

# R&C STORE

*Muscles & Brains*

Alessandro Cecchetto – 1941039

Emanuele Roccia – 1967318

# Indice

1. Introduzione e descrizione dei dati
  - 1.1. Descrizione
  - 1.2. Tipi di utenti
  - 1.3. Casi d'uso
  - 1.4. Strutture dati
  - 1.5. Spiegazione elementi/attributi
2. Manuale utente
  - 2.1. Visitatori
  - 2.2. Clienti
  - 2.3. Gestori
  - 2.4. Amministratori
3. Programmazione
4. Elenco delle credenziali

# 1. Introduzione e descrizione dei dati

## 1.1 Descrizione

La tesina verte sulla gestione di un negozio online di rivendita prodotti per la palestra (proteine, barrette proteiche, ecc...) offrendo un catalogo multimarca.

E' presente un catalogo di prodotti, descritti con testo e immagini, che tutti gli utenti possono consultare, registrati e visitatori.

Il catalogo puo' essere visionato secondo diversi ordinamenti (costo, categoria e marca del prodotto) e si possono escludere i prodotti "out-of-stock" dalla lista.

### Categorie

Ad un prodotto viene associata una categoria specifica, ad esempio di default sono definite:

- Proteine in polvere
- Barrette proteiche
- Vitamine
- Abbigliamento
- Accessori

All'occorrenza il gestore può definire nuove categorie da una pagina apposita.

### Offerte

Un elemento del catalogo puo' prevedere un'offerta.

Le offerte possono essere sconti o bonus.

Uno sconto è una percentuale di riduzione del prezzo dell'articolo (es. 20%), mentre con il bonus si ha una quantità fissata che viene restituita al cliente sotto forma di crediti (es. +3 crediti).

Le offerte hanno un target: sono previste "per tutti" (es. per tutti gli utenti, le proteine sono scontate del 10%), oppure personalizzate per una categoria di cliente (es. per i clienti da più di 2 anni tutti gli articoli sono scontati del 10%).

Nel caso dell'offerta personalizzata alla singola categoria di cliente, viene applicata a tutti i prodotti in catalogo e in base ai seguenti criteri:

- clienti che hanno speso X crediti finora
- clienti che hanno speso X crediti da una certa data Y
- clienti che hanno una reputazione  $\geq X$
- clienti che sono con noi da X anni

Nel caso dello sconto sul prodotto in sé, nel catalogo, viene applicato a tutti i clienti se:

- è un prodotto particolare, scelto dal gestore al momento di definizione dell'offerta
- il prodotto è di una determinata categoria
- è presente in magazzino un'eccedenza del prodotto X (più di Y unità) [[spiegato successivamente](#)]

Uno sconto può essere aggiunto nel novero degli sconti, o eliminato, dal gestore.

## Magazzino

Il sistema tiene traccia delle quantità dei singoli prodotti disponibili in magazzino.

Con l'acquisto dei prodotti essa viene decrementata automaticamente. Se arriva a zero, il prodotto viene contrassegnato nella pagina del catalogo come "out-of-stock" e non è possibile il suo acquisto.

Inoltre, la quantità dei singoli prodotti può essere modificata manualmente dal gestore, ad esempio quando i prodotti arrivano man mano dai fornitori.

Questa è la quantità controllata dal sistema per verificare se un'offerta del tipo "eccedenza di prodotto" è applicabile o meno.

## Crediti

Un cliente acquisisce crediti da spendere facendo una richiesta all'admin, attraverso una funzionalità del sistema.

L'admin vaglia la richiesta, controllando se è stato corrisposto il pagamento in modo "offline", cioè in modo separato da questa applicazione, disponendo di un bottone per accettarla o rifiutarla.

Una volta accettata, la quantità verrà assegnata all'utente.

## Recensioni, domande e risposte

Il sistema permette di scrivere, riguardo ai prodotti in catalogo, recensioni, fare domande, dare risposte.

I clienti possono giudicare i post suddetti, assegnando loro un valore di "supporto" ("accordo", da 1 a 3) e uno di "utilità" ("è stato utile da leggere", da 1 a 5).

## FAQ

Esiste una pagina di FAQ, gestita dall'admin.

Una domanda postata da clienti, con una relativa risposta, può essere "elevata" a far parte delle FAQ.

## Reputazione

In base ai giudizi sui propri contributi, un cliente ha una reputazione.

Ai fini del calcolo della reputazione pesano i risultati sia dei contributi relativi ad acquisti effettivamente fatti dal cliente, sia quelli relativi alle altre offerte: ma i primi pesano di più.

La reputazione viene calcolata usando pesi diversi per i giudizi dati da utenti diversi: il gestore pesa più dei clienti; un cliente pesa in base alla propria reputazione.

## Varie

Il profilo di un cliente contiene dati anagrafici, compresi indirizzo, telefono, codice fiscale, email.

Per un acquisto, l'indirizzo di consegna ha come default quello del profilo, ma può essere modificato prima di finalizzare l'acquisto.

## 1.2 Tipi di utenti

### Utente visitatore

- può accedere al catalogo
- può accedere a pagine informative (contatti, dove siamo, ...)
- può visionare le FAQ e domande e risposte dei singoli prodotti
- può effettuare la registrazione e il login (in caso, dopo l'accesso, diventa cliente)

### Utente registrato (cliente)

- come per visitatore, tranne registrazione
- può accedere al profilo, per vedere il suo carrello, il suo storico degli acquisti, i propri crediti, la propria reputazione, gestire i propri dati anagrafici e di autenticazione
- può effettuare un ordine (con un meccanismo di carrello della spesa)
- può scrivere recensioni, domande, risposte sui prodotti in catalogo
- può giudicare le recensioni, domande, risposte di altri clienti; il giudizio non può essere modificato successivamente
- chiede l'assegnazione di crediti (specificando quanti) tramite l'apposito form

### Utente gestore

- può aggiungere/eliminare/modificare prodotti in catalogo e le quantità disponibili in magazzino
- può definire le offerte (sconti e bonus)
- può consultare il profilo del cliente (reputazione, storico degli acquisti)
- può giudicare ed eliminare i contributi dei clienti
- può rispondere a domande in modo "definitivo"
- può aggiungere una domanda, con relativa risposta, alle FAQ
- può elevare una domanda postata da clienti, aggiungendola alle FAQ, insieme alla risposta migliore oppure alla risposta "definitiva"

### Utente amministratore

- può vedere e modificare i dati anagrafici, username, password degli utenti
- può disattivare (bannare) e attivare utenti
- può accettare o rifiutare le richieste di crediti
- può aggiungere una domanda, con relativa risposta, alle FAQ (come il gestore)
- eleva una domanda postata da clienti, aggiungendola alle FAQ, insieme alla risposta migliore o "definitiva" (come il gestore)

I gestori e amministratori non sono anche clienti (es. non possono acquistare prodotti), e possiedono pagine separate per le loro operazioni.

Alcune operazioni sono comuni tra gestore e amministratore (es. gestione delle FAQ come scritto sopra), ma in generale un gestore può effettuare delle operazioni che un amministratore non può fare (es. gestione offerte).

## 1.3 Casi d'uso

### 1.3.1 Acquisto

Il cliente apre la pagina con il catalogo dei prodotti. Si trova una griglia con, per ogni prodotto, l'immagine, il nome, il prezzo ed eventuali sconti (non bonus). Cliccando su esso si apre un'altra pagina in cui troviamo, ingrandite, le informazioni di prima e in più sono disponibili i bonus, è possibile definire la quantità voluta e con un tasto per aggiungere al carrello. Cliccando quest'ultimo:

- se l'utente non ha effettuato l'accesso, viene portato alla pagina di registrazione/login e al termine viene portato alla pagina del carrello, e vedrà l'articolo selezionato aggiunto.
- se l'utente è loggato, viene portato alla pagina del carrello dove potrà eventualmente continuare l'acquisto.

Dalla pagina del carrello può visionare un resoconto dei prodotti, tornare al catalogo, oppure verrà abilitata la funzione per procedere con l'ordine se il credito è sufficiente. Procedendo con l'ordine viene chiesto l'indirizzo di spedizione, precompilato con l'indirizzo di default associato al profilo. Successivamente il credito verrà scalato, i bonus assegnati e l'ordine verrà eseguito.

### 1.3.2 Recensione

L'utente, visitatore o cliente, all'interno della pagina relativa al prodotto, trova una sezione dedicata alle recensioni, dove per ognuna c'è il nome del recensore, il rating medio (di "supporto" e di "utilità") e il contenuto. Se l'utente è visitatore può solamente visualizzarle, mentre se è un cliente trova in alto nella sezione un box per inserire una recensione e per ogni recensione esistente valutare "supporto" e "utilità" cliccando sulle stelline, ma una volta sola.

Ovviamente un cliente non può valutare le sue stesse recensioni...

Il gestore può valutare le recensioni, come i clienti, ma anche eliminarle se le ritiene inopportune. Trova il tasto apposito sulla pagina del prodotto, vicino ad ogni recensione.

### 1.3.3 Domanda e risposta

Come per le recensioni. l'utente, visitatore o cliente, all'interno della pagina relativa al prodotto, trova una sezione dedicata alle domande e risposte. Per ognuna c'è il nome dell'utente che ha posto la domanda, il rating medio (di "supporto" e di "utilità") e il contenuto. Sotto ognuna di esse c'è l'elenco delle risposte, anch'esse con il nome dell'utente che ha inserito la risposta, il rating medio e il contenuto. Se l'utente è visitatore può solamente visualizzarle. Se è un cliente trova in alto nella sezione un box per inserire una domanda, un box sotto a ogni domanda per inserire una risposta e per ogni domanda e risposta valutare "supporto" e "utilità" cliccando sulle stelline,



ma una volta sola. Ovviamente un cliente non può valutare le sue domande e risposte...

Il gestore può valutare domande e risposte, come i clienti, ma anche eliminarle se le ritiene inopportune. Trova il tasto apposito sulla pagina di ogni prodotto, vicino ad ogni domanda e risposta.

Inoltre, il gestore può scrivere una risposta “definitiva” semplicemente inviandola come fa un cliente, ma non può fare domande.

#### 1.3.4 FAQ

Le FAQ (frequently asked questions) hanno una pagina apposita, accessibile da ogni tipologia di utente, anche visitatore. Visitatori e clienti possono solo visionarle, senza alcuna possibilità di modifica. Gestori e amministratori possono aggiungerle da zero, cliccando sul tasto “aggiungi” in fondo alla pagina. In alternativa, dalla pagina di un prodotto, sotto ogni domanda, trovano il tasto “eleva”. Prima, si sceglie la risposta da associare con un radio button, se le risposte provengono solo da clienti, mentre se è presente la risposta “definitiva” di un gestore viene scelta automaticamente, ma è comunque possibile cambiare la selezione. Cliccando su “eleva”, viene aperta la pagina di aggiunta di una nuova FAQ, precompilata con la suddetta domanda e risposta, per permetterne la modifica prima della pubblicazione.

#### 1.3.5 Offerte

Gli sconti e bonus vengono gestiti esclusivamente dai gestori. Il gestore trova una pagina apposita nella sua dashboard, chiamata “offerte”. All’interno trova una tabella con l’elenco degli sconti/bonus definiti. Al lato di ogni riga, è possibile eliminare l’offerta. Inoltre, in fondo alla tabella, si trovano i campi per l’inserimento di una nuova voce.

Gli sconti saranno visibili direttamente nella griglia del catalogo, mentre i bonus all’interno della pagina dedicata al prodotto. Un visitatore vedrà solo gli sconti/bonus “generali”, cioè non applicati ad una specifica categoria di cliente, mentre un cliente vede sconti/bonus generali ma anche quelli che si applicano ad esso.

Il resoconto degli sconti/bonus applicati sarà disponibile nel carrello.

#### 1.3.6 Gestione prodotti

I clienti/visitatori hanno un accesso in sola lettura ai prodotti in catalogo, cioè possono solo visualizzarli e [acquistarli](#). I gestori hanno una pagina nella loro dashboard con una tabella con un resoconto dei prodotti, tramite la quale possono modificarli, aprendo una pagina di modifica, ed eliminarli ed inoltre con un bottone alla fine della pagina possono aggiungerne di nuovi. Le quantità disponibili in stock possono essere modificate aprendo la pagina di modifica del prodotto.

Inoltre, quando viene finalizzato un acquisto, viene decrementata la quantità acquistata dai prodotti corrispondenti.

Se il prodotto ha quantità zero, cioè non è disponibile in magazzino, al gestore viene segnalato nella sua tabella prodotti da un'icona, mentre ai clienti/visitatori il tasto "aggiungi al carrello" non sarà disponibile e sarà presente un messaggio di prodotto "out-of-stock".

### 1.3.7 Reputazione

Per il calcolo della reputazione, si parte da un punteggio base, assegnato in fase di registrazione, di 30 punti.

Ogni operazione di valutazione ha un effetto sul punteggio dell'utente associato al contributo, ma non si può scendere sotto lo zero.

Le valutazioni sul supporto comportano:

- -2 punti se supporto=1
- 0 punti se supporto=2
- +2 punti se supporto=3

Le valutazioni sull'utilità comportano:

- -4 punti se utilità=1
- -2 punti se utilità=2
- 0 punti se utilità=3
- +2 punti se utilità=4
- +4 punti se utilità=5

Le suddette assegnazioni valgono per utenti che non hanno mai acquistato quel prodotto e non sono gestori.

Nel caso dell'utente che valuta contributi su un prodotto da esso già acquistato (es. valuto la domanda di un utente sotto ad una barretta proteica che ho acquistato la settimana scorsa) le assegnazioni dei punti raddoppieranno (es. valuto utilità=4 e la reputazione dell'utente verrà incrementata di 4 punti).

Nel caso di un gestore, i suoi rating sui contributi degli utenti influiranno su di essi andando a triplicare la normale assegnazione dei punti (es. valuto supporto=1 e la reputazione dell'utente verrà decrementata di 12 punti).

Come accennato nella descrizione iniziale ([cit.](#)) il giudizio di un cliente pesa in base alla propria reputazione ma sempre meno di un gestore. Oltre ai casi sopra citati, c'è un'ulteriore modifica all'assegnazione di punti, in modo che un cliente con un'alta reputazione influisca di più di uno con una reputazione bassa.

Si parte dal moltiplicatore iniziale  $k_i$  (di default 1, altrimenti 2 se il cliente ha acquistato quel prodotto), e con la seguente funzione viene calcolato il nuovo moltiplicatore  $k_f$ , sempre inferiore a 3:

$$k_f = 3 - (3 - k_i)e^{-\frac{x}{m}}$$

La costante  $m = \frac{1000}{\ln(4)}$  è stata scelta in modo che un utente con una reputazione di 1000 abbia un moltiplicatore di 2.5 se non ha acquistato il prodotto e 2.75 se ha acquistato il prodotto.

## 1.4 Strutture dati

### Utenti

#### Esempio

```
<?xml version="1.0" encoding="UTF-8"?>
<utenti xmlns="http://www.lweb.uni/tesina-rcstore/utenti/">
  <utente id="1" tipo="cliente" email="mario.rossi@gmail.com">
    <attivo>true</attivo>
    <nome>Mario</nome>
    <cognome>Rossi</cognome>
    <telefono>+390773123456</telefono>
    <indirizzo>Via Del Corso, 104, Latina, Italia</indirizzo>
    <codiceFiscale>RSSMRA83P12E472K</codiceFiscale>
    <credito>120.00</credito>
    <reputazione>100</reputazione>
    <password>5f4dcc3b5aa765d61d8327deb882cf99</password>
    <dataRegistrazione>2022-10-27</dataRegistrazione>
    <carrello>
      <prodotto id="3" quantita="2"/>
      <prodotto id="6" quantita="2"/>
      <prodotto id="11" quantita="1"/>
    </carrello>
  </utente>
  <utente id="2" tipo="gestore" email="luigi.bianchi@gmail.com">
    <nome>Luigi</nome>
    <cognome>Bianchi</cognome>
    <password>5f4dcc3b5aa765d61d8327deb882cf99</password>
  </utente>
  <utente id="3" tipo="admin" email="francesco.marroni@gmail.com">
    <nome>Francesco</nome>
    <cognome>Marroni</cognome>
    <password>5f4dcc3b5aa765d61d8327deb882cf99</password>
  </utente>
</utenti>
```

#### XSD

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ut="http://www.lweb.uni/tesina-rcstore/utenti/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/utenti/">

  <xsd:element name="utenti">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="ut:utente" minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```

        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="utente">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="ut:attivo" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ut:nome" minOccurs="1" maxOccurs="1" />
            <xsd:element ref="ut:cognome" minOccurs="1" maxOccurs="1" />
            <xsd:element ref="ut:telefono" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ut:indirizzo" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ut:codiceFiscale" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ut:credito" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ut:reputazione" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ut:password" minOccurs="1" maxOccurs="1" />
            <xsd:element ref="ut:dataRegistrazione" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ut:carrello" minOccurs="0" maxOccurs="1" />
        </xsd:sequence>
        <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
        <xsd:attribute name="email" type="xsd:string" use="required" />
        <xsd:attribute name="tipo" type="xsd:string" use="required" />
    </xsd:complexType>
</xsd:element>

<xsd:element name="carrello">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="ut:prodotto" minOccurs="0" maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="prodotto">
    <xsd:complexType>
        <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
        <xsd:attribute name="quantita" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
</xsd:element>

<xsd:element name="attivo" type="xsd:boolean" />
<xsd:element name="nome" type="xsd:string" />
<xsd:element name="cognome" type="xsd:string" />
<xsd:element name="telefono" type="xsd:string" />
<xsd:element name="indirizzo" type="xsd:string" />
<xsd:element name="codiceFiscale" type="xsd:string" />
<xsd:element name="credito" type="xsd:decimal" />
<xsd:element name="reputazione" type="xsd:integer" />
<xsd:element name="password" type="xsd:string" />
<xsd:element name="dataRegistrazione" type="xsd:date" />

</xsd:schema>

```

## DTD

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<!ELEMENT utenti (utente*)>

<!ELEMENT utente (attivo, nome, cognome, telefono, indirizzo, codiceFiscale, credito,
reputazione, password, dataRegistrazione, carrello)>
<!-- ATTLIST utente -->
  id      ID      #REQUIRED
  email   CDATA   #REQUIRED
  tipo    CDATA   #REQUIRED

<!-- ELEMENT attivo -->
<!-- ELEMENT nome -->
<!-- ELEMENT cognome -->
<!-- ELEMENT telefono -->
<!-- ELEMENT indirizzo -->
<!-- ELEMENT codiceFiscale -->
<!-- ELEMENT credito -->
<!-- ELEMENT reputazione -->
<!-- ELEMENT password -->
<!-- ELEMENT dataRegistrazione -->

<!-- ELEMENT carrello (prodotto*)>

<!-- ELEMENT prodotto -->
<!-- ATTLIST prodotto -->
  id      ID      #REQUIRED
  quantita CDATA   #REQUIRED

```

## Prodotti

### Esempio

```

<?xml version="1.0" encoding="UTF-8"?>
<prodotti xmlns="http://www.lweb.uni/tesina-rcstore/prodotti/">
  <prodotto id="1">
    <marca>MyProtein</marca>
    <nome>Proteine 2kg</nome>
    <descrizione>Descrizione</descrizione>
    <costo>20.00</costo>
    <categoria>Proteine</categoria>
    <quantita>54</quantita>
  </prodotto>
</prodotti>

```

### XSD

```

<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:prod="http://www.lweb.uni/tesina-rcstore/prodotti/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/prodotti/">

```

```

<xsd:element name="prodotti">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="prod:prodotto" minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="prodotto">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="prod:marca" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="prod:nome" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="prod:descrizione" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="prod:costo" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="prod:categoria" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="prod:quantita" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
  </xsd:complexType>
</xsd:element>

<xsd:element name="marca" type="xsd:string" />
<xsd:element name="nome" type="xsd:string" />
<xsd:element name="descrizione" type="xsd:string" />
<xsd:element name="costo" type="xsd:decimal" />
<xsd:element name="categoria" type="xsd:string" />
<xsd:element name="quantita" type="xsd:nonNegativeInteger" />

</xsd:schema>

```

## DTD

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT prodotti (prodotto*)>

<!ELEMENT prodotto (marca, nome, descrizione, costo, categoria, quantita)>
<!ATTLIST prodotto
  id ID #REQUIRED>

<!ELEMENT marca      (#PCDATA)>
<!ELEMENT nome       (#PCDATA)>
<!ELEMENT descrizione (#PCDATA)>
<!ELEMENT costo      (#PCDATA)>
<!ELEMENT categoria  (#PCDATA)>
<!ELEMENT quantita   (#PCDATA)>

```

## Ordini

### Esempio

```
<?xml version="1.0" encoding="UTF-8"?>
<ordini xmlns="http://www.lweb.uni/tesina-rcstore/ordini/">
  <ordine id="1" idUtente="3">
    <data>2023-05-15T23:10:47</data>
    <indirizzo>Via del Corso 101, Roma, Italia</indirizzo>
    <prezzoFinale>120.00</prezzoFinale>
    <prodotti>
      <prodotto id="3" quantita="2"/>
      <prodotto id="6" quantita="2"/>
      <prodotto id="11" quantita="1"/>
    </prodotti>
  </ordine>
  <ordine id="2" idUtente="1">
    <data>2023-06-08T17:29:56</data>
    <indirizzo>Via Doria 12, Latina, Italia</indirizzo>
    <prezzoFinale>80.00</prezzoFinale>
    <prodotti>
      <prodotto id="3" quantita="1"/>
      <prodotto id="5" quantita="1"/>
      <prodotto id="12" quantita="3"/>
    </prodotti>
  </ordine>
</ordini>
```

### XSD

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:ord="http://www.lweb.uni/tesina-rcstore/ordini/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/ordini/">

  <xsd:element name="ordini">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="ord:ordine" minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ordine">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="ord:data" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="ord:indirizzo" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="ord:prezzoFinale" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="ord:prodotti" minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```

        <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
        <xsd:attribute name="idUtente" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
</xsd:element>

<xsd:element name="prodotti">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element ref="ord:prodotto" minOccurs="1" maxOccurs="unbounded" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="prodotto">
    <xsd:complexType>
        <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
        <xsd:attribute name="quantita" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
</xsd:element>

<xsd:element name="data" type="xsd:dateTime" />
<xsd:element name="indirizzo" type="xsd:string" />
<xsd:element name="prezzoFinale" type="xsd:decimal" />

</xsd:schema>

```

## DTD

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT ordini (ordine*)>

<!ELEMENT ordine (data, indirizzo, prezzoFinale, prodotti)>
<!ATTLIST ordine
    id ID #REQUIRED
    idUtente CDATA #REQUIRED>

<!ELEMENT prodotti (prodotto*)>

<!ELEMENT prodotto>
<!ATTLIST prodotto
    id ID #REQUIRED
    quantita CDATA #REQUIRED>

<!ELEMENT data (#PCDATA)>
<!ELEMENT indirizzo (#PCDATA)>
<!ELEMENT prezzoFinale (#PCDATA)>

```



## FAQ

### Esempio

```
<?xml version="1.0" encoding="UTF-8"?>
<faqs xmlns="http://www.lweb.uni/tesina-rcstore/faq/">
  <faq id="1">
    <domanda>Quanto costa la spedizione&#63;</domanda>
    <risposta>La spedizione varia tra 1 &euro; e 5 &euro; e dipende dalla
destinazione.</risposta>
  </faq>
</faqs>
```

### XSD

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:faq="http://www.lweb.uni/tesina-rcstore/faq/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/faq/">

  <xsd:element name="faqs">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="faq:faq" minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="faq">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="faq:domanda" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="faq:risposta" minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
      <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="domanda" type="xsd:string" />
  <xsd:element name="risposta" type="xsd:string" />

</xsd:schema>
```

### DTD

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<!ELEMENT faqs (faq*)>

<!ELEMENT faq (domanda, risposta)>
<!--ATTLIST faq
      id ID #REQUIRED-->

<!ELEMENT domanda (#PCDATA)>
<!ELEMENT risposta (#PCDATA)>

```

## Domande

### Esempio

```

<?xml version="1.0" encoding="UTF-8"?>
<domande xmlns="http://www.lweb.uni/tesina-rcstore/domande/">
  <domanda id="1" idProdotto="14">
    <idUtente>24</idUtente>
    <contenuto>Quali gusti sono disponibili?</contenuto>
    <ratings>
      <rating idUtente="37">
        <supporto>1</supporto>
        <utilita>4</utilita>
      </rating>
    </ratings>
    <risposte>
      <risposta id="1">
        <idUtente>51</idUtente>
        <contenuto>Cioccolata e lampone</contenuto>
        <ratings>
          <rating idUtente="29">
            <supporto>3</supporto>
            <utilita>5</utilita>
          </rating>
        </ratings>
      </risposta>
    </risposte>
  </domanda>
</domande>

```

### XSD

```

<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:dom="http://www.lweb.uni/tesina-rcstore/domande/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/domande/">

  <xsd:element name="domande">

```

```

<xsd:complexType>
  <xsd:sequence>
    <xsd:element ref="dom:domanda" minOccurs="0" maxOccurs="unbounded" />
  </xsd:sequence>
</xsd:complexType>
</xsd:element>

<xsd:element name="domanda">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="dom:idUtente" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="dom:contenuto" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="dom:ratings" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="dom:risposte" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
    <xsd:attribute name="idProdotto" type="xsd:positiveInteger" use="required" />
  </xsd:complexType>
</xsd:element>

<xsd:element name="risposte">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="dom:risposta" minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="risposta">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="dom:idUtente" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="dom:contenuto" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="dom:ratings" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
  </xsd:complexType>
</xsd:element>

<xsd:element name="ratings">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="dom:rating" minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="rating">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="dom:supporto" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="dom:utilita" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="idUtente" type="xsd:positiveInteger" use="required" />
  </xsd:complexType>
</xsd:element>

```

```

<xsd:element name="idUtente"      type="xsd:positiveInteger" />
<xsd:element name="contenuto"     type="xsd:string" />
<xsd:element name="supporto"      type="xsd:integer" />
<xsd:element name="utilita"       type="xsd:integer" />

</xsd:schema>

```

## DTD

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT domande (domanda*)>

<!ELEMENT domanda (idProdotto, contenuto, ratings, risposte)>
<!ATTLIST domanda
    id          ID #REQUIRED
    idUtente    CDATA #REQUIRED>

<!ELEMENT risposte (risposta*)>

<!ELEMENT risposta (idUtente, contenuto, ratings)>
<!ATTLIST risposta
    id ID #REQUIRED>

<!ELEMENT ratings (rating*)>

<!ELEMENT rating (supporto, utilita)>
<!ATTLIST rating
    idUtente    CDATA #REQUIRED>

<!ELEMENT idUtente      (#PCDATA)>
<!ELEMENT idProdotto    (#PCDATA)>
<!ELEMENT contenuto     (#PCDATA)>
<!ELEMENT supporto      (#PCDATA)>
<!ELEMENT utilita       (#PCDATA)>

```

## Recensioni

### Esempio

```

<?xml version="1.0" encoding="UTF-8"?>
<recensioni xmlns="http://www.lweb.uni/tesina-rcstore/recensione/">
  <recensione id="1" idProdotto="21">
    <idUtente>12</idUtente>
    <contenuto>Recensione di prova</contenuto>
    <ratings>
      <rating id="1" idUtente="37">
        <supporto>1</supporto>
        <utilita>4</utilita>
      </rating>
    </ratings>
  </recensione>
</recensioni>

```

```
    </ratings>
  </recensione>
</recensioni>
```

## XSD

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:rec="http://www.lweb.uni/tesina-rcstore/recensione/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/recensione/">

  <xsd:element name="recensioni">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="rec:recensione" minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="recensione">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="rec:idUtente" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="rec:contenuto" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="rec:ratings" minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
      <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
      <xsd:attribute name="idProdotto" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ratings">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="rec:rating" minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="rating">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="rec:supporto" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="rec:utilita" minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
      <xsd:attribute name="idUtente" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="idUtente" type="xsd:positiveInteger" />
  <xsd:element name="contenuto" type="xsd:string" />
  <xsd:element name="supporto" type="xsd:integer" />
  <xsd:element name="utilita" type="xsd:integer" />
```

```
</xsd:schema>
```

## DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT recensioni (recensione*)>

<!ELEMENT recensione (idUtente, contenuto, ratings)>
<!ATTLIST recensione
    id          ID #REQUIRED
    idProdotto  CDATA #REQUIRED>

<!ELEMENT ratings (rating*)>

<!ELEMENT rating (supporto, utilita)>
<!ATTLIST rating
    idUtente  CDATA #REQUIRED>

<!ELEMENT idUtente    (#PCDATA)>
<!ELEMENT contenuto    (#PCDATA)>
<!ELEMENT supporto    (#PCDATA)>
<!ELEMENT utilita      (#PCDATA)>
```

## Accrediti

### Esempio

```
<?xml version="1.0" encoding="UTF-8"?>
<accrediti xmlns="http://www.lweb.uni/tesina-rcstore/accrediti/">
  <accredito id="1">
    <data>2023-07-10T17:29:56</data>
    <idUtente>7</idUtente>
    <quantita>20.00</quantita>
  </accredito>
</accrediti>
```

## XSD

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:acc="http://www.lweb.uni/tesina-rcstore/accrediti/"
    targetNamespace="http://www.lweb.uni/tesina-rcstore/accrediti/">
```

```

<xsd:element name="accrediti">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="acc:accredito" minOccurs="0" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<xsd:element name="accredito">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="acc:data" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="acc:idUtente" minOccurs="1" maxOccurs="1" />
      <xsd:element ref="acc:quantita" minOccurs="1" maxOccurs="1" />
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
  </xsd:complexType>
</xsd:element>

<xsd:element name="data" type="xsd:dateTime" />
<xsd:element name="idUtente" type="xsd:positiveInteger" />
<xsd:element name="quantita" type="xsd:decimal" />
</xsd:schema>

```

## DTD

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT accrediti (accredito*)>

<!ELEMENT accredito (data, idUtente, quantita)>
<!ATTLIST accredito
  id ID #REQUIRED>

<!ELEMENT data (#PCDATA)>
<!ELEMENT idUtente (#PCDATA)>
<!ELEMENT quantita (#PCDATA)>

```

## Categorie

### Esempio

```

<?xml version="1.0" encoding="UTF-8"?>
<categorie xmlns="http://www.lweb.uni/tesina-rcstore/categorie/">
  <categoria id="1">
    <nome>Proteine</nome>
  </categoria>
  <categoria id="2">
    <nome>Barrette</nome>
  </categoria>
  <categoria id="3">

```

```

    <nome>Vitamine</nome>
  </categoria>
</categorie>

```

## XSD

```

<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:cat="http://www.lweb.uni/tesina-rcstore/categorie/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/categorie/">

  <xsd:element name="categorie">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="cat:categoria" minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="categoria">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="cat:nome" minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
      <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="nome" type="xsd:string" />

</xsd:schema>

```

## DTD

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT categorie (categoria*)>

<!ELEMENT categoria (nome)>
<!ATTLIST categoria
  id ID #REQUIRED>

<!ELEMENT nome (#PCDATA)>

```



## Offerte

### Esempio

```
<?xml version="1.0" encoding="UTF-8"?>
<offerte xmlns="http://www.lweb.uni/tesina-rcstore/offerte/">
  <offerta id="1">
    <tipo>sconto</tipo>
    <percentuale>20</percentuale>
    <target>credData</target>
    <creditiSpesi>100.00</creditiSpesi>
    <dataInizio>2023-05-15</dataInizio>
  </offerta>
  <offerta id="2">
    <tipo>bonus</tipo>
    <numCrediti>3</numCrediti>
    <target>eccMag</target>
    <idProdotto>51</idProdotto>
    <quantitaMin>100</quantitaMin>
  </offerta>
</offerte>
```

### XSD

```
<?xml version="1.0" encoding="UTF-8" ?>
<xsd:schema
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:off="http://www.lweb.uni/tesina-rcstore/offerte/"
  targetNamespace="http://www.lweb.uni/tesina-rcstore/offerte/">

  <xsd:element name="offerte">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="off:offerta" minOccurs="0" maxOccurs="unbounded" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="offerta">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="off:tipo" minOccurs="1" maxOccurs="1" />
        <xsd:element ref="off:percentuale" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:numCrediti" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:target" minOccurs="1" maxOccurs="1" />

        <xsd:element ref="off:creditiSpesi" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:dataInizio" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:reputazione" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:anni" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:idProdotto" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:idCategoria" minOccurs="0" maxOccurs="1" />
        <xsd:element ref="off:quantitaMin" minOccurs="0" maxOccurs="1" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```

        <xsd:attribute name="id" type="xsd:positiveInteger" use="required" />
    </xsd:complexType>
</xsd:element>

<xsd:element name="tipo" type="xsd:string" />
<xsd:element name="percentuale" type="xsd:positiveInteger" />
<xsd:element name="numCrediti" type="xsd:decimal" />
<xsd:element name="target" type="xsd:string" />

<xsd:element name="creditiSpesi" type="xsd:decimal" />
<xsd:element name="dataInizio" type="xsd:date" />
<xsd:element name="reputazione" type="xsd:positiveInteger" />
<xsd:element name="anni" type="xsd:positiveInteger" />
<xsd:element name="idProdotto" type="xsd:positiveInteger" />
<xsd:element name="idCategoria" type="xsd:positiveInteger" />
<xsd:element name="quantitaMin" type="xsd:positiveInteger" />

</xsd:schema>

```

## DTD

```

<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT offerte (offerta*)>

<!ELEMENT offerta (tipo, percentuale, numCrediti, target, creditiSpesi, dataInizio,
reputazione, anni, idProdotto, idCategoria, quantitaMin)>
<!ATTLIST offerta
    id ID #REQUIRED>

<!ELEMENT tipo (#PCDATA)>
<!ELEMENT percentuale (#PCDATA)>
<!ELEMENT numCrediti (#PCDATA)>
<!ELEMENT target (#PCDATA)>

<!ELEMENT creditiSpesi (#PCDATA)>
<!ELEMENT dataInizio (#PCDATA)>
<!ELEMENT reputazione (#PCDATA)>
<!ELEMENT anni (#PCDATA)>
<!ELEMENT idProdotto (#PCDATA)>
<!ELEMENT idCategoria (#PCDATA)>
<!ELEMENT quantitaMin (#PCDATA)>

```

## 1.5 Spiegazione elementi/attributi

- utenti/utente[*tipo*]: la presenza dell'attributo "tipo" relativo all'utente, è resa necessaria per distinguere le tipologie di utenti (cliente, gestore, admin) presenti e controllare se l'utente dispone dei permessi necessari per eseguire le operazioni. La necessità di questo attributo, la si evince dall'esistenza della sezione 1.2 ([tipi di utenti](#)), in quanto lo scopo di essa è proprio definire le differenze di funzionalità tra utenti.
- utenti/utente/carrello/prodotto[*quantita*]: la presenza dell'attributo "quantita" relativo ad un prodotto all'interno di un carrello, è necessaria per tenere traccia della quantità del singolo prodotto nel carrello corrente dell'utente.  
La necessità di questo attributo è data dalla specifica definita nella sezione 1.3.1 ([acquisto](#)), riga 4.
- utenti/utente/*reputazione*: la presenza dell'elemento "reputazione" associato ad un utente, è necessaria per tenere traccia delle variazioni di valore di quest'ultimo, in quanto è uno dei parametri per l'applicazione personalizzata di eventuali bonus/sconti.  
La necessità di questo elemento si evince dalla sezione 1.1 ([descrizione](#)), paragrafo "[reputazione](#)", riga 1.
- prodotti/prodotto/*categoria*: la presenza dell'elemento "categoria" associato ad un prodotto è necessario per stabilire la categoria di appartenenza del prodotto (es. proteine), in quanto la categoria rappresenta uno dei criteri di ordinamento dei prodotti nel catalogo e di applicazione di sconti/bonus.  
La necessità di questo elemento si evince dalla sezione 1.1 ([descrizione](#)), paragrafo "[categorie](#)", riga 1.
- ordini/ordine[*idUtente*]: la presenza dell'attributo "idUtente" relativo ad un ordine risulta fondamentale per conoscere l'identità del cliente che lo ha effettuato e per permettere di visualizzare lo storico degli ordini di un determinato cliente.  
La necessità di questo attributo la si può evincere dalla sezione 1.2 ([tipi di utenti](#)), paragrafo "[utente registrato](#)", punto 2: "può vedere [...] storico degli acquisti".
- domande/domanda/ratings/*rating*: la presenza dell'elemento "rating", il quale è un ComplexType formato da più elementi, è fondamentale per tenere traccia della valutazione che i clienti associano alle domande/risposte di un prodotto secondo i criteri di "supporto" e "utilità".  
La necessità di questo elemento si evince dalla sezione 1.3.3 ([domande e risposte](#)), riga 3.
- recensioni/recensione[*idProdotto*]: la presenza dell'attributo "idProdotto" all'interno dell'elemento recensione, è necessaria per poter

conoscere a quale prodotto è relativa una recensione, potendo così mostrare nella pagina del prodotto unicamente le recensioni ad esso relative.

La necessità di questo attributo la si può evincere dalla sezione 1.3.2 ([recensioni](#)), riga 1.

- `accrediti/accredito[idUtente]`: la presenza dell'attributo "idUtente" all'interno dell'elemento `accredito` permette di poter risalire all'utente che ha svolto la richiesta di accredito di una determinata somma e mostrarlo all'admin nella pagina apposita, per la verifica "offline" del pagamento. Si evince la sua utilità dalla sezione 1.1 (descrizione), paragrafo [crediti](#), righe 3...6.
- `offerte/offerta/tipo`: la presenza dell'elemento "tipo" relativamente ad un'offerta è necessaria per poter distinguere e definire la natura di quest'ultima quindi se sconto o bonus. Nel primo caso viene considerato l'elemento percentuale, mentre nel secondo caso l'elemento "numCrediti". Inoltre, come descritto nella sezione 1.3.5 ([offerte](#)), righe 6...9, uno sconto è visibile anche dal catalogo, mentre i bonus solo dalla pagina del prodotto.
- `offerte/offerta/target`: la presenza dell'elemento "target" relativamente ad un'offerta va a definire quale sia il suo criterio di applicazione (es. offerta applicata ai clienti che hanno speso X crediti finora). Viene giustificata dalla sezione 1.1 (descrizione), paragrafo [offerte](#), righe 9...21.
- `offerte/offerta/quantitaMin`: la presenza dell'elemento "quantitaMin" risulta necessario nello scenario in cui l'offerta sia relativa ad un'eccedenza di prodotto nel magazzino e si voglia impostare una soglia la quale una volta raggiunta pone fine all'offerta. Si evince la sua necessità dalla definizione dell'offerta di tipo "eccedenza", definita nella sezione 1.1 (descrizione), paragrafo [offerte](#), righe 20-21.

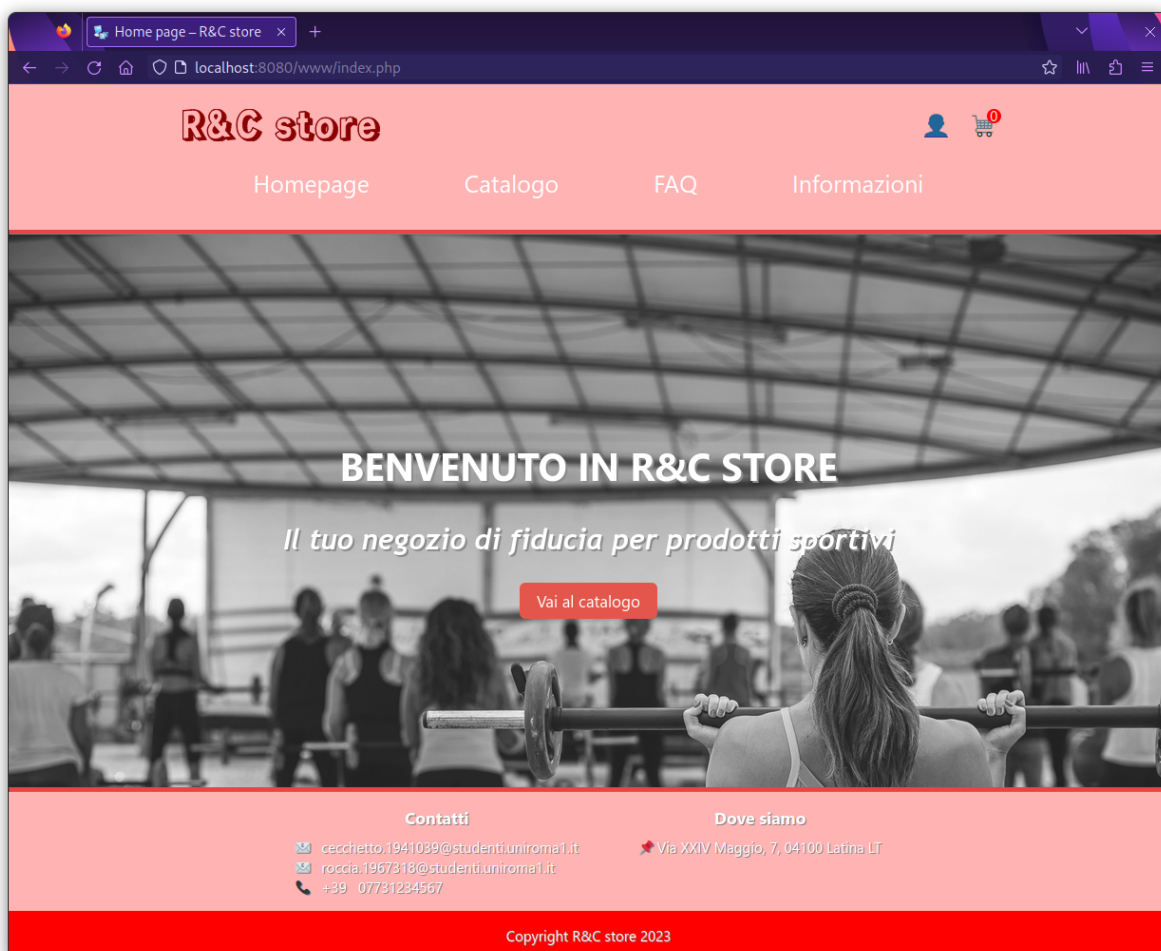
## 2. Manuale utente

- Le parti in blu sono operazioni che può effettuare solo un cliente
- Le parti in rosso sono operazioni proprie dei gestori/amministratori

### 2.1 Struttura

Ogni pagina del sito è divisa in tre parti, header, contenuto, footer, separate visivamente con colori diversi.

- L'header contiene i link alle pagine più importanti: homepage, catalogo, FAQ e informazioni. Inoltre in alto a destra ha l'icona di un utente che svolge una triplice funzione: per fare il login/registrazione se non si è loggati, [accedere al proprio profilo se si è clienti](#) e [accedere alla propria dashboard se si è gestori/amministratori](#). Il cliente inoltre ha l'icona per accedere al [carrello](#) con il conteggio degli articoli attualmente inseriti.
- Il contenuto varia da pagina a pagina
- Il footer presenta le opzioni di contatto del negozio e la sua posizione

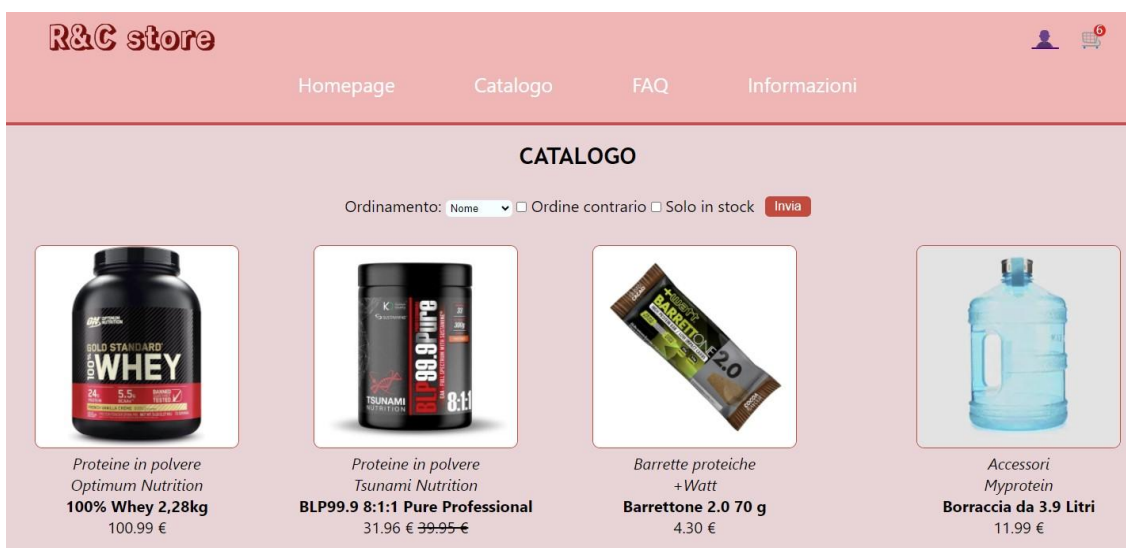


## 2.2 Home page

Appena si accede al sito ci si trova davanti alla pagina iniziale di benvenuto. Per tutti gli utenti è disponibile un tasto per accedere alla pagina del catalogo, dove si possono visionare i prodotti (img. a sinistra), **mentre per gli “utenti speciali” (admin e gestore) è disponibile un ulteriore tasto per accedere alla propria dashboard di gestione.** (img. a destra).



## 2.3 Catalogo



All'interno della pagina “Catalogo” è possibile vedere i prodotti in vendita in una griglia. Per ognuno è specificata, sotto l'immagine, la categoria di appartenenza, la marca, il nome del prodotto, il costo. In caso di sconto applicato, il costo originale è barrato e viene mostrato il costo scontato.

A tutti gli utenti è permesso ordinare i prodotti andando nella sezione «ordinamento» in cima alla pagina, scegliendo il criterio (nome, categoria, marca o prezzo del prodotto), la modalità, se crescente (di default) oppure decrescente (spuntando la checkbox «ordine contrario») ed infine se si vogliono visualizzare unicamente i prodotti in stock, cioè che sono disponibili in magazzino ed è possibile acquistare. Per applicare i parametri, si clicca sul bottone «invia».

Cliccando sul prodotto si apre la sua pagina con ulteriori dettagli.

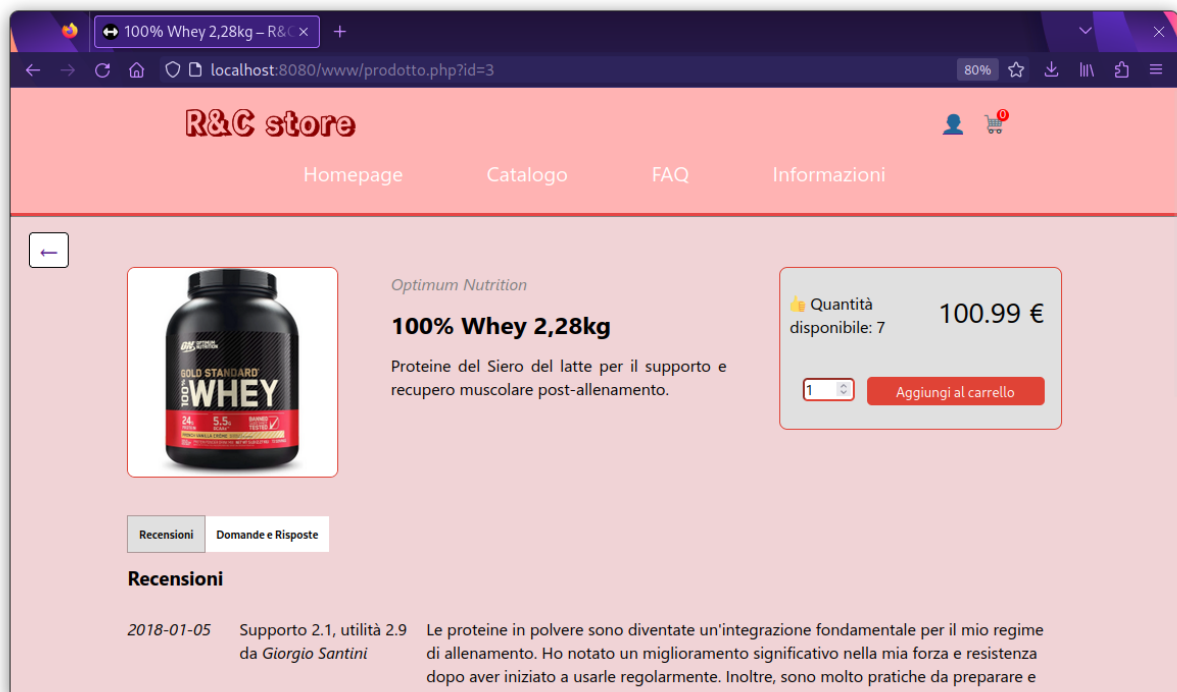
## 2.4 Prodotto

La pagina del prodotto contiene le sue informazioni dettagliate.

Nella parte superiore della pagina troviamo a sinistra l'immagine del prodotto, al centro la sua descrizione **ed eventualmente per i gestori/admin il tasto «modifica info»**, mentre a destra un box in cui sono disponibili le informazioni per l'acquisto: quantità disponibile, sconti e bonus applicati, il box per scegliere la quantità da acquistare e il tasto «aggiungi al carrello».

Cliccando su «aggiungi al carrello», se si è clienti e loggati, si viene reindirizzati alla pagina del carrello da cui è possibile vedere l'esito dell'operazione, altrimenti si viene reindirizzati alla pagina di login per procedere con il login/registrazione ([link](#)).

La parte inferiore invece contiene recensioni e domande e risposte, spiegate di seguito.



### 2.4.1 Recensioni

Cliccando sul bottone «recensioni» sotto l'immagine del prodotto è possibile visionare le recensioni relative a quest'ultimo, le quali sono articolate come segue: data di rilascio della recensione, media delle valutazioni relative al supporto e utilità rilasciate dagli utenti, nome e cognome di chi ha rilasciato la recensione e il testo della recensione.






## Utente visitatore/amministratore

Recensioni		
Domande e Risposte		
Recensioni		
2017-01-29	Supporto 2.0, utilità 1.0 da Mario Rossi	Le proteine in polvere sono diventate un elemento essenziale nella mia routine quotidiana. Mi danno la giusta quantità di energia per affrontare gli allenamenti e mi aiutano a recuperare più velocemente. Non posso fare a meno di raccomandare le proteine in polvere a chiunque sia seriamente interessato al fitness.
2021-05-05	Supporto 2.0, utilità 1.0 da Mario Rossi	Le proteine in polvere sono state una scoperta incredibile per me. Non solo sono facili da assumere e digerire, ma hanno anche un sapore delizioso. Consiglio vivamente di provarle se si è alla ricerca di un modo semplice per raggiungere i propri obiettivi di allenamento.
2019-06-19	Supporto 0.0, utilità 0.0 da Marco Verdi	Sono rimasto soddisfatto delle proteine in polvere che ho acquistato. Ho notato un miglioramento nella mia forza e una diminuzione del tempo di recupero muscolare. Il sapore è ottimo e si mischiano facilmente, senza grumi.
2022-01-19	Supporto 1.0, utilità 4.0 da Mario Rossi	Ho iniziato ad usare le proteine in polvere dopo aver iniziato un programma di allenamento intenso. Mi sono realmente sorpreso dai risultati che ho ottenuto. Mi sento più tonico e i miei muscoli sembrano essere più definiti. Consiglio sicuramente le proteine in polvere agli altri appassionati di fitness.

Nel caso in cui l'utente sia un visitatore/amministratore non vi è alcuna operazione da poter svolgere ma risulta vincolato alla sola visualizzazione delle recensioni.

## Utente cliente

Recensioni		
Domande e Risposte		
Recensioni		
<a href="#">Scrivi una nuova recensione</a>		
2017-01-29	Supporto 2.0, utilità 1.0 da Te 	Le proteine in polvere sono diventate un elemento essenziale nella mia routine quotidiana. Mi danno la giusta quantità di energia per affrontare gli allenamenti e mi aiutano a recuperare più velocemente. Non posso fare a meno di raccomandare le proteine in polvere a chiunque sia seriamente interessato al fitness.
2021-05-05	Supporto 2.0, utilità 1.0 da Te 	Le proteine in polvere sono state una scoperta incredibile per me. Non solo sono facili da assumere e digerire, ma hanno anche un sapore delizioso. Consiglio vivamente di provarle se si è alla ricerca di un modo semplice per raggiungere i propri obiettivi di allenamento.
2019-06-19	Supporto 0.0, utilità 0.0 da Marco Verdi	Sono rimasto soddisfatto delle proteine in polvere che ho acquistato. Ho notato un miglioramento nella mia forza e una diminuzione del tempo di recupero muscolare. Il sapore è ottimo e si mischiano facilmente, senza grumi.
<div>Supporto: ☆☆☆☆ Utilità: ☆☆☆☆☆ <input type="button" value="Invia"/></div>		
2022-01-19	Supporto 1.0, utilità 4.0 da Te 	Ho iniziato ad usare le proteine in polvere dopo aver iniziato un programma di allenamento intenso. Mi sono realmente sorpreso dai risultati che ho ottenuto. Mi sento più tonico e i miei muscoli sembrano essere più definiti. Consiglio sicuramente le proteine in polvere agli altri appassionati di fitness.

Nel caso in cui l'utente sia un cliente per lui sarà possibile riconoscere i propri contributi in quanto questi ultimi saranno contornati e colorati da un colore diverso rispetto agli altri e nella sezione in cui viene riportato nome e cognome dell'autore della recensione è riportata la dicitura "da Te".

Oltre alla visualizzazione della recensione è possibile:

- Aggiungere una recensione: è possibile tramite il bottone apposito "Scrivi una nuova recensione" scrivere nell'area che verrà appositamente mostrata e per inviarla basterà cliccare sul bottone a fianco.



Recensioni
Domande e Risposte

### Recensioni

Scritta una nuova recensione

---

2017-01-29
Supporto 2.0, utilità 1.0
da **Te**

Le proteine in polvere sono diventate un elemento essenziale nella mia routine quotidiana. Mi danno la giusta quantità di energia per affrontare gli allenamenti e mi aiutano a recuperare più velocemente. Non posso fare a meno di raccomandare le proteine in polvere a chiunque sia seriamente interessato al fitness.

- Valutare una recensione: all'interno della sezione in cui viene riportata la valutazione media della recensione è possibile tramite l'apposito box valutare la recensione in termini di supporto e utilità inviando il proprio contributo cliccando sul pulsante "Invia", nel momento in cui una delle due valutazioni non risulti assegnata le verrà assegnato 1 automaticamente.

2019-06-19
Supporto 0.0, utilità 0.0
da **Marco Verdi**

Supporto: ☆☆☆
Utilità: ☆☆☆☆☆

Invia

## Utente gestore

Recensioni
Domande e Risposte

### Recensioni

Scritta una nuova recensione

2017-01-29	Supporto 2.0, utilità 1.0 da <b>Mario Rossi</b>	<div> Supporto: ☆☆☆ Utilità: ☆☆☆☆☆ <div>Invia</div> </div> <p>Le proteine in polvere sono diventate un elemento essenziale nella mia routine quotidiana. Mi danno la giusta quantità di energia per affrontare gli allenamenti e mi aiutano a recuperare più velocemente. Non posso fare a meno di raccomandare le proteine in polvere a chiunque sia seriamente interessato al fitness.</p>
2021-05-05	Supporto 2.0, utilità 1.0 da <b>Mario Rossi</b>	<div> Supporto: ☆☆☆ Utilità: ☆☆☆☆☆ <div>Invia</div> </div> <p>Le proteine in polvere sono state una scoperta incredibile per me. Non solo sono facili da assumere e digerire, ma hanno anche un sapore delizioso. Consiglio vivamente di provarle se si è alla ricerca di un modo semplice per raggiungere i propri obiettivi di allenamento.</p>
2019-06-19	Supporto 0.0, utilità 0.0 da <b>Marco Verdi</b>	<div> Supporto: ☆☆☆ Utilità: ☆☆☆☆☆ <div>Invia</div> </div> <p>Sono rimasto soddisfatto delle proteine in polvere che ho acquistato. Ho notato un miglioramento nella mia forza e una diminuzione del tempo di recupero muscolare. Il sapore è ottimo e si mischiano facilmente, senza grumi.</p>
2022-01-19	Supporto 1.0, utilità 4.0 da <b>Mario Rossi</b>	<div> Supporto: ☆☆☆ Utilità: ☆☆☆☆☆ <div>Invia</div> </div> <p>Ho iniziato ad usare le proteine in polvere dopo aver iniziato un programma di allenamento intenso. Mi sono realmente sorpreso dai risultati che ho ottenuto. Mi sento più tonico e i miei muscoli sembrano essere più definiti. Consiglio sicuramente le proteine in polvere agli altri appassionati di fitness.</p>

Nel caso in cui l'utente sia un gestore è in grado di svolgere le seguenti azioni:

- Valutare una recensione (come l'utente cliente).
- Eliminare una recensione: nella stessa sezione in cui è presente il testo della recensione, ad inizio paragrafo, il gestore trova un'icona la quale se cliccata elimina la recensione ad essa associata.

## 2.4.2 Domande e risposte

Cliccando sul bottone «domande e risposte» sotto l'immagine del prodotto è possibile visionare le recensioni relative a quest'ultimo, le quali sono articolate come segue: data di rilascio della domanda/risposta, il contenuto di quest'ultima, la media delle valutazioni relative al supporto e utilità rilasciate dagli utenti, nome e cognome di chi ha rilasciato il contributo, e nell'ultima sezione se:

- l'utente è loggato e la domanda/risposta rilasciata è un suo contributo non troverà nulla.
- l'utente è loggato e la domanda/risposta rilasciata non è un suo contributo troverà un apposito form di valutazione per valutarlo.
- l'utente è/non è loggato e la domanda/risposta rilasciata appartiene ad un gestore viene visualizzata la scritta "risposta del gestore"

2022-10-27	Quali gusti sono disponibili?	Supporto 0.0, utilità 0.0 da <i>Marco Verdi</i>	Supporto: ☆☆☆☆ Utilità: ☆☆☆☆ <input type="button" value="Invia"/>
2022-10-27	Cioccolata e lampone	Supporto 0.0, utilità 0.0 da <i>Te</i>	
2022-10-27	Cioccolata e mango	Supporto 0.0, utilità 0.0 da <i>Te</i>	
2022-10-27	Risposta gestore	Supporto 0.0, utilità 0.0 da <i>Luigi Bianchi</i>	<b>RISPOSTA DEL GESTORE</b>
2022-10-27	Quanto costa?	Supporto 0.0, utilità 0.0 da <i>Te</i>	

### Utente visitatore/amministratore

Nel caso in cui l'utente sia un visitatore/amministratore non vi è alcuna operazione da poter svolgere ma risulta vincolato alla sola visualizzazione delle domande/risposte.

2022-10-27	Quali gusti sono disponibili?	Supporto 0.0, utilità 0.0 da <i>Marco Verdi</i>	
2022-10-27	Cioccolata e lampone	Supporto 0.0, utilità 0.0 da <i>Mario Rossi</i>	
2022-10-27	Cioccolata e mango	Supporto 0.0, utilità 0.0 da <i>Mario Rossi</i>	
2022-10-27	Risposta gestore	Supporto 0.0, utilità 0.0 da <i>Luigi Bianchi</i>	<b>RISPOSTA DEL GESTORE</b>

## Utente cliente

Nel caso in cui l'utente sia un cliente per lui sarà possibile riconoscere i propri contributi in quanto questi ultimi saranno contornati e colorati da un colore diverso rispetto agli altri e nella sezione in cui viene riportato nome e cognome dell'autore della domanda/risposta è riportata la dicitura "da Te".

Oltre alla visualizzazione dei contributi è possibile:

- Aggiungere un contributo: è possibile tramite il bottone apposito "Scrivi una nuova domanda" aggiungere una nuova domanda scrivendo nell'area che verrà appositamente mostrata e per inviarla basterà cliccare sul bottone a fianco (img. 1), oppure tramite il bottone "rispondi" è possibile rispondere ad una domanda esistente scrivendo nell'area che verrà appositamente mostrata e per inviarla basterà cliccare sul bottone "invia" (img. 2).

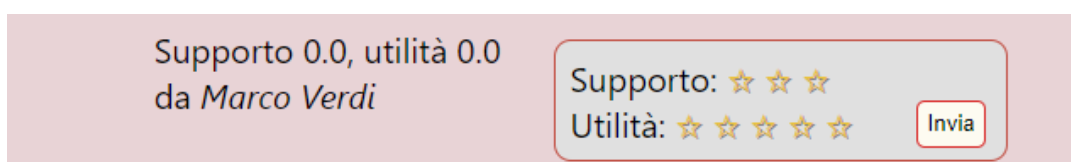


Immagine 1



Immagine 2

- Valutare un contributo: all'interno della sezione in cui viene riportata la valutazione media della recensione è possibile tramite l'apposito box valutare il contributo in termini di supporto e utilità inviando le valutazioni tramite il pulsante "Invia". Nel caso in cui una delle due valutazioni non risulti assegnata le verrà assegnato 1 automaticamente.



## Utente gestore

Nel caso in cui l'utente sia un gestore è in grado di svolgere le seguenti azioni:

- Rispondere: nel caso in cui un gestore risponda ad una domanda, quest'ultima viene trattata come risposta definitiva ed altri utenti non potranno più rispondervi.
- Valutare un contributo (come l'utente cliente).
- Eliminare un contributo: nella stessa sezione in cui è presente il testo del contributo, ad inizio paragrafo, il gestore trova un'icona la quale se cliccata elimina contributo ad essa associato. (img. 1)
- Elevare una domanda alle FAQ: il gestore cliccando su una domanda, visionando le relative risposte può sceglierne una (di default viene selezionata se esistente la risposta di un gestore) e cliccando sul bottone «eleva» può elevare la domanda con relativa risposta selezionata a diventare una FAQ. Prima di essere confermata come FAQ, il gestore viene riportato in un'apposita pagina in cui gli viene data la possibilità di modificare i due contributi e successivamente cliccando sul bottone «aggiungi» i due contributi entreranno a far parte delle FAQ. (img. 2, 3)



Immagine 1

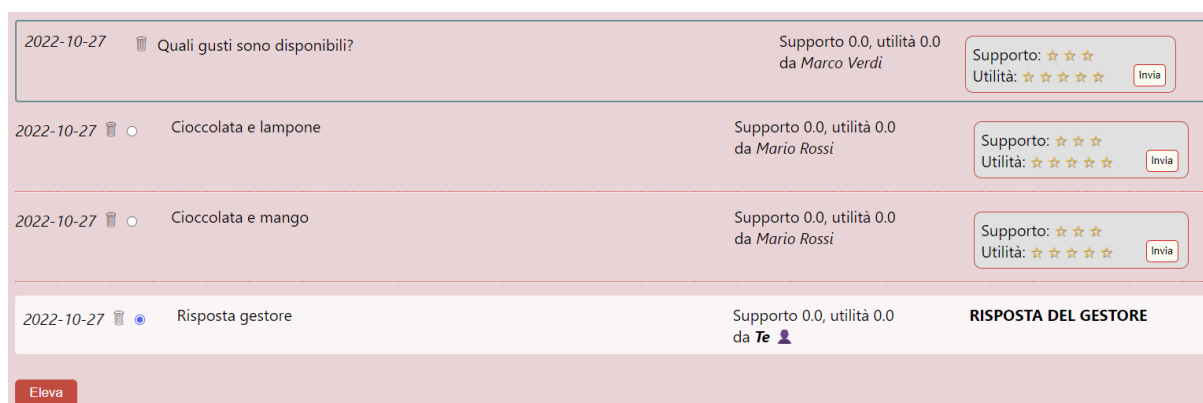


Immagine 2

## AGGIUNGI FAQ

### Domanda:

Quali gusti sono disponibili?

---

### Risposta:

Risposta gestore

---

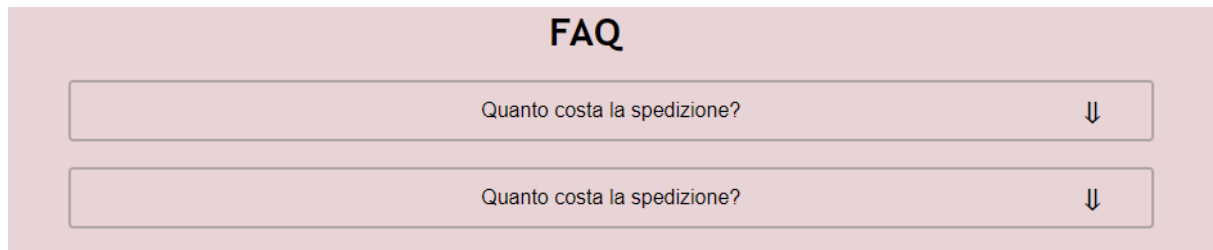
Aggiungi

Torna indietro

Immagine 3

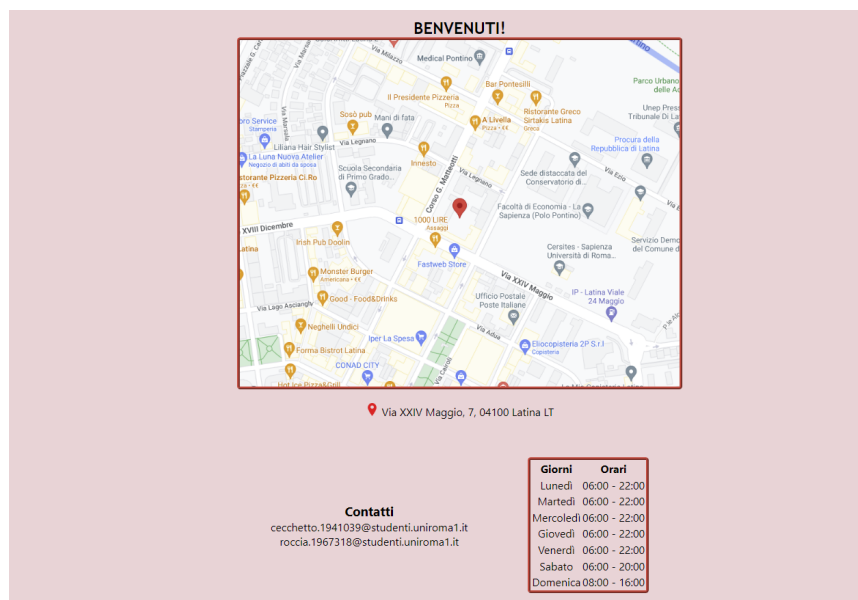
## 2.5 FAQ

La pagina delle FAQ è visibile a tutti gli utenti. Sono elencate le domande che gestori e amministratori hanno deciso di mettere in evidenza, perché chieste frequentemente oppure di particolare utilità. Cliccando sul box della domanda compare la risposta corrispondente.



## 2.6 Informazioni

E' una pagina visibile a tutti contenente delle informazioni sul negozio. E' presente una mappa, i contatti dei proprietari e gli orari di apertura.



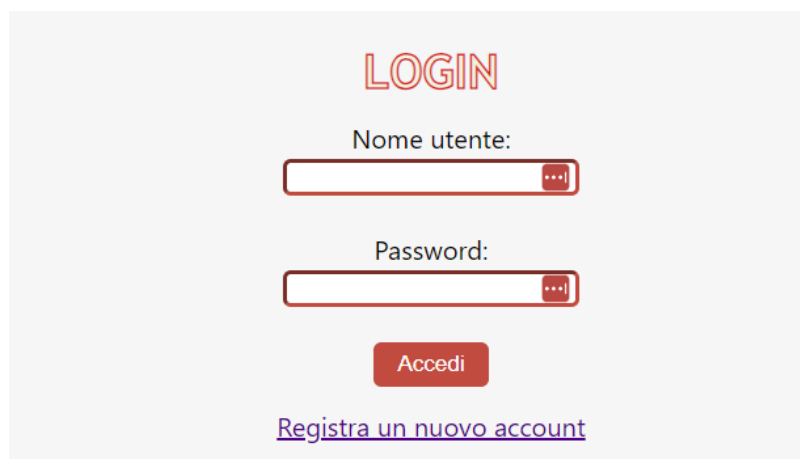
## 2.7 Accesso

L'accesso è necessario per usufruire di alcune funzionalità del sito. Si può giungere alla pagina di login cliccando sul bottone con l'icona del profilo nell'header oppure se si cerca di utilizzare una funzionalità riservata ad una categoria di utenti, ad esempio se si cerca di aggiungere un elemento al carrello. Al termine dell'accesso si viene riportati direttamente alla pagina che si cercava di utilizzare.

Se non si hanno i permessi viene mostrata una pagina di «accesso negato», ma durante un normale utilizzo del sito non capiterà mai...

### 2.7.1 Login

E' la pagina mostrata di default quando è necessario l'accesso. Se si dispone di un account si inseriscono email e password e si preme il bottone «accedi», altrimenti si clicca su «registra un nuovo account».



LOGIN

Nome utente:

Password:

Accedi

[Registra un nuovo account](#)

### 2.7.2 Registrazione

La registrazione è riservata ai nuovi utenti. Per la registrazione sono necessari: nome, cognome, email, password, telefono, indirizzo e codice fiscale.

Si inseriscono i dati e si preme il bottone «registrati».

**Attenzione:** devono essere rispettati determinati requisiti per l'inserimento dei dati, in particolare:

- L'email deve essere valida, ovvero contenere solo caratteri alfanumerici, punti, trattini, con una @ all'interno
- La password deve contenere almeno 8 caratteri, tra i quali sono ammessi
  - lettere maiuscole: A–Z
  - lettere minuscole: a–z
  - numeri: 0–9
  - caratteri speciali: !£\$%&()=?^,.;:\_|

- Il telefono deve contenere il prefisso e solo cifre senza spazi (es. +3907731234567)
- L'indirizzo deve essere nel formato VIA XXX NN, CITTA, STATO (es. Via Doria 13, Latina, Italy)
- Il codice fiscale deve essere nel formato standard italiano, cioè XXXXXXNNXNNXNNNX, dove X sono lettere e N numeri.

**REGISTRAZIONE**

Nome:

Cognome:

Email:

Password:

Conferma password:

Telefono:

Indirizzo:

Codice fiscale:

**Registrati!**

[Accedi con un account esistente](#)

## 2.8 Profilo

Ogni cliente dispone di una pagina «profilo» personale, da cui è possibile visionare i propri dati anagrafici, la reputazione acquisita e il credito. Vi si accede cliccando sull'icona del profilo nell'header.

Cliccando su «modifica profilo» alcuni dei campi prima solo testuali diventano modificabili, ed è possibile rettificare i propri dati. Per modificare la password, è sufficiente inserire la nuova nel campo, altrimenti lasciandolo vuoto non viene modificata.

Al termine, si preme il bottone «conferma modifiche» per salvare i nuovi dati.

**Attenzione:** è necessario che i campi continuino a rispettare i requisiti in fase di registrazione ([link](#)), altrimenti i campi non validi non vengono modificati.



## Profilo

<b>Nome:</b>	Mario
<b>Cognome:</b>	Rossi
<b>Email:</b>	mario.rossi@gmail.com
<b>Password:</b>	—
<b>Telefono:</b>	+390773123456
<b>Indirizzo:</b>	Via Isonzo 104, Latina, Italia
<b>Codice fiscale:</b>	RSSMRA83P12E472K
<b>Credito:</b>	1,000.00 € <a href="#">Ricarica</a>
<b>Reputazione:</b>	100 punti
<b>Data registrazione:</b>	27 October 2022
<b>Storico acquisti:</b>	<a href="#">Visualizza</a>

[Modifica profilo](#)

[Logout](#)

## Ricarica credito

Si giunge a questa pagina cliccando su «ricarica» nella pagina del proprio profilo, oppure dal carrello se si cerca di proseguire l'ordine con un credito insufficiente.

Per effettuare una ricarica, si effettua il pagamento della cifra che si vuole ricaricare ad uno dei proprietari. Successivamente si inserisce la stessa quantità nella pagina e si clicca su «invia richiesta». Un admin controllerà il pagamento effettuato e procederà all'accredito sul profilo.

### Importo da ricaricare:

---

[Invia richiesta](#)

[Torna indietro](#)

## Storico ordini

In questa pagina vengono riepilogati gli acquisti fatti dal cliente, in ordine cronologico inverso (dal più recente al meno recente), con data, indirizzo, prezzo pagato e un elenco degli articoli. Per ognuno è specificata la quantità, il nome e cliccando sul nome si va alla pagina dell'articolo.

## ORDINI

Data e ora	Indirizzo	Prezzo pagato	Articoli
2023-10-01 17:20	Via Doria, Latina, Italy	93.00 €	10x <a href="#">Isolate-pro grass-fed 700g</a>
2023-08-23 17:14	Via Doria, Latina, Italy	126.00 €	3x <a href="#">Hydro Purebar 55g</a> 1x <a href="#">Omega Pure</a> 10x <a href="#">Vitamina B-Complex - 100 Cps</a> 8x <a href="#">Isolate-pro grass-fed 700g</a>
2022-09-18 02:18	Via Doria, Latina, Italy	183.00 €	6x <a href="#">Vitamina B-Complex - 100 Cps</a>
2022-03-02 10:17	Via Doria, Latina, Italy	127.00 €	10x <a href="#">100% Whey 2,28kg</a> 3x <a href="#">Hydro Pure Professional 900 g</a>
2021-03-09 03:32	Via Doria, Latina, Italy	108.00 €	7x <a href="#">CASEIN Pure 900 g</a> 9x <a href="#">Vitamin pure 100cps</a> 9x <a href="#">Grenade Carb Killa 60g</a>
2021-02-14 06:18	Via Doria, Latina, Italy	114.00 €	8x <a href="#">CarbControl Bar 100g</a> 10x <a href="#">Borraccia da 3.9 Litri</a> 2x <a href="#">Vitamin pure 100cps</a> 1x <a href="#">Pantaloncini in cotone MP Tempo da uomo - Neri</a> 6x <a href="#">ISOLAT 908g - 100% Proteine Isolate Isolac®</a>

## 2.9 Carrello e ordine

Al carrello si arriva quando si aggiunge un prodotto dalla sua pagina di dettaglio oppure cliccando sull'icona nell'header.

Gli elementi attualmente in carrello sono disposti in elenco, e per ognuno è specificato il nome, il costo, la quantità e le operazioni di modifica ed eliminazione.

Per modificare la quantità di un prodotto si aggiusta il valore mostrato e si clicca sull'icona della matita, mentre per l'eliminazione è sufficiente cliccare sul cestino.

Se è applicato uno sconto il prezzo originale dell'articolo è barrato e viene specificato il prezzo effettivo, mentre nella riga di sotto viene specificato l'ammontare dello sconto ed un eventuale bonus in crediti che dà l'acquistare quel prodotto.

Dopo l'elenco viene specificato il prezzo finale dell'ordine e i bonus totali.

Per procedere con l'ordine si clicca su termina acquisto. Se si hanno crediti sufficienti si passa alla schermata di scelta dell'indirizzo di spedizione, altrimenti viene visualizzato il messaggio «credito insufficiente» con il tasto per navigare alla pagina di ricarica del credito.

CARRELLO

100% Whey 2,28kg, 100.99 €

2

Iso Whey Zero, 49.87 €

2

Mars protein 57g, 2.99 €

1

BLP99.9 8:1:1 Pure Professional, ~~39.95 €~~ 31.96 €

1

Sconto applicato: 20%

Bonus: +3 crediti

Totale: 336.67€

Bonus: +3 crediti

Indietro

Termina acquisto

## Scelta dell'indirizzo

E' presente il campo per inserire l'indirizzo di spedizione, precompilato con quello scelto in fase di registrazione e modificabile.

**Attenzione:** l'indirizzo deve essere nel formato VIA XXX NN, CITTA, STATO (es. Via Doria 13, Latina, Italy)

Cliccando su «conferma ordine» si invia l'ordine al negozio e si viene riportati al catalogo.

ORDINE

Indirizzo di spedizione:

Via Isonzo 104, Latina, Italia

Conferma ordine

## 2.10 Dashboard di gestione

Il gestore ha a disposizione una dashboard con dei link alle operazioni di gestione che esso può eseguire.

In fondo alla pagina inoltre è disponibile il bottone per effettuare il logout.



### 2.10.1 Gestione dei prodotti

La pagina è un riepilogo dei prodotti attualmente in catalogo. Sono ordinati per categoria e per ognuno sono specificati marca, costo, categoria, quantità e si hanno a disposizione tre azioni:

- Visualizza: apre la pagina del prodotto come vista dall'utente ([link](#)), utile per gestire recensioni, domande e risposte
- Modifica: apre una pagina di modifica per modificare gli attributi del prodotto
- Elimina: elimina il prodotto dal catalogo

Per aggiungere un prodotto è presente in fondo alla tabella il bottone «aggiungi prodotto», cliccando sul quale si apre la pagina di aggiunta di un prodotto.

La quantità è il numero di elementi di quel prodotto attualmente disponibili in magazzino. Viene scalata automaticamente quando viene effettuato un ordine ma il gestore deve assicurarsi che sia coerente con le scorte effettive.

Nel caso in cui la quantità di un prodotto sia zero, viene segnalato attraverso un'icona nella riga del prodotto a sinistra, in modo da aiutare il gestore ad ordinare ai fornitori i prodotti mancanti.

GESTIONE PRODOTTI							
Marca	Nome	Costo	Categoria	Quantita	Visualizza	Modifica	Elimina
Tsunami Nutrition	Cappello Tsunami Official	49.99 €	Abbigliamento	15			
Tsunami Nutrition	Leggings official donna	50.00 €	Abbigliamento	8			
Myprotein	Pantaloncini in cotone MP Tempo da uomo - Neri	19.99 €	Abbigliamento	25			
Myprotein	T-shirt oversize in cotone MP Tempo da uomo - Grigio Slate	25.99 €	Abbigliamento	12			

### Pagina di aggiunta/modifica prodotto

In questa pagina sono presenti delle caselle per inserire o modificare gli attributi di un prodotto: immagine, marca, nome, descrizione, costo, categoria, quantità.

Devono essere rispettati dei vincoli per l'inserimento, ad esempio:

- Il costo deve essere di almeno 0.01 €
- La categoria è vincolata a quelle già definite. Per aggiungerne una nuova si usa la pagina apposita (link).
- La quantità deve essere  $\geq 0$

Al termine dell'inserimento, si preme «aggiungi» o «modifica» in fondo alla pagina per applicare le modifiche.

GESTIONE PRODOTTO

Immagine:

Scegli file | Nessun file selezionato

Marca:

Tsunami Nutrition

Nome:

Cappello Tsunami Official

Cappello da baseball con visiera e stampe 3D del logo ufficiale Tsunami Nutrition.

Descrizione:

Costo:

49.99

Categoria:

Abbigliamento

Quantita:

15

### 2.10.2 Gestione delle offerte

In questa pagina è disponibile un riepilogo delle offerte attualmente attive. Per ognuna è specificato il tipo (se sconto o bonus), la quantità (percentuale o numero di crediti aggiunti), il target e i parametri (vedi dopo). Ognuna si può modificare o eliminare con i tasti appositi a destra.

L'offerta ha un target, cioè la categoria di utenti o prodotti a cui è destinata. E' la specifica situazione in cui si "attiva" l'offerta. I parametri X e Y sono caratteristici del target e lo modificano.

I target disponibili, con i loro parametri, sono:

- Spesi crediti X da data Y  
offerta destinata ad utenti che hanno speso X crediti da una certa data Y
- Spesi crediti X dalla registrazione  
offerta destinata ad utenti che hanno speso X crediti dalla registrazione
- Reputazione  $\geq$  X  
offerta destinata ad utenti che hanno una reputazione  $\geq$  X
- Cliente da X anni  
offerta destinata ad utenti registrati da almeno X anni
- Prodotto == X  
offerta associata al prodotto X
- Categoria == X  
offerta associata alla categoria X
- Quantità di prodotto X  $\geq$  Y  
offerta associata al prodotto X solo se la quantità in magazzino è  $\geq$  Y

Per aggiungere una nuova offerta si preme «aggiungi offerta» in fondo alla pagina.

GESTIONE OFFERTE						
Tipo	Quantità	Target	X	Y	Modifica	Elimina
sconto	20%	spesi crediti X da data Y	100.00	2023-05-15		
bonus	+3	quantità di prodotto X $\geq$ Y	BLP99.9 8:1:1 Pure Professional	100		
sconto	20%	quantità di prodotto X $\geq$ Y	BLP99.9 8:1:1 Pure Professional	100		
<div>Aggiungi offerta</div> <div>Torna indietro</div>						

### Pagina di aggiunta/modifica

Sono disponibili i campi necessari per la definizione degli attributi dell'offerta.

Il primo campo è il tipo di offerta (se sconto o bonus) e modifica il campo successivo. Se è uno sconto viene chiesta la percentuale mentre se è bonus il numero di crediti da aggiungere.

Successivamente viene chiesto il target, e in base alla selezione vengono attivati i campi per la definizione dei parametri necessari, come spiegato sopra.

Al termine dell'inserimento, si preme «aggiungi» o «modifica» in fondo alla pagina per applicare le modifiche.

## GESTIONE OFFERTA

**Tipo di offerta:** Sconto ▼

**Percentuale:**  %

**Target:** Crediti spesi da una determinata data ▼






**Crediti spesi:**

**Data di inizio:**  

### 2.10.3 Gestione delle categorie

La pagina di gestione delle categorie fornisce un elenco delle categorie presenti nel sistema e per ognuna il tasto per eliminarla.

Per aggiungerne una nuova si preme «aggiungi categoria» in fondo alla pagina.

GESTIONE CATEGORIE	
Nome	Elimina
Proteine in polvere	
Barrette proteiche	
Vitamine	
Abbigliamento	
Accessori	
<div>Aggiungi categoria</div> <div>Torna indietro</div>	

Pagina di aggiunta

Viene chiesto il nome della categoria da aggiungere e successivamente si preme il bottone «aggiungi».

## AGGIUNGI CATEGORIA

**Nome:** \_\_\_\_\_



**Aggiungi**

**Torna indietro**

### 2.10.4 Consultazione clienti

In questa pagina vengono elencati i clienti attualmente registrati nel sistema, con nome, cognome, reputazione e un tasto per visionare lo storico degli ordini del cliente.

Cliccando sull'icona dell'occhio si apre la pagina dello storico del cliente, uguale a quella che può vedere quest'ultimo dal suo profilo ([link](#)).

CONSULTAZIONE CLIENTI			
Nome	Cognome	Reputazione	Storico
Mario	Rossi	100	
Marco	Verdi	100	

### 2.11 Dashboard di amministrazione

L'amministratore ha a disposizione una dashboard con dei link alle operazioni di amministrazione che esso può eseguire.

In fondo alla pagina inoltre è disponibile il bottone per effettuare il logout.





### 2.11.1 Gestione utenti

Nella pagina di gestione degli utenti sono elencati gli utenti registrati nel sistema, in ordine di registrazione. Per ognuno vi sono specificati i suoi dati anagrafici, email, e il suo stato (se attivo o bannato).

Per modificare un utente, si clicca sull'icona della matita a destra della riga. A questo punto i campi diventano modificabili e si aggiunge un campo per la modifica della password e una checkbox per definire se l'utente è attivo oppure no.

Per modificare la password è sufficiente inserirne una nuova, mentre se si lascia il campo vuoto non viene modificata.

Al termine delle modifiche si preme il bottone con la spunta verde per applicarle.

**Attenzione:** è necessario che i campi continuino a rispettare i requisiti in fase di registrazione ([link](#)), altrimenti i campi non validi non vengono modificati.

**Suggerimento:** amministratori e gestori possono modificare i propri dati di accesso da questa tabella.

GESTIONE UTENTI									
Tipo	Email	Nome	Cognome	Telefono	Indirizzo	Codice fiscale	Password	Attivo	Modifica
cliente	mario.rossi@gmail.com	Mario	Rossi	+390773123456	Via Isonzo 104, Latina, Italia	RSSMRA83P12E472K	–	<input checked="" type="checkbox"/>	
gestore	luigi.bianchi@gmail.com	Luigi	Bianchi				–	<input type="checkbox"/>	
admin	francesco.marroni@gmail.com	Francesco	Marroni				–	<input type="checkbox"/>	
cliente	marco.verdi@gmail.com	Marco	Verdi	+390773123456	Via Isonzo, 104, Latina, Italia	RSSMRA83P12E472K	–	<input checked="" type="checkbox"/>	
<div>Torna indietro</div>									

<immagine tabella con modifica>

### 2.11.2 Gestione accrediti

In questa pagina vi sono elencate le richieste di accredito effettuate dagli utenti, e per ognuna data e ora, l'utente associato, la quantità da ricaricare e due tasti di azione:

- Accetta: aggiunge la quantità specificata all'utente, sotto forma di crediti
- Rifiuta: elimina la richiesta di accredito dal sistema

Si assume che l'admin controlli se il pagamento è stato effettuato dall'utente, per esempio controllando il proprio conto corrente, e poi decida l'azione da effettuare.





GESTIONE ACCREDITI				
Data	Utente	Quantita	Accetta	Rifiuta
2023-07-10 17:29	Marco Verdi	20.00 €	✓	✗
2023-08-14 10:23	Mario Rossi	150.00 €	✓	✗

Torna indietro

### 2.12 Gestione FAQ

Nella pagina delle FAQ, un gestore o amministratore ha a disposizione anche le seguenti azioni:

- Eliminare una domanda, cliccando sull'icona del cestino sotto di essa
- Modificare una domanda, cliccando sull'icona della matita sotto di essa.
- Aggiungere una domanda, elevando una già esistente dalle domande/risposte di un prodotto ([link](#)) oppure aggiungendone una da zero.

FAQ	
Quanto costa la spedizione?	⇓
	
Quanto costa la spedizione?	⇓
	
Aggiungi FAQ	

## Pagina di aggiunta/modifica

E' costituita da 2 campi testuali, uno per la domanda e uno per la risposta. Nel caso di elevazione o di modifica di una domanda esistente vengono precompilati con i valori corrispondenti. Si possono modificare a proprio piacimento e al termine si clicca su «aggiungi» (o «modifica») per eseguire l'operazione. Si viene reindirizzati alla pagina delle FAQ da cui si può vedere il risultato dell'operazione.

The image shows two side-by-side web forms for managing FAQs. The left form is titled "MODIFICA FAQ" and the right is "AGGIUNGI FAQ". Both forms have a "Domanda:" section with a text input field and a "Risposta:" section with a text input field. The left form has a "Modifica" button and a "Torna indietro" button. The right form has an "Aggiungi" button and a "Torna indietro" button.

## 2.13 Installazione

E' necessario solo copiare il contenuto della cartella www nella cartella di destinazione, per esempio la root del web server.

Il sito non necessita di ulteriori configurazioni, in quanto individua da solo il percorso della root e la sottodirectory del web server in cui è installato.

## 3. Programmazione

Il sito è posizionato nel repository del progetto nella cartella `www`.

Al suo interno troviamo diverse cartelle:

- `data`: contiene i documenti XML con i dati del sito e gli schemi dei suddetti documenti, in formato XSD e DTD
- `lib`: contiene funzioni utilizzate dalle varie pagine del sito, per esempio per accedere e manipolare i dati oppure redirect, controllo accessi, ecc.
- `res`: contiene le risorse utilizzate dalle pagine, in particolare:
  - `css`: contiene i file CSS, ognuno dedicato ad una pagina in particolare, ad eccezione di `common.css`, che contiene classi utili per tutte le pagine
  - `img`: contiene le immagini, in particolare in `prodotti` contiene le immagini di ogni prodotto nominate in base al suo id
  - `js`: contiene gli script JS, anch'essi nominati come la pagina che lo utilizza
- `admin`, `cliente`, `gestore`: contengono le pagine dedicate ad una tipologia di utente. Siccome alcune operazioni sono comuni a `gestore` ed `admin` per esempio, non è stato possibile separarle in modo netto.
- `<pagine nella root>`: sono pagine a cui può accedere anche un utente non registrato, ad esempio `prodotto.php`.
- `favicon.ico`: l'icona che viene mostrata vicino al titolo della scheda del browser.

### 3.1 Struttura delle pagine

Ogni pagina inizia con:

```
$perm_visitatore = true/false;
$perm_cliente = true/false;
$perm_gestore = true/false;
$perm_admin = true/false;

$rc_level = N;
require_once('../lib/start.php');
```

Vengono definite delle variabili che verranno utilizzate in `lib/start.php`:

- `$perm_visitatore` è true se un visitatore ha il permesso di accedere a quella pagina, altrimenti false. Allo stesso modo `$perm_cliente`, `$perm_gestore`, `$perm_admin`.

- `$src_level` definisce il livello di sottodirectory in cui è posizionata la pagina rispetto alla root del sito. Per esempio, `admin/utenti.php` è solo in una sottodirectory e quindi si imposta `$src_level = 1`, oppure `prodotto.php` non è in una sottodirectory quindi si imposta `$src_level = 0`.

Successivamente viene caricato `lib/start.php` attraverso un `require_once()` e si occupa di effettuare operazioni comuni a tutte le pagine ([link](#))

A questo punto vengono caricati altri script se necessari, per esempio per la gestione degli utenti, ed effettuate altre operazioni caratteristiche della pagina.

Viene iniziato a stampare il documento HTML, anche qui la parte iniziale è simile a tutte le pagine:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="it" lang="it">
<head>
  <title>TITOLO &ndash; R&amp;C store</title>

  <link rel="stylesheet" type="text/css" href="<?php echo($src_subdir);
?>/res/css/common.css" />
  <link rel="stylesheet" type="text/css" href="<?php echo($src_subdir);
?>/res/css/header.css" />
  <link rel="stylesheet" type="text/css" href="<?php echo($src_subdir);
?>/res/css/footer.css" />
  <link rel="stylesheet" type="text/css" href="<?php echo($src_subdir);
?>/res/css/PAGINA.css" />
</head>
```

E' presente l'intestazione XML e l'head del documento, in cui viene definito il titolo della pagina e vengono caricati i file CSS:

- `common.css`: comune a tutte le pagine (`common.css`)
- `header.css`, `footer.css`: lo stile dell'header, del footer
- `PAGINA.css`: lo stile specifico della pagina attuale

Segue il body del documento, contenente due `require` e un `div` che racchiude il contenuto caratteristico della pagina.

```
<body>
  <?php require($src_root . '/lib/header.php'); ?>
  <div id="contenuto">
    <!-- ... -->
  </div>
  <?php require($src_root . '/lib/footer.php'); ?>
</body>

</html>
```

I `require` si occupano di stampare il contenuto dell'header e del footer, evitando così di riscriverlo in ogni pagina.

## 3.2 Cartella lib

### 3.2.1 lib/start.php

Questo script si occupa di:

- Definire la variabile `$rc_root`, che contiene il percorso assoluto della root del sito sul file system.

```
$rc_root = realpath(__DIR__ . '/../..');
```

E' necessaria per facilitare i successivi `require_once()`, in quanto altrimenti sono relativi alla directory in cui è contenuto lo script.

Viene ricavata da `__DIR__`, che contiene la directory dello script corrente, in questo caso la cartella `lib`, a cui si aggiunge `../..` per tornare indietro nel percorso di una cartella. Il valore risultante viene passato a `realpath()` per semplificare il percorso risultante (es. da `/srv/www/rc-store/lib/..` diventa `/srv/www/rc-store`)

- Definire la variabile `$rc_subdir`, che contiene la sottodirectory in cui è contenuto il sito, rispetto alla root del webserver (es. in <http://10.173.0.1/rc-store/cliente/storico.php> sarebbe `/rc-store`).

E' necessario anche qui per facilitare il riferimento alle risorse, senza utilizzare i link relativi.

Viene ricavato dalla funzione `make_subdir()`, spiegata successivamente.

- Inizializzare la sessione, utilizzando `session_start()`. Viene creata una nuova se è un utente nuovo per esempio oppure ripresa una esistente se l'utente ha già visitato il sito
- Controllare i permessi utilizzando le variabili definite dalla pagina, ed effettuare redirect al login se non si è loggati, oppure se si è loggati ma non si hanno i permessi sufficienti si viene reindirizzati ad una pagina di "accesso negato"
- Gestire un caso particolare nell'aggiunta di un elemento al carrello ([spiegato successivamente](#))

### 3.2.2 lib/utls.php

In questo script sono definite due funzioni utilizzate frequentemente:

- `make_subdir()`: restituisce la sottodirectory del sito rispetto alla root del web server, a partire dal numero di sottodirectory in cui è posizionata la pagina chiamante.

Quello che fa la funzione è prelevare la variabile `$_SERVER[ 'PHP_SELF' ]`, che contiene il percorso dello script corrente e con un ciclo for togliere:

- dall'ultimo / in poi, che contiene il nome dello script
- tanti / quanto specifica il parametro `$level`, ovvero il numero di sottodirectory

Per esempio, chiamando la funzione dalla pagina `/rc-store/admin/utenti.php` passando 1 come parametro, si ottiene `/rc-store`, perché viene tolto `/utenti.php` e dopo `/admin` (un solo /).

- `redirect()`: Effettua un redirect con il codice HTTP specificato, alla pagina specificata, eventualmente aggiungendo il parametro GET `redirect` e ferma l'esecuzione dello script.

In particolare:

- `$_SERVER[ 'REQUEST_URI' ]` contiene l'URL che il browser ha utilizzato nella richiesta per accedere alla pagina corrente
- `http_response_code()` invia il codice specificato come codice HTTP della risposta (es. 307 redirect temporaneo, 401 non loggato, 403 non autorizzato)
- `header( 'Location: X' )` invia un header HTTP nella risposta per far sì che il browser segua l'URL specificato

E' necessario che questa funzione venga chiamata prima che venga stampata qualsiasi cosa nella risposta, perché agisce sugli header e devono essere inviati per primi.

Il parametro `redirect` viene utilizzato da pagine come login o registrazione per tornare alla pagina che ha richiesto l'accesso dopo che è stato effettuato.

### 3.2.3 lib/xml.php

Contiene funzioni comuni per la gestione dei file XML.

- `load_xml()`: prende come parametro il nome del documento XML da caricare e restituisce il `DOMDocument()` corrispondente.

Viene caricato il file nella cartella `/data` come array di linee, vengono ripulite dai caratteri di spaziatura e concatenate fra loro con un ciclo `for`.

Successivamente viene creato l'oggetto `DOMDocument()` e viene passata la stringa con i dati del documento al suo metodo `loadXML()` per popolare il documento in memoria.

- `save_xml()`: prende come parametro un `DOMDocument()` e lo salva sul file corrispondente nella cartella `/data`.

Viene utilizzato il metodo `save()` di `DOMDocument()` per salvare il documento in memoria su file.

Prima del salvataggio vengono impostati gli attributi `formatOutput = true` e `preserveWhiteSpace = false` per ottenere un output formattato nel file XML.

- `domlist_to_array()`: converte una lista di elementi, come restituita da `childNodes` per esempio, in un array vero e proprio.

E' necessaria per la funzione di ordinamento, perché viene utilizzato `usort()` che effettua l'ordinamento solo su array.

- `sort_by_element_dec()`: prende come parametro un array di elementi, il tag dell'elemento da utilizzare per l'ordinamento e un terzo parametro che è impostato a `false` per ottenere un ordine ascendente, a `true` per un ordine discendente.

Viene utilizzata la funzione `usort()`, a cui viene passato l'array di elementi e una funzione di confronto. Alla funzione di confronto vengono passati i due elementi da confrontare, ma non è possibile un confronto diretto perché sono elementi XML. Con `use()` gli è permesso utilizzare le variabili `$key` e `$desc` esterne alla funzione.

La funzione estrae i valori dall'interno degli elementi:

```
$aValue = $aElement->getElementsByTagName($key)[0]->textContent;
```

Li converte a float con `floatval()` e restituisce il risultato dell'operatore `<=>`, chiamato "spaceship operator". Esso restituisce zero se gli elementi sono uguali, +1 se è maggiore l'elemento a sinistra e -1 se è maggiore l'elemento a destra.

La funzione di ordinamento quindi utilizza questo risultato per disporre gli elementi.

Se il parametro `$desc` è `true` i due lati dell'espressione vengono invertiti in modo da invertire anche l'ordinamento.



- `sort_by_element_txt()`: è analoga a `sort_by_element_dec()` ma considera il valore dell'elemento da ordinare come una stringa anziché un decimale. L'unica differenza con la precedente è che al posto di `<=>`, che confronta decimali, utilizza "string natural compare", `strnatcmp()`. Essa restituisce gli stessi valori di confronto dello spaceship operator, ma confrontando le stringhe considerando l'ordine alfanumerico dei singoli caratteri.
- `get_next_id()`: prende come parametro una lista di elementi, ognuno dotato dell'attributo `id`, e restituisce il primo `id` maggiore di quelli già utilizzati. E' necessaria per l'inserimento dei dati, in quanto spesso nelle strutture dati del progetto vi sono degli `id` per identificare univocamente gli elementi all'interno di una lista.
- `xpath()`: prende come parametri un `DOMDocument()`, il nome del documento e la query XPath da eseguire sul documento, restituendo un array con il risultato della query.  
Viene creato un nuovo oggetto `$xp` di tipo `DOMXPath()` passando il documento al costruttore.  
E' necessario ogni volta crearne uno nuovo perché tra un'esecuzione e l'altra di `xpath()` alcune volte viene aggiornato il documento da operazioni effettuate nella pagina.  
Successivamente viene registrato il namespace del documento come `ns`, ed è necessario inserirlo come prefisso ad ogni elemento nella query XPath.  
Il namespace per ogni documento è della forma '<http://www.lweb.uni/tesina-rcstore/><nome>/' , per questo non è necessario passare la stringa completa come parametro alla nostra funzione, ma solo il nome del documento.  
Viene eseguita la query, chiamando il metodo `evaluate()` di `$xp` passando come parametro la query e viene restituito il risultato.

### 3.2.4 lib/faq.php (e altri)

Ogni file XML ha il suo corrispondente script PHP in `./lib/`, che si occupa delle funzionalità dell'applicazione che lo riguardano. Per questo viene illustrato in dettaglio solo `faq.php`, e degli altri vengono spiegate solo funzionalità particolari.

All'esecuzione dello script (quando viene effettuato il suo require da un altro script) viene caricato il documento XML 'faq' con la funzione `load_xml()` passandogli il nome del documento e il risultato viene assegnato alla variabile globale `$doc_faq`, utilizzata in questo script e in alcuni accessi in lettura al documento fuori dalla pagina.

Ogni funzione avrà globale `$doc_faq` per poter utilizzare la variabile globale.

Le funzioni disponibili sono:

- `aggiungi_faq()`: prende come parametri le stringhe domanda e risposta e aggiunge l'elemento alla root del documento 'faq'

Viene ottenuta l'elemento root del documento (in questo caso l'elemento 'faqs') con `$doc_faq->documentElement`, dopodichè viene ricavato da questo elemento l'array di figli (gli elementi 'faq') con `$root->childNodes` per passarlo a `get_next_id()`, che restituisce l'id dell'elemento da aggiungere.

Viene creato l'elemento 'faq' con `$doc_faq->createElement('faq')` e impostato l'attributo id a `$id` con `$faq->setAttribute('id', $id)`.

Viene creato anche l'elemento 'domanda', con `$doc_faq->createElement('domanda', $domanda)`, ma qui viene passato un secondo parametro che costituirà il contenuto testuale del nodo (il `textContent`).

Allo stesso modo viene creato l'elemento 'risposta' e memorizzato in `$risposta`.

Con `$faq->appendChild($domanda/risposta)` e `$root->appendChild($faq)` vengono rispettivamente inseriti `$domanda` e `$risposta` come figli di `$faq` e `$faq` come figlio di 'faqs', ovvero la root del documento ottenuta prima.

Viene successivamente salvato il documento con `save_xml()` e per un bug di `xpath` (paragrafo 3.6), viene anche ricaricato con `$doc_faq = load_xml('faq')`

- `elimina_faq()`: prende come parametro l'id della faq ed elimina l'elemento corrispondente dal documento XML.

Internamente utilizza `xpath()` su `$doc_faq`, con la query:

```
/ns:faqs/ns:faq[@id=' $id ']
```

Cerca, a partire dalla root, un elemento di tipo 'faqs', con all'interno un elemento di tipo 'faq' con attributo `id = $id`.

Viene estratto il nodo con `$result[0]`, perchè `xpath()` restituisce sempre un array di elementi, anche se ve n'è uno solo.

A questo punto viene ottenuto il suo nodo genitore (cioè 'faqs') utilizzando parentNode e memorizzato in \$faqs.

Viene chiamato il metodo removeChild() di \$faqs passando \$faq, ovvero l'elemento da rimuovere, e poi viene salvato il documento modificato con save\_xml(), passando il documento \$doc\_faq e il nome del documento.

- `modifica_faq()`: prende come parametro l'id della faq da modificare e le stringhe con la nuova domanda e risposta.

Utilizza la stessa query XPath descritta sopra, e poi per la modifica si utilizza:

```
$faq->getElementsByTagName('domanda')[0]->textContent = $domanda;
```

Viene cercato in \$faq un elemento di tipo 'domanda', viene selezionato il primo (l'unico in questo caso), e viene modificato il suo textContent assegnandogli \$domanda.

Allo stesso modo si modifica l'elemento 'risposta' e viene salvato il documento modificato con save\_xml().

### 3.2.5 lib/categorie.php

Sono presenti le funzioni:

- `ottieni_categoria()`: prende come parametro l'id della categoria, e, se trovata, restituisce il suo nome, altrimenti restituisce <categoria sconosciuta>.
- `elimina_categoria()`: elimina la categoria con l'id, passato come parametro, dall'elenco categorie (vedi `elimina_faq`)
- `aggiungi_categoria()`: aggiunge una categoria all'elenco categorie, il cui nome è passato come parametro (vedi `aggiungi_faq`)
- `ordina_per_categoria()`: prende come parametri un array di prodotti, true/false per ordinamento ascendente/discendente e li ordina in base alla categoria.

Internamente è molto simile a `sort_by_element_txt()`, con l'unica differenza che dopo aver estratto il contenuto dell'elemento categoria del prodotto, viene chiamata la funzione `ottieni_categoria()` per ottenere il nome della categoria dal suo id, e ordinare in base a quello.

### 3.2.6 lib/constants.php

Contiene le espressioni regolari utilizzate per la validazione dei campi: codice fiscale, email, indirizzo, password, numero di telefono

## 3.3 Funzionalità particolari

### 3.3.1 Upload dell'immagine del prodotto

Nella pagina di aggiunta/modifica di un prodotto, `./gestore/prodotto.php`, è presente un campo per effettuare l'upload di un'immagine associata al prodotto.

Nel form contenente i campi con i dati del prodotto, è presente l'attributo `enctype="multipart/form-data"`, per far sì che i dati nella richiesta POST HTTP vengano mandati come form multipart e possano essere inviati anche dati binari come le immagini appunto.

Il campo per l'upload dell'immagine è un input di tipo file che accetta solo file di tipo PNG (`image/png`)

```
<input type="file" name="immagine" accept="image/png" />
```

All'invio del POST, viene chiamata la funzione `aggiungi_prodotto()`, oppure `modifica_prodotto()`, passando il campo immagine dell'array `$_FILES`, che contiene le informazioni sull'upload del file.

Nelle funzioni `aggiungi_prodotto()` e `modifica_prodotto()`, è presente questo codice per gestire l'upload:

```
$upload_ok = isset($immagine['error']) &&
    $immagine['error'] === UPLOAD_ERR_OK;
if ($upload_ok) {
    $nome_temp = $immagine['tmp_name'];
    $nome_finale = $rc_root . '/res/img/prodotti/' . $p_id .
        '.png';
    move_uploaded_file($nome_temp, $nome_finale);
}
```

In pratica, se è stato effettuato l'upload di un file ed è andato a buon fine (il campo `error` è `UPLOAD_ERR_OK`), viene preso il nome del file temporaneo contenente l'immagine (il campo `tmp_name`) e viene costruito il percorso di destinazione del file. Viene chiamata la funzione `move_uploaded_file()` che si occupa di spostare il file caricato nel nuovo percorso ma anche di controllare se il nome temporaneo (il primo parametro) è effettivamente di un file caricato, e non di un file già esistente sul filesystem, in modo da evitare problemi di sicurezza.

### 3.3.2 Aggiunta al carrello da non loggato

Quando l'utente non è loggato, ha comunque abilitato il tasto «aggiungi al carrello», ma per effettuare l'operazione è necessario aver effettuato l'accesso.

Alla pressione del tasto, viene effettuato un POST a `./cliente/carrello.php`, ma non essendo loggato non ha i permessi per accedere al carrello, quindi viene reindirizzato da `./lib/start.php` alla pagina di login, perdendo i dati del POST.

Per tenere i dati sull'operazione di aggiunta, in `./lib/start.php` è stato inserito questo codice:

```
if ($_SERVER['PHP_SELF'] === $rc_subdir . '/cliente/carrello.php') {  
    if (isset($_POST['azione']) && $_POST['azione'] === 'aggiungi') {  
        $_SESSION['agg_carr_id_prod'] = $_POST['id_prodotto'];  
        $_SESSION['agg_carr_qta'] = $_POST['quantita'];  
    }  
}
```

Se la pagina è `./cliente/carrello.php` ed è stato effettuato un POST per aggiungere un prodotto al carrello, vengono memorizzati i dati temporaneamente nella sessione.

Siccome la sessione viene inizializzata in `./lib/start.php`, ma in questo file viene anche fatto il controllo dei permessi, non è stato possibile inserire questo codice né prima né dopo il `require()` di `./lib/start.php`, in quanto prima la sessione non è ancora inizializzata e il dopo, in caso di utente non loggato, non viene eseguito, perché la funzione `redirect()` effettua anche un `exit()`.

Il posto più logico dove inserire questo codice appunto è stato tra il `session_start()` e il controllo dei permessi.

Al termine dell'accesso, si viene reindirizzati a `./cliente/carrello.php` che si occupa di terminare l'aggiunta:

```
if (isset($_SESSION['agg_carr_id_prod'])) {  
    aggiungi_carrello($_SESSION['agg_carr_id_prod'], $_SESSION['agg_carr_qta']);  
    unset($_SESSION['agg_carr_id_prod']);  
    unset($_SESSION['agg_carr_qta']);  
}
```

### 3.3.3 Form di rating

Il form di rating nella pagina del prodotto, per valutare le recensioni, domande e risposte, è costituito da stelline cliccabili, ma il risultato viene inviato tramite POST.

Le stelline sono degli elementi di tipo `<a>` che hanno come contenuto testuale il carattere con l'icona della stella (vuota è `&#x2606` e piena è `&#x2605`).

Questo effetto è stato ottenuto disaccoppiando le file di stelline dal form di invio, e utilizzando JS per aggiornare dei campi che nel form sono "hidden".

Al click su una stellina, viene chiamata la funzione `setCampo()`, passando l'id del form corrispondente, il nome del campo da aggiornare e il suo valore.

In `setCampo()` viene aggiornato il valore del campo nel form, individuando il form con `document.forms[id]` e modificando il valore dell'elemento con `form.elements[campo].value = valore`.

Inoltre, vengono sostituiti i `textContent` degli elementi contenenti le stelline con i nuovi caratteri (es. cliccando sulla terza stellina vengono accese le prime tre e spente le ultime due).

Il bottone «invia» fa parte del form, e alla pressione invia il POST dei campi hidden.

### 3.3.4 Disposizione pagina prodotto (flex)

Nella pagina del prodotto è stata utilizzata molte volte `display:flex` per ottenere una divisione orizzontale dello spazio in base a delle proporzioni date.

Ad esempio, nella parte superiore della pagina, troviamo tre sezioni orizzontali: una con l'immagine del prodotto, una con le informazioni, una con quantità, costo, offerte e il tasto «aggiungi al carrello».

Anche nelle recensioni è stato utilizzato, con le sezioni data, rating e autore, contenuto.

Queste sezioni sono tre div contenuti in un div. Sul div esterno è stata applicata la classe CSS da noi creata con le seguenti proprietà:

```
.flex-row {  
  display: flex;  
  flex-direction: row;  
}
```

Queste due proprietà rispettivamente fanno sì che venga applicata la modalità di visualizzazione con "flexbox", e che agisca sulla dimensione orizzontale.

I div interni invece hanno la proprietà `flex-basis`, utilizzata per definire le proporzioni degli elementi, applicata mediante la classe `fb-X`:

```
.fb-X {  
  flex-basis: X%;  
}
```

Estratto del codice del box recensioni:

```
<div class="flex-row">  
  <div class="fb-12">  
    <!-- Data e tasto elimina -->  
  </div>  
  <div class="fb-20">  
    <!-- Form di rating -->  
  </div>  
  <div class="fb-68">  
    <!-- Contenuto -->  
  </div>
```

Le proporzioni sono scelte in modo che in un div con la classe `flex-row` la somma delle proporzioni in percentuale dei div interni sia 100% (la totalità dello spazio orizzontale a disposizione del div esterno).

### 3.3.5 Tasti «torna indietro»

Ad alcune pagine si giunge da diversi punti. Ad esempio, alla pagina cliente/ricarica.php si giunge dalla pagina del profilo del cliente e dal carrello se il credito è insufficiente.

In quel caso, il tasto «torna indietro» deve riportare alla pagina precedente corretta.

Abbiamo utilizzato in un primo momento la funzione JS `history.back()`, ma se si giungeva alla pagina con una richiesta POST il browser non riusciva a tornare correttamente alla pagina precedente.

Per questo è stata utilizzata la variabile `$_SERVER['HTTP_REFERER']`, che contiene il valore dell'header HTTP 'Referer' della richiesta, ovvero la pagina da cui si è giunti.

In assenza del suddetto header, si considera come pagina precedente quella più probabile, ovvero nella pagina di ricarica è cliente/profilo.php.

```
$ce_referer = isset($_SERVER['HTTP_REFERER']) && $_SERVER['HTTP_REFERER'] != '';  
if ($ce_referer) {  
    $pagina_prec = $_SERVER['HTTP_REFERER'];  
} else {  
    $pagina_prec = $rc_subdir . '/cliente/profilo.php';  
}
```

### 3.3.6 Scomparsa label e comparsa input

In alcune pagine (es. pagina profilo) sono presenti dei button i quali permettono la scomparsa dell'elemento html label e la comparsa di campi di input (es. per modificare i dati del profilo), questa funzionalità in primo luogo è implementata assegnando ad entrambi gli elementi una classe denominata «con-toggle» la quale non aggiunge delle funzionalità ma viene utilizzata per andare ad identificare gli elementi coinvolti in questa operazione di comparsa/scomparsa, in aggiunta all'elemento input, vi è applicata la classe denominata «nascosto» la quale non ne permette la visualizzazione a priori.

```
.nascosto {  
    display: none;  
}
```

Lo swap tra i due elementi è reso possibile grazie all'evento «onclick» presente su un bottone che una volta cliccato, manda in esecuzione una funzione JS, la quale preleva dal documento gli elementi con la classe «con-toggle» e vi applica il metodo `toggle("nascosto")` cosicché la classe «nascosto» verrà applicata agli elementi che non la presentano, mentre verrà rimossa agli elementi che già la presentano.

```
function visibilita() {  
    var elemInfo = document.getElementsByClassName('con-toggle');  
    for (var i = 0; i < elemInfo.length; i++) {  
        elemInfo[i].classList.toggle("nascosto");  
    }  
}
```

## 3.4 Bug riscontrati

### 3.4.1 Aggiunta elemento e poi xpath()

Abbiamo riscontrato un bug nell'utilizzo di `xpath()`. In pratica, se viene effettuata l'aggiunta di un elemento ad un altro con `appendChild()`, nelle chiamate successive ad `xpath()` che coinvolgono gli stessi nodi non viene restituita la lista con il nuovo elemento.

Invece con le operazioni di modifica ed eliminazione dei nodi sembra funzionare correttamente, ma viene ignorata l'aggiunta del nodo anche se è stata effettuata prima (vedi esempio).

E' stato isolato il codice che attiva questo bug ed è presente nel repo git a `varie/bug-xpath.php`, e il documento utilizzato è `www/data/prova.xml`.

Un esempio di esecuzione:

- Caricamento documento  
1: AAAAA => BBBBB  
2: CCCCC => DDDDD
- Aggiunta elemento 3 (EEEEEE => FFFFF)  
1: AAAAA => BBBBB  
2: CCCCC => DDDDD
- Modifica elemento 2  
1: AAAAA => BBBBB  
2: GGGGG => HHHHH
- Eliminazione elemento 1  
2: GGGGG => HHHHH
- Normalizzazione documento  
2: GGGGG => HHHHH
- Ricarica documento  
2: GGGGG => HHHHH  
3: EEEEE => FFFFF

Come si evince, il nuovo elemento compare solo dopo aver ricaricato il documento da disco con la nostra funzione `load_xml()`.

Abbiamo provato, senza successo, a chiamare il metodo `normalizeDocument()` sull'oggetto `DOMDocument`, in quanto nella documentazione di PHP si dice:

«Questo metodo funziona come se avessi salvato e poi caricato il documento, mettendo il documento in un formato "normale"».

Inoltre, non usando `xpath()` ma `$doc->documentElement->childNodes` per ottenere la lista di nodi, viene restituita la lista aggiornata con il nuovo nodo, come ci si aspetta.

Non trovando altre informazioni online, abbiamo implementato un "workaround", ovvero abbiamo salvato e ricaricato il documento da disco dopo ogni aggiunta di un nodo.



### 3.4.2 Eliminazione di nodi in foreach()

Un altro bug riscontrato è nell'eliminazione dei nodi con un foreach(), ad esempio nella seguente funzione:

```
function elimina_offerte_prodotto($id_prodotto) {
    global $doc_offerte;
    $root = $doc_offerte->documentElement;
    $offerte = $root->childNodes;

    foreach ($offerte as $offerta) {
        $target = $offerta->getElementsByTagName('target')[0]->textContent;
        if ($target !== 'prodSpec' && $target !== 'eccMag') {
            continue;
        }

        $id_prod_off = $offerta->getElementsByTagName('idProdotto')[0]->textContent;
        if ($id_prod_off == $id_prodotto) {
            $root->removeChild($offerta);
        }
    }

    save_xml($doc_offerte, 'offerte');
    return true;
}
```

Se viene trovata un'offerta da eliminare, viene chiamato, dentro il ciclo, il metodo removeChild dell'elemento root del documento, passando l'elemento da eliminare. Questo fa sì che il ciclo smetta di seguire l'ordine corretto degli elementi, in particolare dopo l'iterazione che coinvolge l'elemento da eliminare non ne viene processato nessun'altro.

La soluzione che abbiamo pensato è quella di inserire i nodi da eliminare in un array separato ed eliminarli con un altro ciclo su quell'array.

```
function elimina_offerte_prodotto($id_prodotto) {
    // ...
    $da_eliminare = [];

    foreach ($offerte as $offerta) {
        // ...
        if ($id_prod_off == $id_prodotto) {
            array_push($da_eliminare, $offerta);
        }
    }

    foreach ($da_eliminare as $offerta) {
        $root->removeChild($offerta);
    }
    // ...
}
```

## 4. Elenco credenziali

Tipo	Email	Password	Note
Cliente	mario.rossi@gmail.com	password	–
Cliente	marco.verdi@gmail.com	password	–
Cliente	luca.gialli@gmail.com	password	–
Cliente	giorgio.santini@gmail.com	password	–
Cliente	federico.napoli@gmail.com	password	Non attivo
Cliente	rossana.filippi@gmail.com	password	Non attivo
Cliente	giulia.capasso@gmail.com	password	Non attivo
Cliente	sergio.rossini@gmail.com	password	Non attivo
Cliente	renato.monti@gmail.com	password	Non attivo
Cliente	ignazio.carta@gmail.com	password	Non attivo
Gestore	luigi.bianchi@gmail.com	password	–
Admin	francesco.marroni@gmail.com	password	–