|  |
| --- |
| **Università degli Studi di Salerno Corso di Ingegneria del Software** |

**The Spectacles**  
**System Design Document**  
**Versione 0.8**



Data: 10/02/2022

**Coordinatore del progetto:**

|  |  |
| --- | --- |
| Nome | Matricola |
|  |  |
|  |  |

**Partecipanti:**

|  |  |
| --- | --- |
| Nome | Matricola |
| Roberto Piscopo | 0512109906 |
| Alessandro Satta | 0512110929 |
| Mario Ranieri | 0512110017 |
| Luca Di Meglio | 0512110767 |
|  |  |
|  |  |

|  |  |
| --- | --- |
| **Scritto da:** | Alessandro Satta, Roberto Piscopo, Mario Ranieri, Luca Di Meglio |

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Descrizione | Autore |
| 30/11/2022 | 0.1 | Inizio documentazione del progetto | Alessandro Satta e Luca Di Meglio |
| 14/12/2022 | 0.2 | Aggiunte informazioni riguardo ai punti: 1.4,1.5,1.6,2,3,3.1,3.2,3.3,3.3.1,3.4,3.4.1,3.4.2 | Alessandro Satta |
| 15/12/2022 | 0.3 | Aggiunte informazioni riguardanti ai punti: 3.5 | Alessandro Satta |
| 21/12/2022 | 0.4 | Aggiunta informazioni 3.2 | Alessandro Satta, Mario Ranieri |
| 27/12/2022 | 0.5 | Aggiunta informazioni riguardo i punti: 3.6, 3.7 | Luca Di Meglio |
| 13/01/2023 | 0.6 | Aggiunta parziale informazioni punto 4 | Luca Di Meglio |
| 10/02/2022 | 0.7 | Modificato Deployment Diagrams e tabella accessi | Roberto Piscopo |
| 10/02/2023 | 0.8 | Modifiche ai punti 2, 3, 4 | Alessandro Satta |

Indice

1. INTRODUZIONE

1.1 Scopo Sistema

1.2 Ambito del Sistema

1.3 Obiettivi e criteri di successo del progetto

1.3.1 Criteri di performance

1.3.2 Criteri di Affidabilità

1.3.3 Criteri di costi

1.3.4 Criteri di manutenzione

1.4 Design trade-offs

1.4.1 Tempo di rilascio VS Funzionalità

1.3.1 Prestazioni VS Affidabilità

1.5 Definizioni, acronimi e abbreviazioni

1.6 Riferimenti .

1.7 Panoramica

2. Architettura di sistemi simili

3. Architettura sistema proposto

3.1 Panoramica

3.2 Decomposizione Sottosistemi

3.3 mappatura Hardware/Software

3.3.1 Deployment Diagram

3.4 gestione dei dati persistenti

3.4.1 Schema concettuale

3.4.2 Schema logico

3.5 Controllo degli accessi e sicurezza

3.6 Controllo flusso globale del software

3.7 Condizioni Limite

3.7.1 Server

3.7.2 Client

4. Servizi dei Sottosistemi

1. INTRODUZIONE

## Scopo Sistema

The Spectacles è un e-commerce finalizzato alla vendita di occhiali da sole e da vista. L’obiettivo è quello di soddisfare i gusti più vari, permettendo di adattare qualsiasi montatura in base ad ogni esigenza.

## Ambito del Sistema

Il sistema nasce per entrare a far parte del mondo degli e-commerce in ambito wearable.

Il sistema deve supportare:

* Acquisto prodotto
* Lasciare una recensione
* Aggiunta al carrello dei prodotti
* Visualizzare tipologia e categorie prodotti
* Pagare attraverso carta o alla consegna
* Visualizzare ordini

## Obiettivi e criteri di successo del Progetto (Design Goals)

***1.3.1 Criteri di performance***

* *Tempo di risposta:* Il sito web deve fornire un tempo di risposta basso da poter permettere una fluida navigazione a tutti gli utenti.
* *Memoria:* La memoria fornita dal DB dovrà essere scalabile per accomodare influssi di nuovi utenti e creazione di un numero qualsiasi di nuove entry.

***1.3.2 Criteri di affidabilità***

* *Disponibilità:* Il sito web dovrà essere online 24/7.
* *Sicurezza e privacy:* Il sito richiederà e-mail e password per la registrazione e successive autenticazioni.
* *Robustezza:* Input non validi verranno prontamente individuati e bloccati dal sistema, e l’utente verrà notificato un messaggio di errore.

***1.3.3 Criteri di costi***

* *Sviluppo:* Il costo complessivo di sviluppo stimato è di circa 120 ore (30 ore per ogni membro del progetto).

***1.3.4 Criteri di manutenzione***

* *Estendibilità:* Sarà possibile aggiungere nuove funzionalità al sistema, in base alle esigenze dell’utenza e del mercato dell’occhialeria.
* *Portabilità:* La portabilità sarà garantita in quanto l’interazione con il sistema avverrà tramite un browser web.

## Design trade-offs

## Tempo di rilascio VS Funzionalità

Al fine di avere un impatto positivo con i primi clienti è necessario che il sistema implementi tutte le funzionalità necessarie, per cui si preferisce mettere in secondo piano i tempi di rilascio per avere un e-commerce quanto più possibile funzionale e privo di bug.

## Prestazioni VS Affidabilità

Elenco obiettivi:

* Dare la possibilità di consultare il sito a tutti gli utenti (ospiti compresi)
* Pagamenti rapidi e sicuri
* Selezionare i marchi con più gradimento
* Criteri di successo

## Definizioni, acronimi e abbreviazioni

**SDD**: System Design Document.

**DB**: Database.

**Admin:** Soggetto che si occupa di eliminare/aggiungere/modificare prodotti all’interno della piattaforma, può conoscere gli ordini effettuati dagli utenti e conoscere gli ordini effettuati in un intervallo di tempo.

**Utente:** Un utilizzatore della piattaforma che abbia fatto il login.

**Ospite:** Un qualsiasi utilizzatore della piattaforma che non ha effettuato il login.

**Web browser:** Software per l’applicazione, la presentazione e la navigazione di risorse sul web.

**Web server:** Software che, in esecuzione sul server, è in grado di gestire le richieste di trasferimento di pagine web di un client, tipicamente web browser.

## Riferimenti

* Bernd Bruegge & Allen H. Dutoit, *Object-Oriented Software Engineering: Using UML, Patterns and Java*, (2nd edition),Prentice-Hall, 2003.
* Ian Sommerville, Software Engineering, Addison Wesely.

## Panoramica

Il seguente documento SDD è diviso nelle seguenti sezioni:

* Al secondo punto del documento verrà presentato il sistema corrente.
* Al terzo punto verrà presentata l’architettura del sistema proposto in cui gestiremo la decomposizione in sottosistemi, il mapping hardware/software, i dati persistenti, il controllo degli accessi e sicurezza, il controllo del flusso globale del sistema, le condizioni limite.
* Al quarto punto verranno presentati i servizi dei sottosistemi.

1. ARCHITETTURA DI SISTEMI SIMILI

Per impossibilità di verificare le architetture simili dei competitor è stato preferito optare autonomamente per un architettura three-layer.

1. SISTEMA PROPOSTO

## 3.1 Panoramica

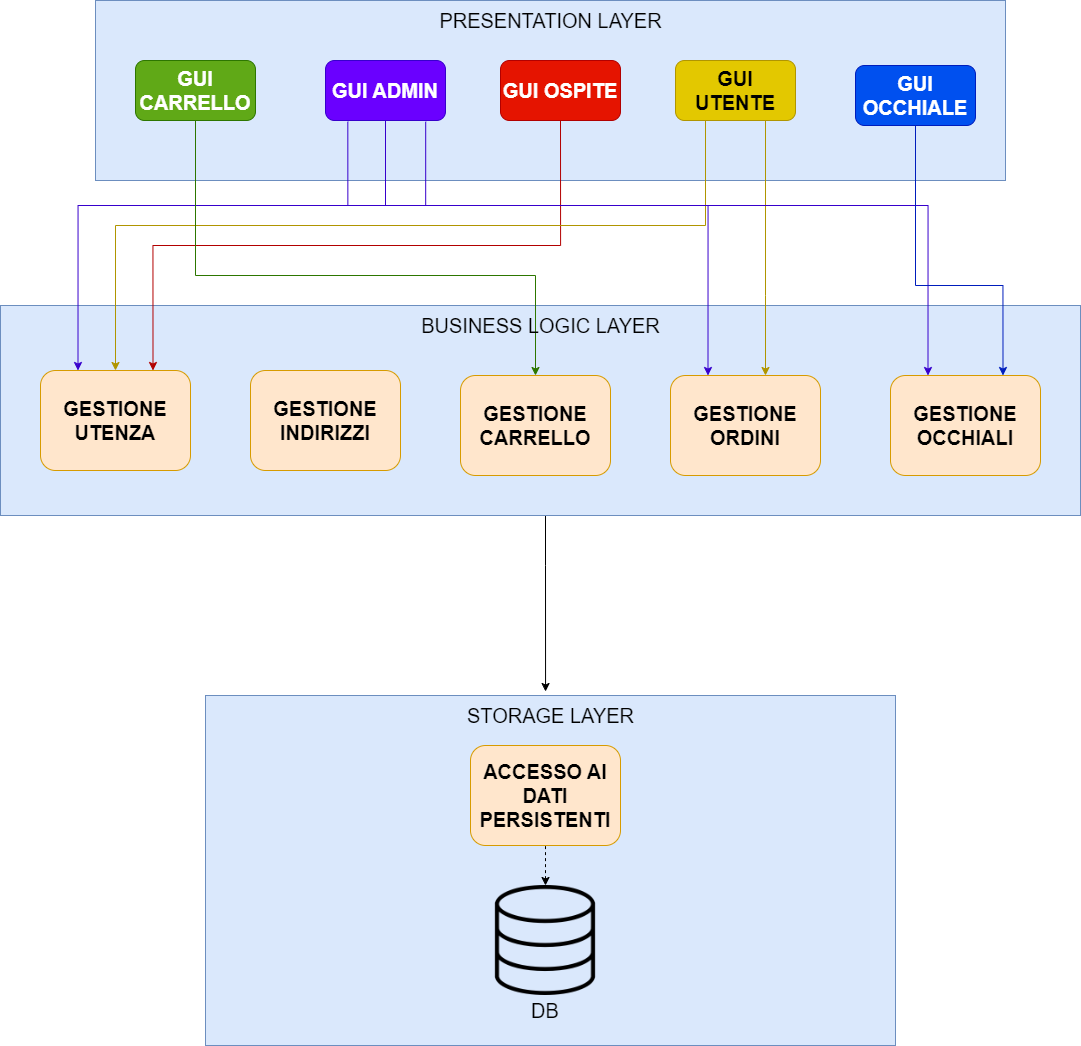
Il sistema che andiamo a proporre è un e-commerce specializzato nella vendita di occhiali, il suo obiettivo, è di fornire una piattaforma che permetta agli utenti di acquistare occhiali.

Il sistema prevede funzioni limitate per utenti non registrati, funzionalità più avanzate per utenti registrati e privilegi di gestione del catalogo occhiali e possibilità di visualizzare gli ordini in un determinato range di date oppure per cliente.

Al fine di ridurre l’accoppiamento fra presentazione dei dati e logica di business, il sistema verrà diviso in client e server: il client gestirà la parte di presentazione e della logica di controllo connessa all’interfaccia grafica, il server invece gestirà la logica relativa ai dati e la gestione di questi ultimi tramite un database (anch’esso salvato su server). La divisione delle funzionalità del sistema sarà in tre layer logici: presentation, business e data storage.

## Decomposiozione in Sottosistemi

## 



Il sistema si compone di tre layer:

• **Il Presentation Layer** si occupa di presentare le informazioni ai diversi utenti e consente agli stessi di interagire con il sistema in modo semplice e intuitivo.

• **Il Business Logic Layer** definisce la logica applicativa.

•**Data Layer** si occupa della gestione dei dati persistenti.

Il layer di presentazione presenta cinque sottosistemi:

• **Gui Utente:** rappresenta l’insieme delle pagine web con le quali l’utente “Utente” interagisce per poter usufruire dell’e-commerce.

• **Gui Ospite:** rappresenta l’insieme delle pagine web con le quali un Ospite può interagire.

• **Gui Admin:** rappresenta l’insieme delle pagine web con le quali l’Admin del sistema interagisce per poter effettuare operazioni che riguardano la gestione degli occhiali o la ricerca degli ordini effettuati dagli utenti del sistema.

• **Gui Carrello:** rappresenta la pagina del carrello.

Il layer di business logic è diviso in cinque sottosistemi:

• **Gestione Indirizzi:** fornisce le funzionalità per inserire un nuovo indirizzo

• **Gestione Occhiali:** fornisce le funzionalità per inserire o cancellare o modificare un occhiale nel database.

• **Gestione Utenza:** fornisce le funzionalità per effettuare l’autenticazione e, in generale, ciò che concerne la definizione e la gestione dei permessi per poter visualizzare determinate aree e funzionalità del sistema.

• **Gestione Carrello:** fornisce le funzionalità per modificare la quantità di un prodotto nel carrello o eliminarlo dallo stesso.

• **Gestione Ordini:** fornisce le funzionalità per acquistare uno o più prodotti , selezionare l’indirizzo di spedizione desiderato tra quelli creati precedentemente dall’utente, inserire i dati della carta di credito e creare un Ordine e fornisce le funzionalità per visualizzare gli ordini effettuati

Il layer di persistenza è costituito da un unico sottosistema:

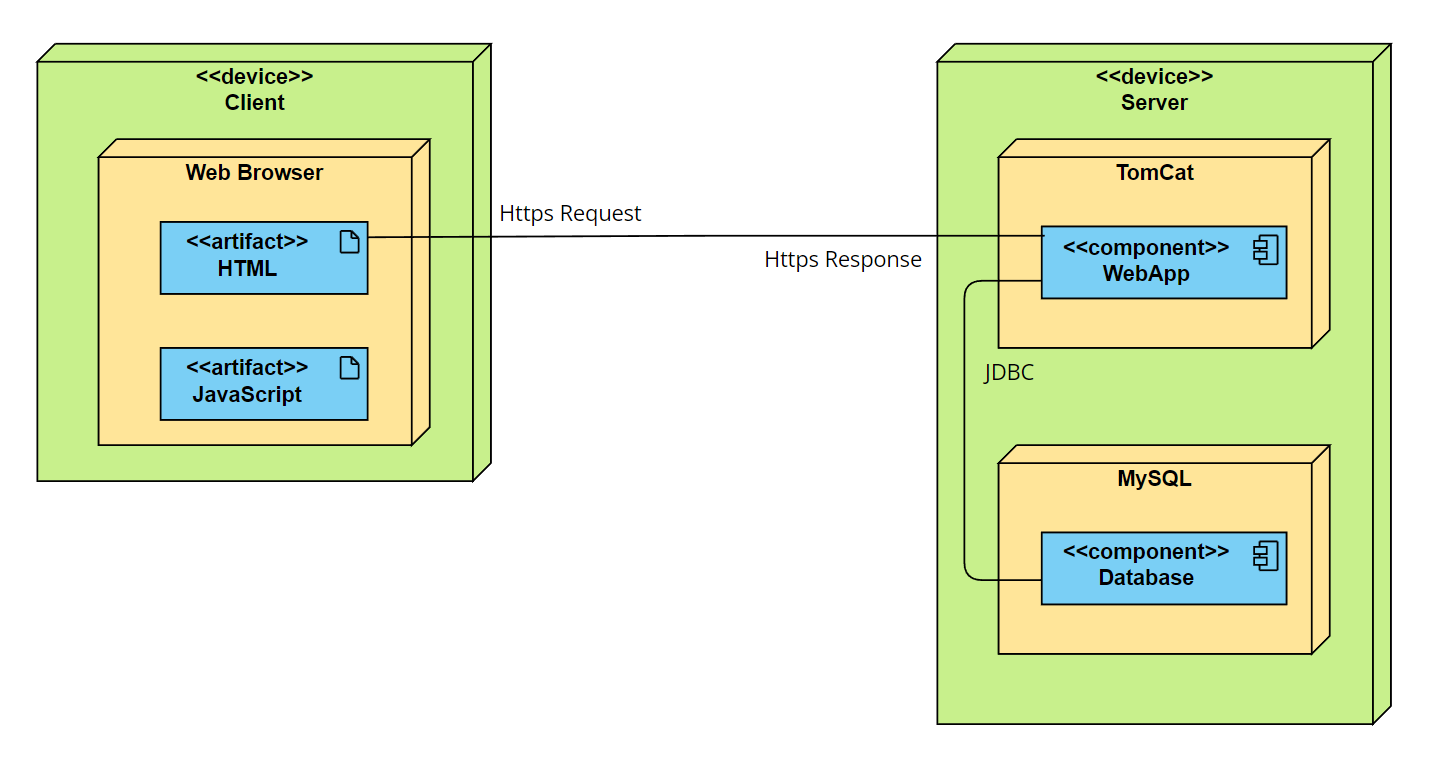
**Accesso ai dati persistenti:** si occupa di leggere e modificare i dati memorizzati sul Database.

## Mapping Hardware/Software

Per il sistema, basato su un’architettura three-layer, sono state scelti questi tipi di configurazioni:

* Per il Presentation Layer usiamo le JSP, HTML, CSS, JavaScript.
* Per il Business Logic Layer usiamo Apache Tomcat e Java.
* Per il Data Layer usiamo MySQL con JDBC.
* Come Protocollo di comunicazione usiamo HTTP, TCP/IP.

## Deployment Diagram



## Gestione dei dati persistenti

È stato preferito l’utilizzo di un DBMS relazionale, in specifico MySQL per la gestione dei dati.

In più i DBMS garantiscono una gestione concorrente, l’accesso al DBMS è trasparente allo sviluppatore ed anche al cliente.

* + 1. ***Schema concettuale***

Per lo schema concettuale si rimanda al Class Diagram presente nel documento RAD\_THE\_SPECTACLES.

* + 1. ***Schema logico***

***Tabella utente***

|  |  |  |
| --- | --- | --- |
| ***ATTRIBUTO*** | ***TIPO*** | ***VINCOLI*** |
| nome | VARCHAR(20) | Not null |
| cognome | VARCHAR(20) | Not null |
| data di nascita | DATE | Not null |
| e-mail | VARCHAR(45) | Primary key,  Not null |
| password | VARCHAR(50) | Not null |
| ruolo | INT | Not null |

La tabella utente contiene le informazioni riguardanti gli utenti presenti nel sistema. Un utente è identificato univocamente all’interno del sistema dalla sua e-mail (non possono esistere due utenti con la stessa username). Un utente nel database può essere sia un utente registrato al sistema, un gestore oppure un amministratore.

***Tabella ordine***

|  |  |  |
| --- | --- | --- |
| ***ATTRIBUTO*** | ***TIPO*** | ***VINCOLI*** |
| idOrdine | VARCHAR(36) | Primary key  ,Not null |
| data | TIMESTAMP | Not null |
| e-mail | VARCHAR(36) | Not null |
| stato | TEXT | Not null |

La tabella ordine contiene tutte le informazioni riguardanti l’ordine, quest’ultimo viene identificato univocamente tramite idOrdine, viene memorizzato il timestamp, cioè la data

e l’orario in cui si effettua l’odine, l’e-mail per associare un utente all’ordine effettuato e

lo stato in cui si trova l’ordine (in lavorazione, spedito).

***Tabella occhiale\_ordine***

|  |  |  |
| --- | --- | --- |
| ***ATTRIBUTO*** | ***TIPO*** | ***VINCOLI*** |
| id | INT | Primary key  ,Not null,  Auto increment |
| idOcchiale | VARCHAR(45) | Not null |
| idOrdine | VARCHAR(45) | Not null |
| prezzo\_reale | INT | Not null |
| quantità | INT | Not null |
| iva | FLOAT | Not null |

La tabella Occhiale\_Ordine è generata da una relazione N-M tra la tabella ordine e la tabella occhiale. Si indentifica univocamente tramite id.

***Tabella occhiale***

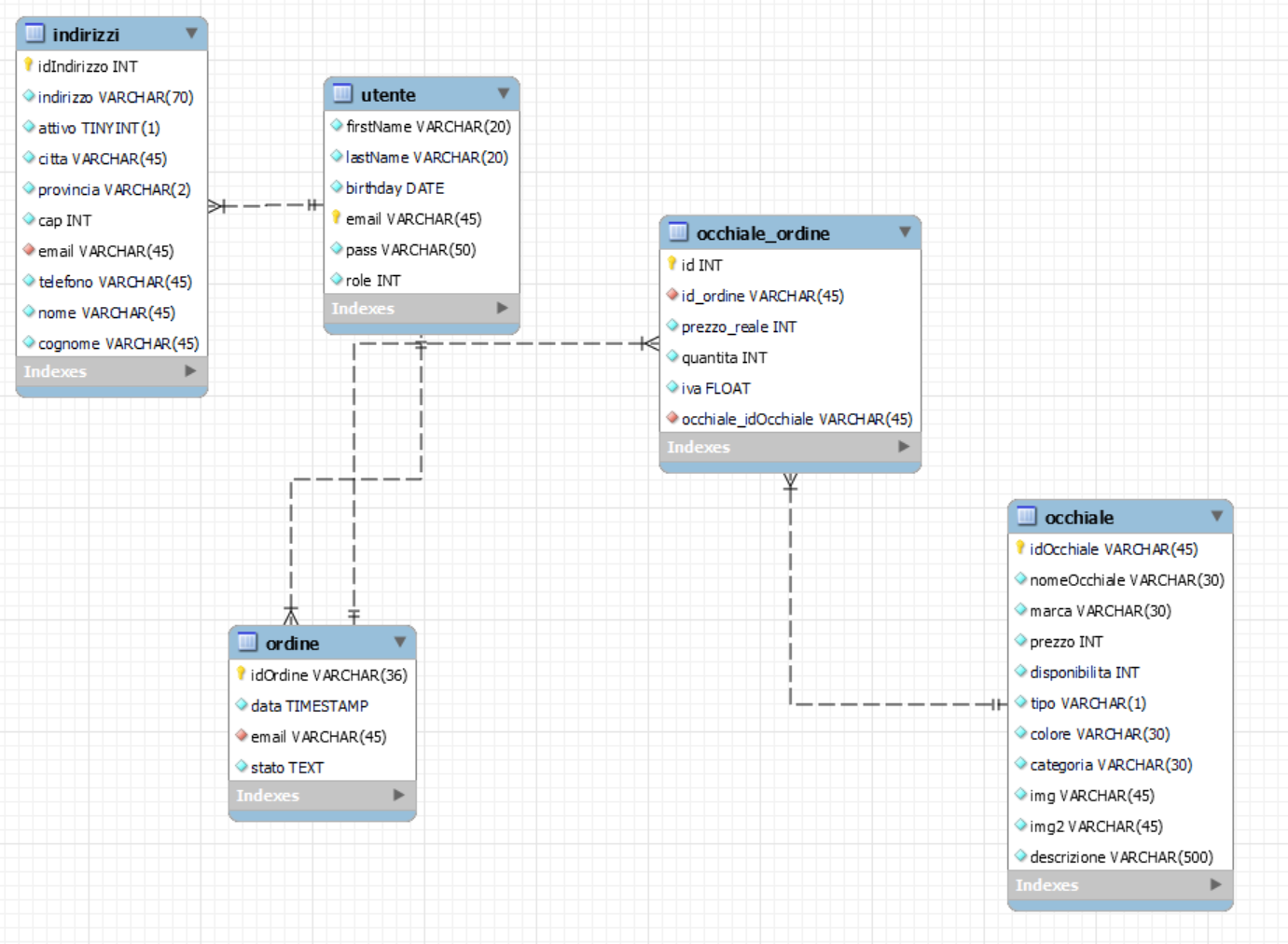
|  |  |  |
| --- | --- | --- |
| ***ATTRIBUTO*** | ***TIPO*** | ***VINCOLI*** |
| idOcchiale | VARCHAR(45) | Primary key  ,Not null |
| nomeOcchiale | VARCHAR(30) | Not null |
| marca | VARCHAR(30) | Not null |
| prezzo | INT | Not null |
| disponibilità | INT | Not null |
| tipo | VARCHAR(1) | Not null |
| colore | VARCHAR(30) | Not null |
| categoria | VARCHAR(30) | Not null |
| img | VARCHAR(45) | Not null |
| img | VARCHAR(45) | Not null |
| descrizione | VARCHAR(500) | Not null |

La tabella occhiale contiene tutte le informazioni riguardanti gli occhiali, questi vengono identificati univocamente tramite un idOcchiale, inoltre ci sono tutti gli attributi che lo caratterizzano.

***Tabella indirizzi***

|  |  |  |
| --- | --- | --- |
| ***ATTRIBUTO*** | ***TIPO*** | ***VINCOLI*** |
| idIndirizzi | INT | Primary key  ,Not null,  Auto increment |
| Indirizzo | VARCHAR(45) | Not null |
| Attivo | TINYINT(1) | Not null |
| città | VARCHAR(45) | Not null |
| Provicia | VARCHAR(45) | Not null |
| Cap | INT | Not null |
| E-mail | VARCHAR(45) | Not null |
| Telefono | VARCHAR(45) | Not null |

La tabella indirizzi contiene tutte le informazioni riguardanti l’indirizzo, quest’ultimo è associato ad un utente, il quale può avete più indirizzi associati. Un indirizzo viene identificato univocamente tramite un idIndirizzo.



## Controllo degli accessi e sicurezza

Nel sistema si hanno tre tipologie diverse di utenti: l’utente non registrato, l’utente registrato, l’admin.

Ogni utente può accedere a diverse funzionalità del sistema e ha diritti di accesso differenti, regolamentati sulla base delle varie tipologie di utenza.

Per tenere traccia dei diritti di accesso usufruiamo della seguente tabella di controllo degli accessi:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Oggetti**  **Attori** | **GESTIONE UTENTE** | **GESTIONE ORDINE** | **GESTIONE**  **OCCHIALE** | **GESTIONEACQUISTO** | **GESTIONE CARRELLO** |
| **OSPITE** | * Effettuare registrazione * Accedere alla piattaforma |  | * Visualizza catalogo |  | * Aggiungi Prodotto * Modifica Quantità * Rimuovi Prodotto |
| **UTENTE** | * Logout * Visualizza dati personali | * Visualizza ordini effettuati | * Visualizza catalogo | * Crea indirizzi * Scegli indirizzo * Aggiungi dati carta di credito | * Aggiungi Prodotto * Modifica Quantità * Rimuovi Prodotto |
| **ADMIN** | * Logout | * Visualizza ordini effettuati da un cliente * Visualizza ordini effettuati in un intervallo di tempo | * Aggiungere nuovi occhiali * Modificare occhiali esistenti * Eliminare occhiali esistenti |  |  |

## Controllo flusso globale del software

Il flusso è Event-Driven in cui ad ogni azione dell’utente è associato un evento, questo perché si tratta di un sistema web fortemente guidato dall’interazione dell’utente con le pagine web.

## Condizioni limite (Boundary conditions)

**3.7.1 SERVER**

Fase di Startup:

La fase di avvio del server Apache Tomcat dura circa cinque secondi, dopo di che viene avviato il server dove è contenuto il DBMS MySQL. Viene effettuata la connessione al DBMS tramite il driver JDBC. Successivamente viene avviata la JVM sul server e vengono letti i dati di configurazione del database. Infine, vengono caricati tutti i dati in memoria volatile. Il sistema è pronto all'uso.

Fase di Terminazione:

Il server può essere spento nel caso si verifichi un guasto o sia necessario un intervento di manutenzione. Una volta conclusa la riparazione del guasto o l’intervento di manutenzione il server viene riavviato normalmente.

Fallimento:

Il server può andare in crash quando viene sollevata un'eccezione che non è stata gestita.

**3.7.2 CLIENT**

Fase di Terminazione:

L’utente chiude la pagina del sistema interrompendo la comunicazione e quindi la sessione TCP.

## Scenario “Startup”

Un amministratore vuole avviare la piattaforma The Spectacles. Inizia avviando il server Apache Tomcat dopodiché configura il DBMS MySQL per essere raggiunto dal medesimo web server. A questo punto, carica su Tomcat il file .war della piattaforma The Spectacles. Dopo cinque secondi dall’avvio la piattaforma è operativa e raggiungibile dagli utenti.

## Scenario “Fallimento”

Il server mostra al client una schermata di errore.

1. SERVIZI DEI SOTTOSISTEMI

**Gestione utenza:**

|  |  |
| --- | --- |
| ***Servizio*** | ***Descrizione*** |
| Registrazione | il sottosistema permette di creare un nuovo account |
| Login | il sottosistema permette di effettuare l’autenticazione |
| Logout | il sottosistema permette di far disconnettere l’utente autenticato |
| Visualizzazione dati utente | il sottosistema permette di visualizzare i dati personali dell’utente |

**Gestione occhiali:**

|  |  |
| --- | --- |
| ***Servizio*** | ***Descrizione*** |
| Visualizzazione lista occhiali | il sottosistema permette di visualizzare tutti gli occhiali presenti nel catalogo |
| Visualizzazione brand occhiali | il sottosistema permette di visualizzare la lista degli occhiali di un certo brand |
| Visualizzazione per sesso occhiali | Il sottosistema permette di visualizzare la lista degli occhiali per un certo sesso |
| Visualizzazione per categoria occhiali | Il sottosistema permette di visualizzare la lista degli occhiali di una certa categoria(da vista, da sole, filtro luce blu) |
| Ricerca occhiali | Il sottosistema la ricerca di un’occhiale specifico |
| Aggiunta Nuovo occhiale | Il sottosistema permette di inserire un nuovo occhiale al catalogo |
| Modifica occhiale | Il sottosistema permette di modificare un occhiale del catalogo |
| Rimozione Occhiale | Il sottosistema permette di rimuovere un occhiale dal catalogo |

**Gestione carrello:**

|  |  |
| --- | --- |
| ***Servizio*** | ***Descrizione*** |
| Inserimento prodotto nel carrello | il sottosistema permette ad un utente di inserire un prodotto nel carrello |
| Modifica prodotto nel carrello | il sottosistema permette ad un utente di modificare la quantità di un prodotto già presente nel carrello |
| Eliminazione prodotto dal carrello | il sottosistema permette ad un utente di eliminare un prodotto dal carrello. |

**Gestione Ordini**

|  |  |
| --- | --- |
| ***Servizio*** | ***Descrizione*** |
| Visualizzazione Ordini | il sottosistema permette ad un utente di visualizzare i suoi ordini |
| Visualizzazione Ordini per Data | il sottosistema permette all’admin di visualizzare gli ordini entro un range di date |
| Visualizzazione Ordini Admin | il sottosistema permette all’admin di visualizzare gli ordini di un cliente |
| Selezione indirizzo di consegna | il sottosistema permette di selezionare un indirizzo di consegna |
| Inserimento dati carta di credito | il sottosistema permette ad un utente di inserire i dati della carta di credito |
| Acquistare un prodotto | il sottosistema permette ad un utente di finalizzare l’acquisto di uno o più prodotti |

**Gestione Indirizzo:**

|  |  |
| --- | --- |
| ***Servizio*** | ***Descrizione*** |
| Inserimento indirizzo | l sottosistema permette di aggiungere un indirizzo di consegna |
| Visualizzazione Indirizzi | il sottosistema permette di visualizzare gli indirizzi aggiunti |

**Accesso ai dati persistenti:**

|  |  |
| --- | --- |
| ***Servizio*** | ***Descrizione*** |
| Eliminazione occhiali | Il sottosistema permette l’eliminazione dal database degli occhiali individuandoli attraverso il loro ID. |
| Inserimento occhiali | il sottosistema permette di salvare sul database le informazioni che riguardano degli occhiali. |
| Inserimento indirizzo | il sottosistema permette di associare un nuovo indirizzo nel database ad un utente registrato identificandolo tramite e-mail. |
| Modifica occhiali | il sottosistema permette di modificare le informazioni riguardanti degli occhiali nel database identificandoli tramite il loro ID. |
| Eliminazione indirizzo | il sottosistema permette di eliminare un indirizzo associato ad un utente identificandolo tramite e-mail |
| Inserisci ordine | il sottosistema permette di associare un nuovo ordine ad un utente registrato identificandolo per e-mail. |
| Ottieni ordini | il sottosistema permette di reperire gli ordini di un utente indentificandolo tramite e-mail. |
| Ottieni occhiali | il sottosistema permette di ottenere le informazioni riguardanti gli occhiali identificandoli tramite ID |
| Ottieni informazioni utente | il sottosistema permette di ottenere le informazioni riguardanti l’utente identificandolo tramite email |