato il login, in caso contrario viene visualizzato a schermo il messaggio “Le due password devono coincidere!”.

**Shop.php**

Nella pagina shop.php visibile solo da un cliente viene effettuata un query sulla tabella prodotti del database per mostrare all’utente i prodotti disponibili, il loro prezzo e inoltre viene caricata l’immagine utilizzando il campo immagine della tabella “orders” come percorso nella cartella “uploaded\_img”. Tramite un form e un pulsante è possibile selezionare la quantità da acquistare di uno stesso prodotto e aggiungerlo al carrello.

**Contact.php**

La pagina contact.php è importante poiché è l’unico modo che hanno gli utenti per interagire con gli admin. Viene data la possibilità all’utente di mandare tramite un form e una query un messaggio all’admin per segnalare eventuali problemi, ma anche per suggerire modifiche/migliorie.

La pagina presenta quindi un unico form che prende in input nome, mail e messaggio in stringa che l’utente vuole lasciare e lo salva nella tabella messages del database. I messaggi inviati in questo modo saranno poi visibili sulla pagina admin\_contacts riservata agli admin.

**Cart.php**

La pagina cart.php disponibile solo per gli users mostra a ogni cliente tutti gli ordini effettuati. Viene eseguita una query iniziale che prende in input lo user id salvato in una variabile all’apertura della connessione al database e interroga la tabella cart per recuperare tutti gli ordini fatti dal cliente.

La pagina presenta un form che consente all’utente di eliminare il prodotto dal carrello tramite un pulsante (“X”). Tale pulsante fa partire una query che elimina l’ordine dalla tabella cart. Nel form è presente inoltre il bottone aggiorna che consente all’utente di aggiornare la quantità di un prodotto ordinato. Viene eseguita una query che seleziona l’ordine che si vuole modificare dalla tabella cart e modifica la quantità.

Tramite il pulsante elimina tutto, il cliente può svuotare il carrello eliminando tutti i prodotti che aveva inserito. Viene utilizzato lo user\_id, salvato in una variabile all’apertura della connessione per selezionare dalla tabella orders del database tutti gli ordini che corrispondono a tale user\_id e eliminarli.

**Checkout.php**

La pagina di checkout presenta un form che l’utente deve compilare per procedere con l’ordine, una volta inseriti i dati l’utente conferma l’ordine con il pulsante Ordina, che avvia una query che prende in input tutti i dati inseriti dall’utente e li va a salvare nella tabella orders del db.

*Lato admin:*

**Admin\_page.php**

Nella pagina admin\_page viene mostrata tramite una query che recupera tali dati una dashboard con tutti i dati riguardanti il sito di book\_store: il totale degli acquisti completati, se ci sono messaggi e quanti sono, gli utenti nel database del sito sia admin che users normali e il numero di account users e admin, il numero di prodotti aggiunti e il numero di ordini effettuati.

**Admin\_products.php**

La pagina admin\_orders.php consente agli admin di caricare nuovi prodotti all’interno dello shop, visibili poi dai clienti nella pagina shop.php. Viene effettuata una query per mostrare all’admin tutti i prodotti caricati sullo shop. L’admin ha la possibilità di rimuovere un prodotto cliccando sul pulsante elimina: in questo caso il prodotto tramite query viene rimosso dal database. Tramite il pulsante aggiorna si può modificare nome, immagine e prezzo del prodotto. L’admin inoltre può aggiungere un nuovo prodotto tramite un form inserendo nome, prezzo e immagine del prodotto che si vuole caricare. Nel database vengono salvati nome, prezzo e una stringa contentente il “nome” dell’immagine, mentre l’immagine vera e propria viene salvata nella cartella upload\_img. Se si effettua l’eliminazione di un prodotto viene rimossa anche la sua immagine dalla cartella upload\_img.

**Admin\_orders**

La pagina admin orders mostra all’admin un elenco degli ordini fatti dai vari clienti, ciascuno con nome, numero, indirizzo, totale , spesa. Inizialmente viene fatta una query andando a selezionare tutti i record della tabella orders del db per recuperare tutti gli ordini dei clienti. Se il numero degli ordini è zero viene mostrato all’admin il messaggio “nessun ordine effettuato”. Altrimenti viene visualizzato l’elenco degli ordini con le varie informazioni citate prima.

**Admin\_users**

La pagina admin users mostra all’admin tutti gli utenti registrati ciascuno con la propria mail, username e livello di accesso (admin, users). All’admin è concesso tramite un pulsante di poter eliminare un utente dal sito. Inizialmente tramite una query vengono recuperati tutti gli utenti della tabella users, poi con un ciclo while vengono mostrati nella pagina ciascuno con i dati prima citati e con un pulsante elimina che consente di eliminare tale utente da quelli registrati.

**Admin contacts**

La pagina admin\_contacts è molto simile a admin\_users, con la differenza che mostra all’admin un elenco dei messaggi mandati da parte di un qualsiasi user. Viene inizialmente fatta una query sulla tabella messages del db recuperando tutti i messaggi mandati, e di ciascun messaggio l’utente che l’ha inviato con la propria mail e il contenuto in stringa del messaggio. Se il numero dei messaggi in tabella è zero viene mostrato il messaggio “Nessun messaggio!”. L’admin può leggere il messaggio e una volta letto lo può eliminare tramite apposito pulsante elimina. Viene quindi generata una query che seleziona il messaggio dalla tabella messages e lo elimina dal db.

**Parti non sviluppate e criticità**

Per mostrare tutte le funzionalità del sito a ogni utente è lasciata la possibilità di registrarsi come user o admin senza alcun controllo e ad ogni admin è consentito eliminare qualsiasi utente registrato compreso un altro admin. Inoltre avrei voluto dare la possibilità agli admin di rispondere ai messaggi mandati dai clienti.

**Sviluppo**

\* 19/10 – Sviluppo delle pagine about us, config, register, login e logout

\*26/10 – Fine del frontend, inizio dello sviluppo del backend

\* 4/11 – Problemi con il salvataggio delle immagini sul database

\* 10/11 – Problema salvataggio immagini sul DB risolto

\*11/11 – Problema con la pagina di checkout

\*23/11 – Risolti i problemi con la pagina checkout

\*24/11 – aggiunte funzioni di password\_hash e password\_verify per criptare password, sito terminato