

*Università degli studi di Salerno*

*Dipartimento di Informatica*

*Corso di Laurea in Informatica*

*Ingegneria del Software*

*System Design Document*

*“Techblin”*

**Docente:**

Andrea De Lucia

**Studenti:**

***Nome***

***Matricola***

Della Corte Giacomo

06078

Di Pierno Andrea

05724

Tuozzo Gabriele

05916

*Anno Accademico: 2020/21*

# **INDICE**

## Sommario

|   |    |
|---|----|
| 1. Introduzione.....                            | 4  |
| 1.1 Obiettivi di progettazione .....            | 4  |
| 1.2 Definizioni, acronimi e abbreviazioni ..... | 5  |
| 1.3 Riferimenti.....                            | 6  |
| 1.4 Architettura software corrente .....        | 6  |
| 1.5 Architettura software proposta .....        | 6  |
| 1.6 Overview.....                               | 6  |
| 2. Decomposizione in sottosistemi .....         | 7  |
| 3. Mapping hardware/software.....               | 9  |
| 4. Gestione dei dati persistenti .....          | 10 |
| 5. Controllo e sicurezza degli accessi .....    | 14 |
| 6. Controllo globale del software.....          | 14 |
| 7. Boundary condition.....                      | 15 |
| 7.1 Failure.....                                | 15 |

# Revisioni

| Data       | Versione | Descrizione  | Autore   |
|------------|----------|--|--|
| 10/12/2020 | 1.0      | Creazione documento ed inserimento paragrafi   | Tuozzo Gabriele<br>Di Pierno Andrea                        |
| 11/12/2020 | 1.1      | Inserimento paragrafi: "Introduzione",<br>"Decomposizione in sottosistemi", "Mapping<br>hardware/software", "Obbiettivi di progettazione",<br>"Mapping hardware/software", "Controllo globale<br>del software", "Condizioni limite". | Della Corte Giacomo  |
| 02/02/2021 | 2.0      | Completamento ultime parti del documento con<br>inserimento "Gestione dei dati persistenti" e<br>"Boundary condition"  | Tuozzo Gabriele  |
| 04/02/2021 | 3.0      | Revisione documento  | Della Corte Giacomo<br>Tuozzo Gabriele<br>Di Pierno Andrea |

# 1. Introduzione

## 1.1 Obiettivi di progettazione

### Usability

- **Interfaccia facilmente comprensibile:** Il sito deve avere un'interfaccia intuitiva, familiare all'utente che conosce altri siti di e-commerce, prevedendo una barra di navigazione che guidi l'utente all'interno del sito.
- **Responsività:** Il sito deve essere responsive, in modo da garantire che il layout del sito si adatti a qualsiasi schermo. A tale scopo dovrà essere usato il framework Bootstrap.

### Performance

- **Tempi di risposta brevi:** la web application deve avere tempi di risposta brevi, ogni pagina deve caricarsi entro un tempo di 4 secondi. Questo tempo può essere monitorato attraverso strumenti come MySQL Workbench per verificare il tempo impiegato per effettuare una query nel database.

### Supportability

- **Supporto ai cambiamenti:** La web application dovrà essere sviluppata seguendo il pattern MVC in modo che i cambiamenti possano essere gestiti facilmente.

### Legal

- **Informazioni legali e sulla privacy:** deve essere implementata una pagina contenente le informazioni sull'utilizzo dei dati personali e le condizioni di utilizzo della web application.

### Security

- **Comunicazione sicura:** la comunicazione deve avvenire attraverso appositi protocolli di sicurezza web(SSL), utilizzando un certificato rilasciato da enti autorizzati.
- **Anti-SQL injection:** la web application deve essere protetta da possibili attacchi SQL injection, implementando le query parametriche.
- **Anti-Hijacking:** la web application deve impedire ad un malintenzionato di inserire oggetti invisibili con hyperlink verso altri siti web, utilizzando gli strumenti messi a disposizione dal server web Apache Tomcat 9.0.

## 1.2 Definizioni, acronimi e abbreviazioni

| Definizione, acronimo o abbreviazione | Significato  |
|---------------------------------------|--|
| MVC                                   | Model View Controller, pattern architetturale per lo sviluppo del sistema.   |
| GDPR                                  | Il Regolamento Generale sulla Protezione dei Dati.   |
| SQL Injection                         | Tecnica di code injection, usata per attaccare applicazione che gestiscono dati attraverso database relazionali.   |
| Hijacking                             | Il termine hijacking indica una tecnica di attacco informatico che consiste nel modificare opportunamente dei pacchetti dei protocolli TCP/IP al fine di dirottare i collegamenti ai propri siti web e prenderne il controllo. |
| SSL                                   | Secure Socket Layer. Protocollo che permette una connessione sicura.   |
| HTML                                  | Linguaggio di mark-up per la creazione di pagine web.  |
| DBMS                                  | Un Database Management System è un sistema software progettato per consentire la creazione, la manipolazione e l'interrogazione efficiente di database   |
| JSP                                   | Java Server Page.  |
| JS                                    | JavaScript. Linguaggio di programmazione per script nelle pagine web lato client.  |
| HTTPS                                 | Protocollo di comunicazione sicura.  |
| CSS                                   | Cascading Style Sheet. Linguaggio utilizzato per definire la formattazione di pagine HTML.   |
| BOOTSTRAP                             | Bootstrap è un framework utilizzato per progettare pagine HTML, andando a modellare l'interfaccia utente.  |
| APACHE TOMCAT                         | Apache Tomcat è un server web open source.   |
| JDBC                                  | Connettore per database che consente l'accesso ai dati persistenti utilizzato in Java.   |

### **1.3 Riferimenti**

- Requirement Analysis Document (RAD)

### **1.4 Architettura software corrente**

Il progetto Techblin è un greenfield engineering project. Quindi non vi è alcun sistema da rimpiazzare ma è costruito da zero.

### **1.5 Architettura software proposta**

#### **1.6 Overview**

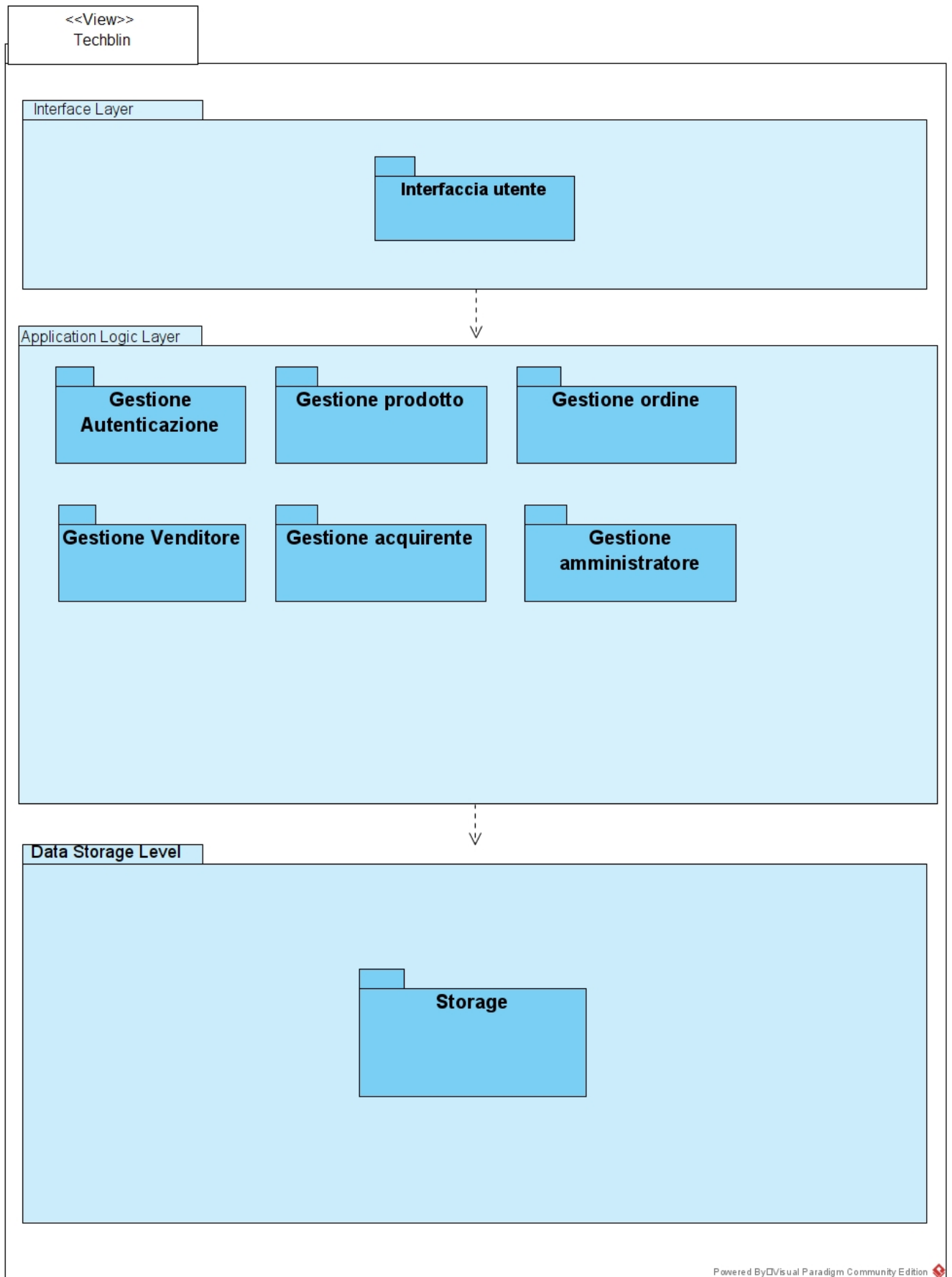
Per l'implementazione del sito è stato adottato lo stile architetturale Model – View – Controller.

I componenti Model forniscono l'accesso ai dati persistenti attraverso i metodi, sono l'unica componente del sistema che accede ai dati persistenti.

I componenti View si occupano di rappresentare le informazioni e i dati richiesti dall'utente. Quindi le view è la parte che si interfaccia con l'utente.

I componenti Controller hanno il compito di comunicare con le View sui cambiamenti del Model e gestiscono le iterazioni con l'utente che utilizza l'applicazione.

## 2. Decomposizione in sottosistemi

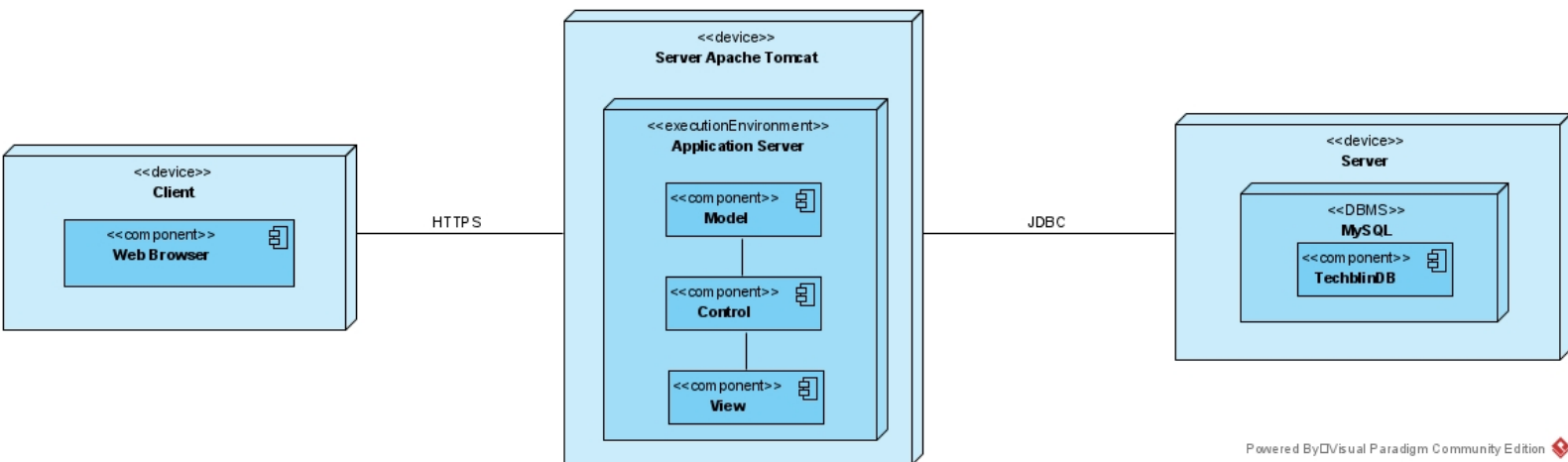


- **Gestione autenticazione**
  - Registrazione venditore
  - Registrazione acquirente
  - Login
  - Logout
  - Modifica e-mail
  - Modifica password
- **Gestione prodotto**
  - Aggiungi prodotto
  - Modifica prodotto
  - Elimina prodotto
  - Ricerca prodotto
- **Gestione ordine**
  - Checkout
  - Visualizzazione ordini
- **Gestione acquirente**
  - Aggiungi prodotto al carrello
  - Rimuovi prodotto dal carrello
  - Svuota carrello
- **Gestione venditore**
  - Visualizzazione vetrina prodotti in vendita
- **Gestione amministratore**
  - Visualizzazione statistiche
  - Creazione nuovo amministratore
  - Visualizzazione catalogo



### 3. Mapping hardware/software

Il sistema è sviluppato su un server che utilizza un DBMS MySQL per la gestione dei dati persistenti. Il client può essere un qualsiasi dispositivo con connessione internet e un browser. Il server utilizzato è Apache Tomcat 9.0 che fornirà la funzione di Web Server. Le pagine JSP/HTML realizzano il front-end del sistema e cioè, realizzano le interfacce del sistema, mentre le Servlet Java realizzano la logica applicativa. Il protocollo utilizzato per scambiare dati fra client e server sarà HTTPS, fornendo dunque sicurezza nello scambio dei dati. Il server comunica con lo strato persistente utilizzando le API JDBC.

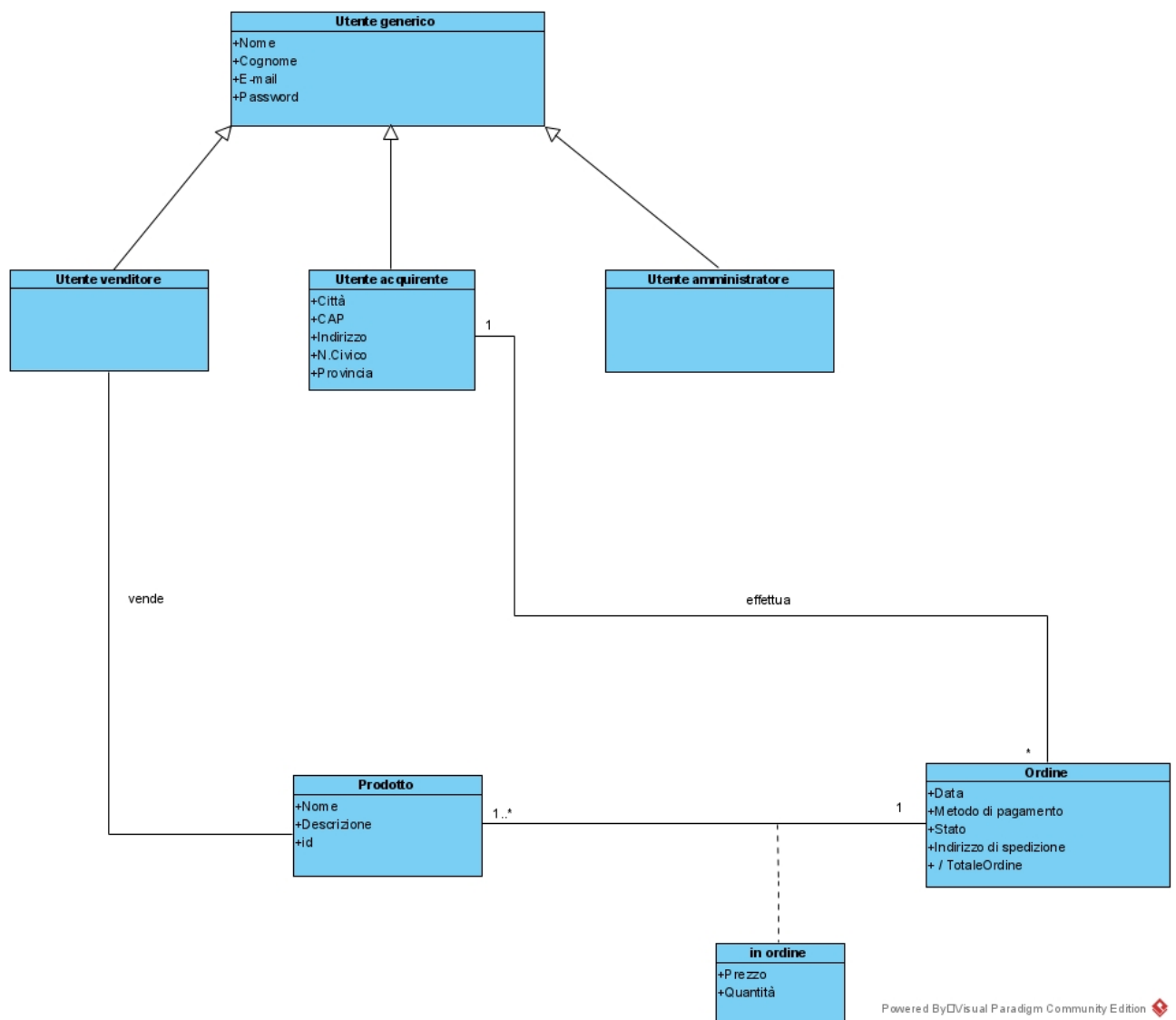


## 4. Gestione dei dati persistenti

I dati persistenti saranno memorizzati attraverso il DBMS MySQL seguendo il diagramma delle entità.

La scelta dell'utilizzo di un database, a discapito del file system, risiede nel fatto che per un e-commerce risulta più appropriato usare un database, le motivazioni essenzialmente sono le seguenti:

- Possibilità di mettere in relazione le tabelle e di conseguenza i dati.
- Il poter recuperare i dati in modo più efficiente attraverso le query.
- Supporto alle transazioni e quindi operazioni ACID (cruciale per un'applicazione di e-commerce).
- Operazioni di backup automatiche e di recupero di dati.
- Maggiore sicurezza.
- Inoltre, la possibilità di poter accedere in modo concorrente ai dati.



Utilizziamo il class diagram individuato in fase di analisi come schema per andare a creare il nostro database. Tuttavia, abbiamo effettuato alcune modifiche, in quanto il carrello non sarà mantenuto come entità persistente ma sarà gestito univocamente all'interno della sessione http. Un'altra modifica è stata quella relativa al catalogo, in quanto nel nostro database non prevederemo alcuna tabella che andrà a mantenere le informazioni relative ai prodotti nel catalogo. Il resto del class diagram è invariato e sarà mappato sul database nel seguente modo.

## VENDITORE

| Nome          | Tipo | Null     | Chiave          |
|---------------|------|----------|-----------------|
| ID            | CHAR | Not Null |                 |
| Nome          | CAHR | Not Null |                 |
| Cognome       | CHAR | Not null |                 |
| E-mail        | CHAR | Not null | Chiave primaria |
| Password      | CHAR | Not null |                 |
| Registrazione | DATE | Not null |                 |

## ACQUIRENTE

| Nome          | Tipo | Null     | Chiave          |
|---------------|------|----------|-----------------|
| Nome          | CHAR | Not Null |                 |
| Cognome       | CAHR | Not null |                 |
| E-mail        | CHAR | Not null | Chiave primaria |
| Password      | CHAR | Not null |                 |
| Registrazione | Date | Not null |                 |

**AMMINISTRATORE**

| Nome          | Tipo | Null     | Chiave          |
|---------------|------|----------|-----------------|
| Nome          | CHAR | Not Null |                 |
| Cognome       | CHAR | Not null |                 |
| E-mail        | CHAR | Not null | Chiave primaria |
| Password      | CHAR | Not null |                 |
| Registrazione | DATE | Not null |                 |

**PRODOTTO**

| Nome        | Tipo | Null     | Chiave          |
|-------------|------|----------|-----------------|
| Code        | INT  | Not Null | Chiave primaria |
| Nome        | CHAR | Not null |                 |
| Descrizione | CHAR | Not null |                 |
| Prezzo      | INT  | Not null |                 |
| Quantità    | INT  | Not null |                 |
| Visibile    | INT  | Not null |                 |

**ORDINE**

| Nome                | Tipo | Null     | Chiave          |
|---------------------|------|----------|-----------------|
| Id                  | INT  | Not Null | Chiave primaria |
| Totale              | INT  | Not null |                 |
| Data Ordine         | DATE | Not null |                 |
| Stato               | CHAR | Not null |                 |
| Acquirente          | CHAR | Not null | Chiave esterna  |
| Indirizzo           | CHAR | Not null |                 |
| CAP                 | CHAR | Not null |                 |
| Città               | CHAR | Not null |                 |
| Provincia           | CHAR | Not null |                 |
| Tipologia pagamento | CHAR | Not null |                 |
| Dati pagamento      | CHAR | Not null |                 |

**PRODOTTI ORDINE**

| Nome     | Tipo | Null     | Chiave          |
|----------|------|----------|-----------------|
| ID       | INT  | Not Null | Chiave primaria |
| Prodotto | INT  | Not null | Chiave esterna  |
| Ordine   | INT  | Not null | Chiave esterna  |

## 5. Controllo e sicurezza degli accessi

La Sicurezza viene in parte garantita dall'utilizzo di un protocollo HTTPS, non verranno accettate in alcun modo connessioni con protocolli non sicuri come http.

**Matrice degli accessi**

| Oggetti                      | Utente                              | Prodotto  | Carrello  | Ordine                             |
|------------------------------|-------------------------------------|---|---|------------------------------------|
| <b>Attori</b>                |                                     |   |   |                                    |
| <b>Utente non registrato</b> | Registrazione()                     | RicercaProdotto()   |   |                                    |
| <b>Utente venditore</b>      | Login()<br>Logout()                 | MostraProdotti()<br>AggiungiProdotto()<br>ModificaProdotto()<br>CancellaProdotto()<br>RicercaProdotto() |   |                                    |
| <b>Utente amministratore</b> | Login()<br>Logout()<br>NuovoAdmin() | MostraProdotti()<br>ModificaProdotto()<br>CancellaProdotto()<br>RicercaProdotto()                       |   |                                    |
| <b>Utente acquirente</b>     | Login()<br>Logout()                 | RicercaProdotto()   | AggiungiProdotto()<br>SvuotaCarrello()<br>EliminaProdotto() | AggiungiOrdine()<br>MostraOrdini() |

## 6. Controllo globale del software

Il controllo globale del software è di tipo event-based. Abbiamo il Web Server che si occupa di intercettare le richieste degli utenti (i client). Il server poi processa le richieste dei client attraverso i Control, dunque le Servlet, che gestiscono la richiesta del client ed interagiscono con i Manager (Model) nel caso in cui è richiesto l'accesso ai dati persistenti. Elaborata la richiesta, ed eventualmente dopo aver interrogato il Model ed ottenuto un risultato, i Control hanno il compito di aggiornare le View, e cioè le JSP, attraverso la generazione di pagine HTML che formattano la risposta alla richiesta del client.

## 7. Boundary condition

### 7.1 Failure

|  |                 |  |
|--|-----------------|--|
| <b>Identificativo</b><br>[UC001]                 |                 | <b>Server non raggiungibile</b>  |
| <b>Descrizione</b>                               |                 | <i>Failure dovuta a server non raggiungibile</i>   |
| <b>Attore Principale</b>                         |                 | <b>Utente generico</b>   |
| <b>Attori secondari</b>                          |                 |  |
| <b>Entry condition</b>                           |                 | L'utente cerca di raggiungere il sito.   |
| <b>Extension point</b>                           |                 |  |
| <b>Generalization of</b>                         |                 |  |
| <b>FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO</b> |                 |  |
| <b>1</b>   | <b>Attore:</b>  | L'utente cerca di raggiungere il sito digitando l'indirizzo del sito nella barra di ricerca. |
| <b>2</b>   | <b>Sistema:</b> | Viene restituita una pagina di errore con codice di errore 5xx.                              |
|  |                 |  |
| <b>Exit condition</b><br>On success              |                 | Il sistema non è raggiungibile.  |
|  |                 |  |

|  |                 |  |
|--|-----------------|--|
| <b>Identificativo</b><br>[UC002]                 |                 | <b>Errore durante una transazione</b>  |
| <b>Descrizione</b>                               |                 | <i>Failure dovuta ad un errore durante una transazione.</i>  |
| <b>Attore Principale</b>                         |                 | <b>Utente generico</b>   |
| <b>Attori secondari</b>                          |                 |  |
| <b>Entry condition</b>                           |                 | L'utente sta concludendo un ordine.  |
| <b>Extension point</b>                           |                 |  |
| <b>Generalization of</b>                         |                 |  |
| <b>FLUSSO DI EVENTI PRINCIPALE/MAIN SCENARIO</b> |                 |  |
| <b>1</b>   | <b>Attore:</b>  | L'utente compila i dati riguardanti la spedizione ed invia il form. Durante l'invio si verifica un'assenza di connessione. |
| <b>2</b>   | <b>Sistema:</b> | Effettua il rollback della transazione e l'inserimento dell'ordine fallisce.   |
|  |                 |  |
| <b>Exit condition</b><br>On success              |                 | L'ordine non viene inserito.   |
|  |                 |  |