

Progetto Ecommerce Web

2024 - 2025

PANORAMICA

Il progetto consiste nel realizzare una piattaforma online Ecommerce che tratta la vendita di orologi. Gli orologi venduti possono essere nuovi o usati, appartenenti a qualsiasi marchio di qualsiasi valore. La piattaforma ammette acquirenti oppure venditori che decidono di vendere i propri orologi.

OBIETTIVI

1. Grafica accattivante ma allo stesso tempo elegante, in grado di rispecchiare la modernità della cultura orologiera.
2. Uso di tecniche di programmazione avanzate object oriented: ORM ed uso di Template Engine
3. Sistemi di gestione dati sicuri e inviolabili
4. Fruibilità del sito e accessibile a tutti.

IMPOSTAZIONE DEL SITO

Il sito web verrà mostrato sui motori di ricerca attraverso la sua home page dove viene dato il benvenuto all'utente e gli vengono mostrati tutti i listini prodotti inseriti dai venditori.

L'utente sarà in grado di scegliere il suo prodotto preferito dal catalogo prodotti con una ricerca filtrata dell'articolo che è più di suo gradimento.

L'utente potrà aggiungere articoli al carrello e acquistarli da registrato oppure no, gli articoli verranno comunque salvati nel carrello per un totale di giorni.

Il sito web disporrà di una pagina login e una per la registrazione, se l'utente effettua acquisti dopo il login è in grado di salvarli e visualizzare nella sua area personale gli acquisti fatti in passato.

Se l'utente registrato si qualifica come venditore, avrà accesso alla sua area personale dove avrà visione degli articoli da lui venduti e la possibilità di inserirne dei nuovi. Avrà inoltre dei dati statistici sulle sue vendite visibili e sarà in grado di ricevere notifiche sui nuovi ordini effettuati dai clienti.

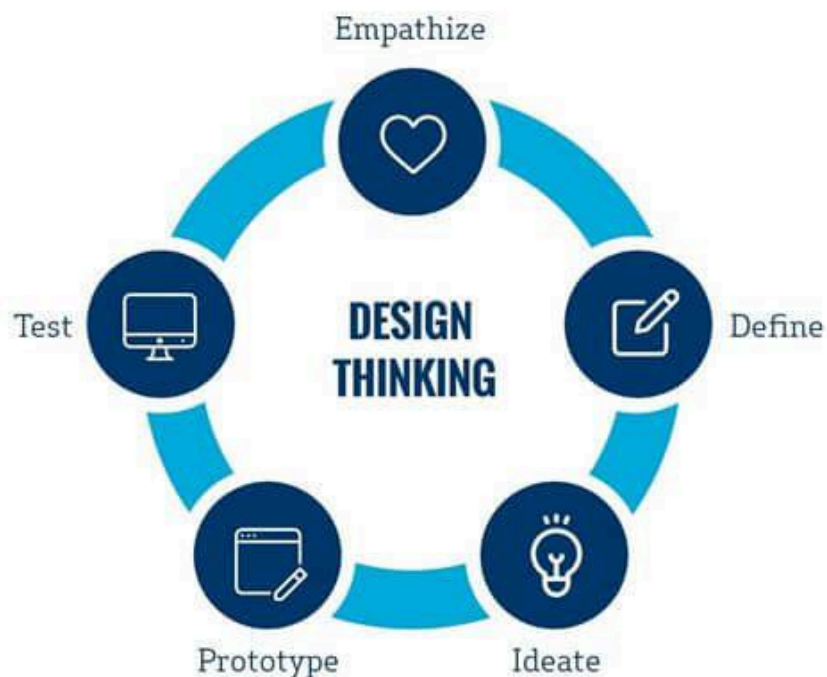
LIBRERIE ESTERNE

Twig: motore di templating che facilita la separazione tra la logica di applicazione e la presentazione visiva, migliorando la leggibilità e la manutenibilità del codice

PDO: offre un'interfaccia sicura e indipendente dal database per interagire con i dati, proteggendo da SQL injection e migliorando la portabilità del codice

DESIGN

Il focus del design è stato principalmente sui principi di Mobile First, User Centered Design e Accessibilità. Questa sezione descrive il processo di progettazione che ha portato alla realizzazione del sito, con particolare attenzione al metodo utilizzato per sviluppare il design, che ha coinvolto attivamente gli utenti attraverso diverse fasi.



Processo di Produzione del Design

Il processo di progettazione del sito web si è sviluppato in vari passaggi chiave, che hanno garantito che il design fosse effettivamente in linea con le esigenze degli utenti e fosse accessibile su tutti i dispositivi. Le fasi sono state progettate per seguire il ciclo di feedback continuo, coinvolgendo gli utenti fin dalle prime fasi del progetto.

Sessione di Design Partecipativo con un Gruppo di Utenti

La fase iniziale del progetto ha visto il coinvolgimento diretto di un gruppo di utenti attraverso una sessione di design partecipativo. Durante questo incontro, sono stati discussi i requisiti, le preferenze e le aspettative degli utenti finali riguardo al sito web e alla sua usabilità. Gli utenti hanno avuto l'opportunità di esprimere le proprie opinioni su diversi aspetti del sito, tra cui:

- La disposizione dei contenuti e la facilità di navigazione
- Le funzionalità desiderate (ad esempio, filtri per la ricerca degli orologi)
- Le preferenze visive, come colori, font e stile grafico
- L'importanza di una user experience fluida su dispositivi mobili.

Il design partecipativo ha permesso di raccogliere feedback preziosi, che hanno costituito la base per i passaggi successivi. Questo approccio è stato fondamentale per garantire che il sito fosse costruito attorno alle necessità reali degli utenti, facendo emergere aspetti da considerare sin dall'inizio, come l'importanza della navigazione mobile-friendly.

Creazione dei Mockup

Sulla base dei feedback ricevuti durante la sessione di design partecipativo, sono stati sviluppati i mockup del sito. I mockup sono stati creati con l'obiettivo di tradurre le idee in una rappresentazione visiva chiara e funzionale. In questa fase, il design ha adottato il principio di Mobile First, progettando prima per schermi più piccoli e successivamente adattando il layout per dispositivi più grandi, come i tablet e i desktop.

Il design è stato pensato in modo da rispondere a diverse necessità:

Navigazione semplice e intuitiva, con un menu chiaro e facilmente accessibile

Adattabilità alle dimensioni dello schermo, con un layout che si adatta perfettamente a dispositivi mobili e desktop

Visibilità dei prodotti: i mockup sono stati progettati per far risaltare gli orologi, con immagini di alta qualità e informazioni chiare su ciascun prodotto

Call to Action evidenti: bottoni per l'acquisto, il carrello e la navigazione sono stati posizionati in modo strategico, per guidare l'utente verso l'acquisto in maniera semplice ed efficace.

La creazione di questi mockup ha coinvolto l'uso di strumenti di progettazione visiva come Balsamiq, per creare prototipi interattivi che hanno facilitato la visualizzazione e la navigazione del sito.

Focus Group di Controllo e Modifica del Design

Dopo aver creato i mockup, è stato organizzato un focus group di controllo con un nuovo gruppo di utenti per valutare l'efficacia del design. Il focus group ha avuto il compito di testare i mockup e fornire feedback critici riguardo la navigazione, l'usabilità, l'estetica e l'efficacia delle funzionalità proposte. Questo passaggio ha avuto un'importanza cruciale, in quanto ha permesso di raccogliere opinioni sulla facilità d'uso, sull'accessibilità, e sulla coerenza dell'esperienza utente.

Alcuni degli aspetti presi in considerazione durante il focus group erano:

Usabilità mobile: assicurarsi che il sito fosse facilmente navigabile anche su dispositivi mobili, considerando la crescente importanza dell'utilizzo di smartphone per gli acquisti online.

Accessibilità: valutazione della leggibilità del testo, del contrasto dei colori, e della facilità di navigazione anche per utenti con disabilità visive o motorie.

Facilità di completamento dell'acquisto: l'efficacia del flusso di acquisto e il posizionamento delle call-to-action sono stati esaminati per garantire che il sito supportasse l'utente dal momento in cui seleziona un prodotto fino al completamento dell'acquisto.

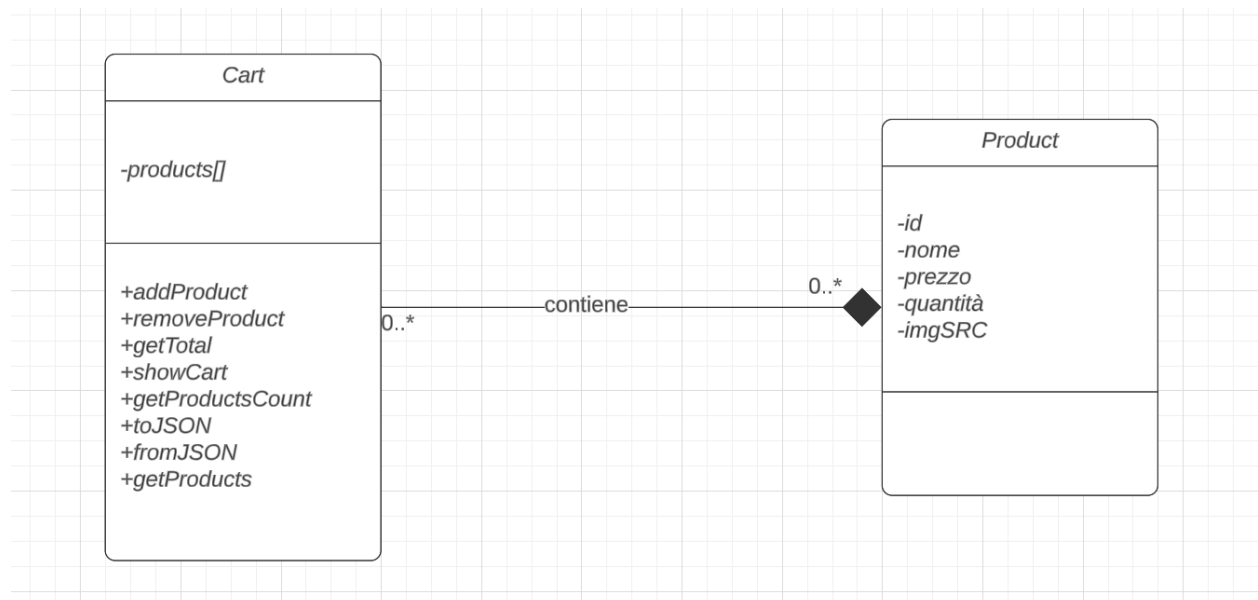
Sulla base del feedback ottenuto, sono stati apportati alcuni aggiustamenti al design:

- Ottimizzazione della navigazione mobile, migliorando la visibilità dei menu e l'accessibilità ai prodotti.
- Aggiunta di descrizioni per ciascun orologio, con informazioni aggiuntive sulle caratteristiche e il prezzo.
- Modifica del contrasto dei colori e dei font per migliorare l'accessibilità, in particolare per gli utenti con disabilità visive.

SVILUPPO

Gestione del carrello per gli utenti

Il carrello è interamente gestito lato frontend con Javascript, sfruttando due classi: Cart e Product.



Lato frontend viene reindirizzato su html ogni prodotto contenuto nel carrello, attraverso una programmazione imperativa del DOM, sfruttando direttamente le sue API come `createElement` o `appendChild` per evitare XSS (Cross-Site Scripting).

Routing degli URL ed MVC

Invece che usare la classica programmazione php dove ogni file viene mappato come url abbiamo valutato l'uso di un Dispatcher per catturare l'url inserito su browser e reindirizzare così pagine diverse a seconda di esso, in questo modo ogni pagina sfrutta il controller per accedere alla view e passargli dati direttamente dal model.

ORM e BaseModel

La classe BaseModel rappresenta una implementazione base di un ORM (Object-Relational Mapping), fornendo un'interfaccia per operare su tabelle di database come fossero oggetti PHP. Essa è progettata per essere estesa da modelli specifici che rappresentano tabelle del database, semplificando le operazioni di CRUD (Create, Read, Update, Delete) e altre interazioni con il database.

Principali caratteristiche e funzionalità

1. Mappatura dei campi
2. Metodi CRUD
3. Dinamismo tramite metodi magici
4. Preparazione dei dati
5. Validazione
6. Supporto alla selezione parziale

Vantaggi:

Astrazione del database: Semplifica l'interazione con il database evitando l'uso diretto di query SQL nella logica applicativa.

Manutenibilità: La centralizzazione delle operazioni CRUD in una singola classe riduce la duplicazione del codice.

Estendibilità: Può essere facilmente estesa per aggiungere funzionalità specifiche ai modelli.

Limitazioni:

Assenza di gestione avanzata delle relazioni: La classe non gestisce relazioni tra tabelle (es. `hasMany`, `belongsTo`), una caratteristica comune negli ORM completi come Doctrine o Eloquent.

Dipendenza da un database preconfigurato: Presuppone l'esistenza di una connessione a un database tramite la classe DB.

Overhead per piccole applicazioni: Potrebbe risultare eccessivo in progetti molto semplici.

Implementazione Login

Il codice gestisce la logica per il login di un utente in PHP. Quando viene ricevuta una richiesta POST, il sistema recupera l'email e la password dai dati inviati dal modulo. Questi valori vengono passati al metodo `validateUser` del modello `UserModel`, che verifica la validità delle credenziali. All'interno di `validateUser`, viene eseguita una query per cercare un utente nel database con l'email fornita. Se l'utente viene trovato e la password hashata nel database corrisponde alla password fornita tramite `password_verify`, i dati dell'utente vengono restituiti. Se l'utente è valido, vengono salvate le informazioni rilevanti dell'utente nella sessione, come email, nome, cognome, ruolo di venditore, indirizzo e città. La sessione viene quindi chiusa con `session_write_close` per garantire che i dati siano salvati, e l'utente viene reindirizzato alla homepage.

Implementazione Registrazione

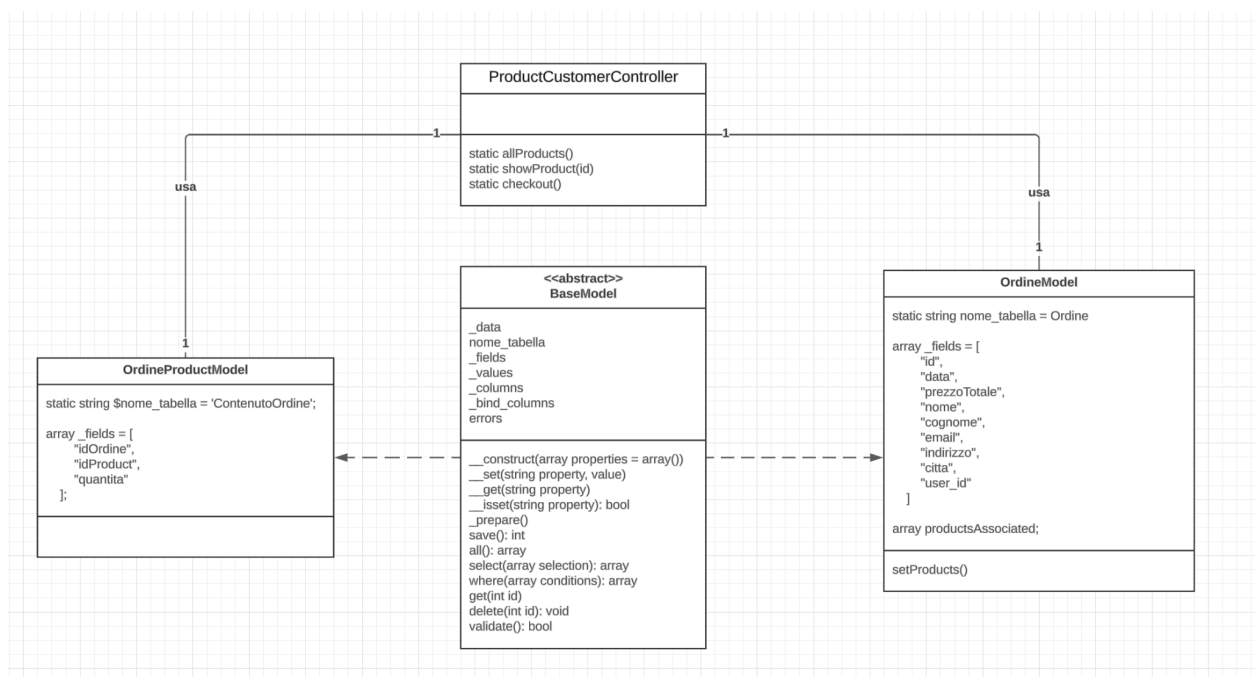
Il codice gestisce la logica per il processo di registrazione di un nuovo utente in un'applicazione PHP. La funzione `checkSignupErrors` viene utilizzata per controllare vari errori durante la registrazione. Vengono verificati tre errori principali: se la password e la conferma della password non corrispondono, se l'email è già utilizzata da un altro utente nel database, e se la password è troppo corta (meno di 6 caratteri).

La funzione `signup` gestisce l'effettiva registrazione. In primo luogo, chiama `checkSignupErrors` per ottenere eventuali errori. Se non ci sono errori, viene

creato un nuovo oggetto UserModel con i dati inviati dal modulo, tra cui email, password (che viene criptata con password_hash), nome, cognome, indirizzo, città e il ruolo di venditore. L'oggetto UserModel viene poi salvato nel database con il metodo save. Se l'utente viene salvato con successo, viene effettuato un reindirizzamento alla pagina di login.

In caso di errore durante il salvataggio (ad esempio, un errore di connessione al database), viene catturata un'eccezione e stampato l'errore.

Implementazione del checkout al pagamento



Il codice gestisce il processo di checkout per un ordine, inclusi il salvataggio dell'ordine e dei prodotti associati. Quando una richiesta POST viene ricevuta, la funzione checkout crea un nuovo oggetto OrdineModel con i dati relativi all'ordine, come la data corrente, il prezzo totale, le informazioni personali dell'utente (nome, cognome, email, indirizzo e città) e l'ID dell'utente (se loggato, ottenuto dalla sessione). L'oggetto OrdineModel viene poi salvato nel database tramite il metodo save.

Se il salvataggio dell'ordine ha successo, la funzione prosegue recuperando i dettagli del carrello dell'utente tramite la funzione `getCartFromCookie()`. Per ogni prodotto nel carrello, viene creato un oggetto `OrdineProductModel` che rappresenta il legame tra l'ordine e il prodotto, associando l'ID dell'ordine con l'ID del prodotto e la quantità acquistata. Ogni legame ordine-prodotto viene quindi salvato nel database. Se si verifica un errore durante il salvataggio del legame, viene catturata un'eccezione, viene mostrato un messaggio di errore e il codice dell'errore.

Dopo aver salvato tutti i prodotti associati all'ordine, il carrello viene svuotato tramite la funzione `clearCartCookie()` e l'utente viene reindirizzato alla pagina di ringraziamento `thankyou_page`.

Se si verifica un errore durante il salvataggio dell'ordine, viene catturata un'eccezione e viene mostrato un messaggio di errore.

Lettura delle notifiche

Abbiamo implementato una funzionalità che permette all'utente di segnare una notifica come "letta" senza dover ricaricare l'intera pagina. Utilizzando AJAX (Asynchronous JavaScript and XML), abbiamo inviato una richiesta POST al server con i dati della notifica selezionata. Il server ha elaborato questi dati, aggiornando lo stato della notifica nel database, e ha risposto con un messaggio di conferma che è stato gestito direttamente dalla nostra applicazione front-end.

Per cosa l'abbiamo sfruttato

1. Aggiornamento asincrono dei dati: Abbiamo inviato al server l'ID della notifica da aggiornare e lo stato "letta", evitando il caricamento completo della pagina.
2. Interfaccia utente reattiva: Dopo aver ricevuto la risposta dal server, abbiamo aggiornato l'interfaccia utente per riflettere immediatamente il nuovo stato della notifica (rimuovendo il bottone o cambiando lo stile visivo).
3. Integrazione dinamica: Grazie ad AJAX, abbiamo integrato il front-end e il back-end in modo che comunichino in tempo reale senza interrompere l'esperienza dell'utente.

1

Total EUR		5000.00€
Orologio Lusso A1	pz: 1	5000.00€

Seleziona il metodo di pagamento

- Nome completo _____ Numero della carta di credito _____

il nome e cognome riportati sulla carta di credito.

Catalogo Prodotti

Nome orologio

RB 2-504

Marca

Rolex Daytona

Condizioni

Nuovo

0 Costo Minimo €

0 Costo Massimo €

Ricerca



Orologio Lusso A1

Rolex Orologio da p...

Informazioni

Aggiungi al carrello



Orologio
Elegante B2

Omega Orologio ele...

Informazioni

Aggiungi al carrello



Orologio Sportivo C3

Tag Heuer Orologio ...

Informazioni

Aggiungi al carrello



Orologio Luxury
D4

Patek Philippe Orolo...

Informazioni

Aggiungi al carrello

Scegli il tuo orologio



Benvenuto nel tuo negozio di orologi

Email address

Nome

Cognome

Indirizzo

Città

Password

Superiore a 6 caratteri

Ripeti Password

☐ Sei un venditore?

Registrati

[Hai già un account? Entra](#)