**PROBLEM STATEMENT**

**Problem**

Nel mercato, al giorno d’oggi, è diventata di vitale importanza per un’attività commerciale ampliare il proprio bacino di utenza attraverso l’uso di piattaforme online.

Il successo di Zalando o di altri siti di e-commerce specializzati, sottolinea l’esigenza di una nicchia di utenti ad avere un mercato frammentato, a discapito dei grandi colossi general-purpose.

Nello specifico, il sistema vuole essere un supporto per rivenditori di occhiali da sole, fornendo loro tutte le funzionalità per la compravendita dei prodotti.

Pertanto, il sito di commercio elettronico, quattrocchi.it, mira ad essere un software semplice e intuitivo, con poche essenziali funzionalità ma efficaci nell’attuazione di transizioni.

**Objectives**

Gli obiettivi di Quattrocchi.it sono quelli di:

* Fornire un’interfaccia semplice, ma completa, che consenta all’utente di trovare facilmente il prodotto di cui ha bisogno nel minor tempo possibile;
* Semplificare il ruolo degli amministratori per la gestione del magazzino e la visualizzazione delle vendite.

**Example scenarios**

**Ricerca di un prodotto:** Beatrice ha intenzione di comprare un paio di occhiali RayBan con le lenti blu specchiate al suo fidanzato Luca. Si collega al sito e nella barra di ricerca in alto scrive “RayBan”. Beatrice preme invio ed una pagina contenente i primi risultati viene mostrata. Ora Beatrice cerca di raffinare la ricerca per trovare esattamente l’articolo che vuole, quindi nella barra di sinistra inserisce nel campo “Colore della lente” il colore “blu” e attiva la spunta “Lente specchiata”. Non appena modifica questi parametri vengono modificati la lista di prodotti si aggiorna e nel nuovo elenco Beatrice trova il primo occhiale con nome “RayBan da sole Mod. XN12”, per avere maggiori informazioni sull’articolo clicca sul nome e viene mostrata una pagina con altre foto e caratteristiche.

**Aggiunta di un prodotto nel carrello e acquisto:** Beatrice decide di acquistare questi occhiali. Nella pagina descrittiva dell’occhiale in cui si trova al termine della ricerca clicca su “Aggiungi al carrello” ed il numero affianco al carrello nella barra superiore del sito sul lato destro passa da 0 ad 1. Ora Beatrice decide di completare l’acquisto quindi dopo aver verificato di aver effettuato il login o la registrazione clicca sul pulsante con il simbolo del carrello in alto a destra. Nella pagina che si carica Beatrice può vedere l’occhiale che ha aggiunto con la relativa quantità, nel suo caso “1”. Per procedere Beatrice clicca sul pulsante “Procedi con l’ordine” in basso a destra. Una nuova pagina viene mostrata che chiede a Beatrice di scegliere un metodo di pagamento o di inserirne uno nuovo, Beatrice seleziona un nuovo metodo di pagamento e le viene chiesto d’inserire i dati della carta, scrive “Beatrice Rossi” nel campo intestatario, “1111222233334444” nel campo del numero della carta, “12/2020” nel campo della data di scadenza e “123” nel campo cvv. Ricontrolla le informazioni e preme su “Procedi con l’ordine in basso a destra”. La nuova pagina caricata le chiede di inserire un indirizzo di spedizione o di sceglierne uno di quelli salvati in precedenza. Beatrice trova il suo indirizzo di casa nell’elenco degli indirizzi salvati in precedenza e lo seleziona, poi clicca su “Termina ordine” in basso a destra per completare l’ordine. Beatrice ritorna alla pagina principale del sito e le viene mostrata una notifica al centro dello schermo che la informa dell’ordine sottomesso con successo.

**Aggiornare quantità di un prodotto nel catalogo:** L’amministratore Marco è intenzionato ad aggiornare la quantità dell’occhiale “Lindeberg titanium CA11” mostrata dal sistema. In primis Marco si assicura di aver effettuato la login con il suo account di amministratore. Poi esegue la ricerca inserendo il nome dell’articolo e premendo invio. Il primo risultato è esattamente l’articolo che stava cercando. Con un clic sull’articolo mostrato nella pagina della ricerca viene mostrata la pagina dell’articolo. Marco individua il campo “Quantità” sul lato destro della pagina affianco le informazioni tecniche, lo modifica scrivendo “10” e clicca sul pulsante “Salva” posto al di sotto del campo. Una notifica viene mostrata nella pagina per avvisare marco del successo dell’operazione che a questo punto termina.

**Requirements**

L’utente che ha intenzione di utilizzare il sistema deve essere in grado di:

* Connettersi al sito
* Registrarsi o effettuare il login
* Visualizzare il proprio profilo e modificarlo
* Aggiungere dati di pagamento e di spedizione associati al proprio profilo
* Ricercare prodotti all’interno del sistema
* Aggiungere prodotti al carrello
* Visualizzare il carrello
* Eseguire il checkout e completare un ordine
* Visualizzare lo storico ordini

L’amministratore del sito deve essere in grado di:

* Connettersi al sito
* Eseguire il login
* Ricercare i prodotti all’interno del catalogo
* Aggiungere nuovi prodotti
* Aggiornare le specifiche di ogni prodotto
* Visualizzare lo storico ordini globale
* **Semplice trovare articoli.** L’interfaccia deve essere minimale e l’utente deve intuire le funzioni di ogni campo per la ricerca senza utilizzare manuali utente.
* **Trovare sempre il giusto articolo.** Deve essere possibile specificare ricerche complesse che consentano una ricerca all’interno del magazzino efficiente che restituisce risultati appropriati.
* **Sito responsive.** L’interfaccia dev’essere chiara ed adattabile a qualsiasi dispositivo che ne faccia uso.
* **Tempo di caricamento pagina inferiore a 5 secondi.** Ogni caricamento di pagina deve avvenire in tempi ragionevoli per l’utente.
* **Veloce procedura di checkout.** Al cliente sarà richiesto il minimo indispensabile (indirizzo, metodo di pagamento e recapito telefonico) per il completamento di un ordine.

**Target enviroment**

* Quattrocchi.it deve essere scritto usando le java servlet, jsp e javascript.

Il sito deve funzionare su Mozilla Firefox, Google Chrome, Safari, Microsoft Edge.

**Deliverables**

* Problem Statement
* Requirement Analysis Document
* System Design Document
* Object Design Document
* Test Plan
* Test Case Specification
* Test Execution Report

**Deadlines**

1. Proposta di progetto e kick-off meeting: 3 ottobre 2017
2. Problem statement: 12 ottobre 2017
3. Requisiti e casi d’uso: 24 ottobre 2017
4. Requirements Analysis Document: 7 novembre 2017
5. System Design Document: 28 novembre 2017
6. Specifica delle interfacce dei moduli del sottosistema da implementare: 19 dicembre 2017
7. Piano di test di sistema e specifica dei casi di test per il sottosistema da implementare: 19 dicembre 2017
8. Ulteriore documentazione e codice: Discussione del progetto

**Acceptance Criteria**

Il sistema creato è accettato se:

* Il sito è accessibile da Mozilla Firefox, Google Chrome, Safari e Microsoft Edge.