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

***Data Persistence Document 1.0***

***05/12/2022***

****

0512107476 **– Salvatore Santoriello**0512107395 **– Bruno Farano**0512106126 **– Emanuele Milito**0512107080 **– Gianmichele Cancellaro**

***Revision History***

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versione** | **Descrizione** | **Autore** |
| 05/12/2022 | 1.0 | Prima stesura del documento | Membri del team |

***Indice***

**1.** Introduzione

**2.** Descrizione delle entità persistenti

**3.** Diagramma dei dati persistenti   
 **3.1** *Schema ER* **3.2** *Modello logico-relazionale* **3.3** *Vincoli di chiave* **3.4** *Vincoli non esprimibili*

1. Introduzione

I dati persistenti vengono gestiti attraverso un database relazionale (eSoccerceDB). Si sceglie di utilizzare un database relazionale perché offre molti vantaggi tra cui:

* Permette di gestire le transazioni con l’eventuale possibilità di fare rollback nel caso in cui la transazione non va a buon fine.
* I dati sono mappati in tabelle che ci permettono di identificare facilmente in modo univoco le istanze, inoltre permette di rappresentare facilmente le relazione trai dati.
* Garantisce, inoltre le proprietà di ACID (Atomicity, Consistency, Isolation, Durability).

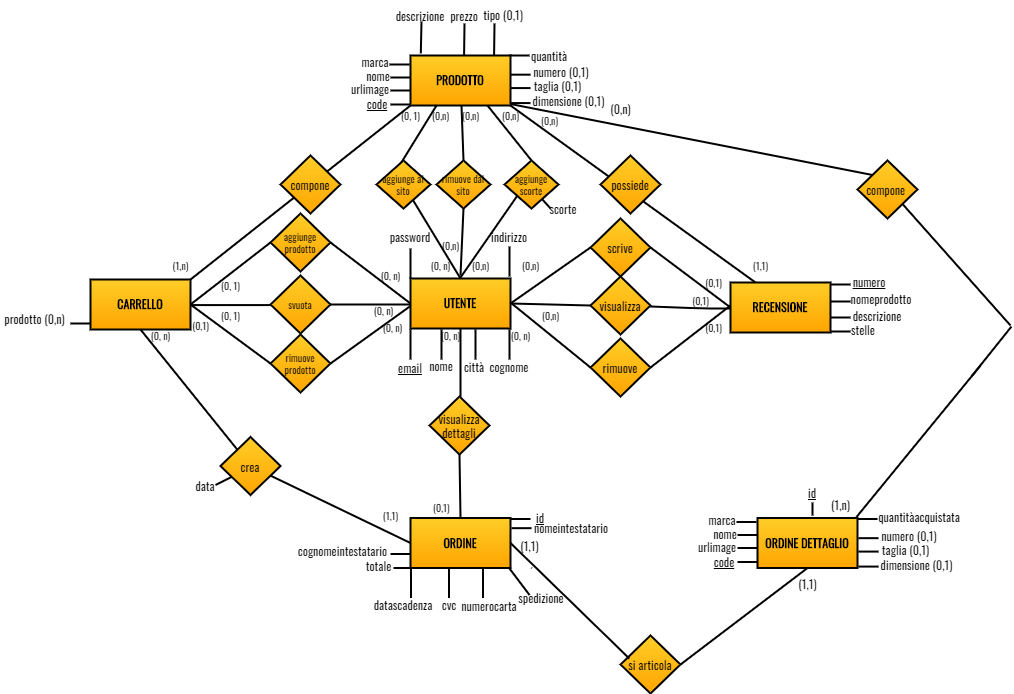
**2.** Descrizione delle entità persistenti

I dati persistenti da memorizzare vengono mappati nelle seguenti entità all’interno del database:

|  |  |
| --- | --- |
| Entità | Descrizione |
| *Utente* | Rappresenta gli utenti che si registrano al sito, effettuando la procedura di registrazione |
| *Amministratore* | Rappresenta l’utente che ha un ruolo speciale, ovvero l’amministratore, colui che si occupano della gestione dei prodotti e della piattaforma |
| *Prodotto* | Rappresenta l’insieme di prodotti presenti in magazzino, nonché quelli disponibili alla vendita o, eventualmente, indisponibili al momento |
| *Ordine* | Rappresenta l’insieme di ordini effettuati dagli utenti, ogni ordine è ovviamente riferito ad un singolo utente ed è composto da 1 o più prodotti |
| *Ordine dettaglio* | Rappresenta i dettagli e le informazioni utili per quanto riguarda le componenti di un ordine, dunque, articoli acquistati, quantità, data spedizione e altri |
| *Recensione* | Rappresenta l’insieme delle recensioni relative ad un singolo prodotto aggiunti e rimossi dall’utente |

**3.** Diagramma dei dati persistenti

**3.1** *Schema ER*

**

**3.2** *Modello logico-relazionale*

***Utente –*** (email, nome, cognome, città, password, indirizzo)  
***Prodotto –*** (code, urlimage, nome, marca, descrizione, prezzo, quantità, taglia\*, numero\*, dimensione\*, tipo\*)  
***Recensione –*** (numero, nomeprodotto, descrizione, stelle, email)  
***Ordine –*** (id, totale, numerocarta, cvc, datascadenza, nomeintestatario, cognomeintestatario, dataordine, spedizione, email)  
***Ordine dettaglio –*** (id, code, marca, nome, quantitàacquistata, urlimage, nome, taglia\*, numero\*, dimensione\*, tipo\*)  
***Aggiunge\_al\_sito –*** (email, code)  
***Rimuove\_dal\_sito –*** (email, code)  
***Aggiunge\_scorte –*** (email, code, scorte)

**3.3** *Vincoli di chiave*

* L’entità **Ordine** è associata all’entità **Utente** mediante un’associazione 1:N. Infatti, l’entità **Ordine** possiede un attributo, email, il quale è chiave primaria dell’entità **Utente**, quindi, in questo caso, siamo in presenza di una chiave esterna. Sulla chiave esterna email, dunque, sono applicate le seguenti politiche di reazione: ON UPDATE CASCADE ON DELETE CASCADE quindi se l’attributo email in **Utente** venisse aggiornato, l’aggiornamento verrebbe applicato anche sulla chiave esterna nell’entità **Ordine**, così come l’eliminazione.
* L’entità **Recensione** è associata all’entità **Utente** mediante un’associazione 1:N. Infatti, l’entità **Recensione** possiede un attributo, email, il quale è chiave primaria dell’entità **Utente**, quindi, in questo caso, siamo in presenza di una chiave esterna. Sulla chiave esterna email, dunque, sono applicate le seguenti politiche di reazione: ON UPDATE CASCADE ON DELETE CASCADE quindi se l’attributo email in **Utente** venisse aggiornato, l’aggiornamento verrebbe applicato anche sulla chiave esterna nell’entità **Recensione**, così come l’eliminazione.
* L’entità **Recensione** è associata all’entità **Prodotto** mediante un’associazione 1:N. Infatti, l’entità **Recensione** possiede un attributo, code, il quale è chiave primaria dell’entità **Prodotto**, quindi, in questo caso, siamo in presenza di una chiave esterna. Sulla chiave esterna code, dunque, sono applicate le seguenti politiche di reazione: ON UPDATE CASCADE ON DELETE CASCADE quindi se l’attributo code in **Prodotto** venisse aggiornato, l’aggiornamento verrebbe applicato anche sulla chiave esterna nell’entità **Recensione**, così come l’eliminazione.
* L’entità **Aggiunge\_al\_sito** è l’entità che nasce dall’associazione N:N tra **Utente** e **Prodotto**. La chiave primaria dell’entità **Aggiunge\_al\_sito** è una chiave composta formata dalle due chiavi esterne email e code che referenziano rispettivamente l’entità **Utente** e l’entità **Prodotto**, alle quali sono applicate le seguenti politiche di reazione: ON UPDATE CASCADE ON DELETE CASCADE quindi se venisse aggiornata la chiave primaria nelle rispettive entità cui hanno gli attributi email e code come chiave primaria, questa modifica verrebbe applicata anche nell’entità **Aggiunge\_al\_sito**, così come l’eliminazione.
* L’entità **Rimuove\_dal\_sito** è l’entità che nasce dall’associazione N:N tra **Utente** e **Prodotto**. La chiave primaria dell’entità **Rimuove\_dal\_sito** è una chiave composta formata dalle due chiavi esterne email e code che referenziano rispettivamente l’entità **Utente** e l’entità **Prodotto**, alle quali sono applicate le seguenti politiche di reazione: ON UPDATE CASCADE ON DELETE CASCADE quindi se venisse aggiornata la chiave primaria nelle rispettive entità cui hanno gli attributi email e code come chiave primaria, questa modifica verrebbe applicata anche nell’entità **Rimuove\_dal\_sito**, così come l’eliminazione.
* L’entità **Aggiunge\_scorte** è l’entità che nasce dall’associazione N:N tra **Utente** e **Prodotto**. La chiave primaria dell’entità **Aggiunge\_scorte** è una chiave composta formata dalle due chiavi esterne email e code che referenziano rispettivamente l’entità **Utente** e l’entità **Prodotto**, alle quali sono applicate le seguenti politiche di reazione: ON UPDATE CASCADE ON DELETE CASCADE quindi se venisse aggiornata la chiave primaria nelle rispettive entità cui hanno gli attributi email e code come chiave primaria, questa modifica verrebbe applicata anche nell’entità **Aggiunge\_scorte**, così come l’eliminazione.

**3.4** *Vincoli non esprimibili*

* L’attributo **numerocarta** dell’entità **Ordine** è un intero che deve essere lungo obbligatoriamente 16 cifre.
* L’attributo **stelle** dell’entità **Recensione** è un intero che può assumere valore 1, 2, 3, 4, 5 e non rappresentano altro che il numero di stelle associato alla recensione per valutare la qualità del prodotto a cui si riferisce la recensione stessa.
* L’attributo **cvc** dell’entità **Ordine** è un intero che deve essere lungo obbligatoriamente 3 cifre.
* L’attributo **datascadenza** dell’entità **Ordine** è una stringa che deve essere obbligatoriamente in un formato xx/xx (x rappresentano interi).
* L’attributo **password** dell’entità **Utente** è una stringa che deve essere lunga almeno 8 caratteri.
* Gli attributi con il carattere \* associato possono assumere valore null.