

Università degli Studi di Salerno

Corso di Ingegneria del Software

MusicParadise.com

RAD

Versione 2.0

LOGO PROGETTO



Partecipanti:

Nome	Matricola
Antonio Spera	0512103538
Alessandro De Riso	0512103802
Domenico Pannone	0512103730
Vincenzo Pandolfo	0512103694

Revision History

Data	Versione	Descrizione	Autore
18/10/2017		Inizio dei Requisiti funzionali e descrizione del sistema.	Alessandro De Riso
20/10/2017		Revisione requisiti funzionali.	Alessandro De Riso
21/10/2017		Inizio casi d'uso	Antonio Spera
22/10/2017		Scenari	Domenico Pannone
23/10/2017		Revisione casi d'uso	Antonio Spera
23/10/2017		Requisiti non funzionali	Vincenzo Pandolfo
26/10/2017		Revisione requisiti non funzionali	Antonio Spera, Alessandro De Riso
11/11/2017		Aggiunti diagrammi dei casi d'uso	Antonio Spera
11/11/2017		Aggiunti mockup (aggiungi prodotto al carrello, acquista prodotto)	Antonio Spera
11/11/2017		Aggiunti MockUp (aggiungi prodotto, aggiungi offerta)	Alessandro De Riso
17/11/2017		Aggiunti sequence diagrams (Login, Registrazione, Aggiungi prodotto al carrello, Acquista Prodotto, modifica stato ordine)	Antonio Spera.
17/11/2017		Aggiunti class diagrams (Login, Registrazione, aggiungi prodotto al carrello, acquista prodotto, modifica stato ordine)	Antonio Spera
17/11/2017		Aggiunti Sequence Diagram (aggiungi nuovo prodotto, aggiungi offerta a un prodotto)	Alessandro De Riso
20/11/2017		Aggiunto sequence diagrams (modifica prodotto)	Vincenzo Pandolfo
23/11/2017		Aggiunti class diagrams(acquista prodotto, aggiungi offerta prodotto e aggiungi prodotto al catalogo)	Alessandro De Riso
06/12/2017		Revisione e aggiunta path navigazionali	Antonio Spera
10/01/2018		Revisione del documento	Antonio Spera
		Ingegneria del Software	Pagina 2 di 71

Sommario

3. Sistema proposto	5
3.1. Requisiti funzionali	5
3.2. REQUISITI NON FUNZIONALI	7
3.2.1. USABILITA'	7
3.2.2. Affidabilità	7
3.2.3. PERFORMANCE	7
3.2.4. SUPPORTABILITA'	7
3.3. Pseudo-requisiti	7
3.3.1. Interfaccia	7
3.3.2. Implementazione	8
3.3.3. Documentazione	8
3.4. System model	9
3.4.1. Scenari	9
3.4.2. MODELLO CASI D'USO	11
.....	11
.....	12
3.4.3. Object model	28
3.4.3.1 Class Diagrams	28
3.4.3.2. Class diagrams Login	29
3.4.3.3. Class Diagrams Registrazione	29
3.4.3.4. Class Diagrams Aggiungi prodotto al carrello	30
3.4.3.5. Class Diagrams Acquista prodotto	30
3.4.3.6. Class Diagrams Aggiungi prodotto al catalogo	31
3.4.3.7. Aggiungi offerta prodotto	32
3.4.3.8. Class Diagrams Modifica prodotto	34
3.4.3.9. Class Diagrams Modifica dati cliente	34
3.4.3.10. Class Diagrams Rimuovi prodotto dal carrello	35
3.4.3.11. Class Diagrams Modifica stato ordine	36
3.4.4.12. Class Diagram Aggiungi indirizzo	36
3.4.4.12. Class Diagrams Aggiungi carta	37
3.4.4.13. Class Diagram Ricerca Prodotto	37
3.4.4.14. Visualizza Profilo	38
3.4.5. Dynamic model	39
3.4.5.12. Login	39
3.4.5.13. Registrazione	40
3.4.4.3. Aggiungi prodotto al carrello	41
3.4.4.4. Acquista prodotto	42
3.4.4.5. Gestore Prodotti – Aggiungi Prodotto	43
3.4.4.6. Gestore Offerte– Aggiungi Offerta Prodotto	44
3.4.4.7. Modifica prodotto	45
3.4.4.8. Modifica dati utente	46
3.4.4.9. Rimuovi prodotto dal carrello	47
3.4.4.10. Modifica stato ordine	48
3.4.4.11. Aggiungi indirizzo	49
3.4.4.12. Aggiungi carta	50
3.4.4.13. Visualizza catalogo	51

3.4.4.14.	Diagramma di stato carrello.....	52
3.4.5.	MOCKUP	54
3.4.5.1.	Aggiungi prodotto al carrello.....	54
3.4.5.2.	ACQUISTA PRODOTTO.....	57
3.4.5.3.	Gestore Prodotti	61
3.4.5.4.	Gestore Offerte.....	64
3.4.6.	Path navigazionali	67
3.4.6.1.	Visitatore	67
3.4.6.2.	Cliente	67
3.4.6.3.	Gestore prodotti.....	68
3.4.6.4.	Gestore ordini.....	68
3.4.6.5.	Gestore utenti	69
3.4.6.6.	Gestore offerte.....	69

3. Sistema proposto

3.1. Requisiti funzionali

Elenchiamo i requisiti del cliente per tipologia di utente del sistema:

Sessione utente (visitatore)

Il sistema deve prevedere una sessione pubblica accessibile da ogni utente. Fornisce informazioni relative al negozio di e-commerce ed alle modalità di registrazione:

- Informazioni generiche sui prodotti;
- Area per gli accessi e la registrazione;
- Link ai social iscritti;
- Contatti;

Le aree private del sistema sono accessibili tramite autenticazione.

Sessione Utente (Cliente/Gestore/Amministratore)

- Login RF1
- Logout RF3

Sessione Utente (Cliente)

- Registrazione RF4
- Visualizzare profilo; RF5
- Aggiunta di un nuovo indirizzo; RF6
- Aggiunta di un nuovo metodo di pagamento; RF7
- Rimuovi indirizzo RF8
- Rimuovi metodo di pagamento RF9
- Cercare prodotti; RF10
- Visualizza storico ordini; RF11
- Aggiungi prodotto al carrello RF12
- Rimuovi prodotto dal carrello; RF13
- Visualizza catalogo RF14
- Visualizza carrello RF15
- Checkout RF16

Sessione Gestore prodotti

- Inserimento di nuovi prodotti; RF17
- Modifica informazioni relative ai prodotti già in vendita; RF18
- Cancellazione dei prodotti; RF19

Sessione Gestore offerte

- Inserimento di nuove offerte; RF20
- Aggiunta offerta al prodotto RF21
- Cancellazione offerte; RF22

Sessione Gestore utenti

- Visualizzare tutti gli utenti registrati al sito; RF23
- Bannare un utente; RF24

Sessione Gestore ordini

- Visualizza ordini in preparazione, da spedire e consegnati; R25
- Modifica stato dell'ordine da in preparazione a spedito e da spedito a consegnato.; RF26

Applicazione Admin

- Creare utenti con privilegi; RF27
- Eliminare utenti con privilegi; RF28

3.2.REQUISITI NON FUNZIONALI

3.2.1. USABILITA'

- a. Il sito è essere responsive, in modo da adattarsi graficamente al dispositivo con il quale viene visualizzato (Pc, Tablet, smartphone, web tv).

3.2.2. Affidabilità

- b. Il sistema deve essere disponibile 24h al giorno 7/7 giorni, salvo aggiornamenti o malfunzionamenti del server.
- c. Gli aggiornamenti del server, preferibilmente, dovranno essere effettuati nelle ore notturne.
- d. I dati sensibili devono essere memorizzati all'interno di un database, accessibile tramite delle credenziali dagli admin.

3.2.3. PERFORMANCE

- a. Non esistono vincoli di prestazione visto che il sistema è di natura web.
- b. I tempi di risposta sono legati alla latenza della rete internet e dallo stato del server.
- c. Il sistema può soddisfare molteplici richieste da parte dei clienti contemporaneamente. Possono verificarsi ritardi nell'utilizzo quando si presenta un gran numero (circa più di 50) di clienti connessi simultaneamente.

3.2.4. SUPPORTABILITA'

- a. Il sistema viene continuamente modificato dopo il suo rilascio iniziale. Lo scopo principale delle modifiche non è unicamente quello di risolvere gli errori commessi, ma soprattutto quello di "evolvere" il sistema.

3.3.Pseudo-requisiti

3.3.1. Interfaccia

- a. Ogni scheda prodotto contenuta nel catalogo *deve* contenere almeno una fotografia a colori del prodotto, il suo nome, il nome del produttore, il prezzo e una descrizione sintetica ma completa, 6 righe di testo al massimo
- b. Le interfacce utilizzano una struttura grafica in cui sono presenti:
 - i. **Bottoni**
 - ii. **Finestre di dialogo**
 - iii. **Finestre scorrevoli**
 - iv. **Aree di immissioni di dati**

3.3.2. Implementazione

- a. Il software sarà implementato con diversi linguaggi di programmazione: JavaScript, HTML, CSS, JQuery, Ajax. Il database sarà implementato nel linguaggio SQL.

3.3.3. Documentazione

- a. La documentazione che viene rilasciata al cliente è costituita di una descrizione completa del software, mostrando tutte le sue potenzialità, in particolare verranno rilasciate ed evidenziate diagrammi e descrizioni testuali contenenti le caratteristiche delle interfacce.

3.4.System model

3.4.1.Scenari

Nome scenario: acquista prodotto.

Attori partecipanti: antonio: cliente

Flusso eventi:

1. Antonio apre www.musicParadise.com
2. Antonio accede al proprio account
3. Antonio vuole cercare se è disponibile la tastiera YAMAHA PSR EW400
4. Antonio tramite la barra delle ricerche digita il nome della tastiera
5. Il sito elenca tutti i risultati trovati inerenti a YAMAHA PSR EW400
6. Antonio in corrispondenza della tastiera YAMAHA PSR EW400 clicca visualizza prodotto.
7. Il sistema a video mostrerà tutte le informazioni relative alla tastiera YAMAHA PSR EW400
8. Antonio nella pagina del prodotto clicca sul tasto aggiungi al carrello.
9. Il sistema notifica che al carrello è stato aggiunto un nuovo prodotto.
10. Antonio clicca sull'icona del carrello e visualizza correttamente l'aggiunta del prodotto.
11. Antonio visualizza il totale della spesa e clicca sul bottone paga ora.
12. Il sistema chiede ad Antonio gli estremi di fatturazione per l'acquisto.
13. Antonio compila il modulo di pagamento, inserendo i propri dati inclusi quella della carta di pagamento.
14. Il sistema notifica che l'ordine N° 837857 è stato effettuato.
15. Antonio accede tramite il menù nella pagina i miei ordini, per visualizzare lo stato della spedizione dell'ordine N° 837857.

Nome Scenario: Aggiunta offerte Yamaha M200

Attori Partecipanti: Domenico: Gestore offerte

Flusso Eventi:

1. L'azienda vuole scontare il prodotto Yamaha M200
2. Domenico esegue l'accesso sul sito www.musicparadise.com
3. seleziona il prodotto Yamaha M200
4. Aggiunge la data di inizio/fine
5. Aggiunge la percentuale di sconto
6. Conferma
7. Visualizza il responso di avvenuta modifica
8. Scenario concluso

Nome Scenario: Modifica stato da esaurito a disponibile

Attori Partecipanti: Antonio: Gestore prodotti

Flusso Eventi:

1. Il prodotto tastiera CASIO non è più disponibile.
2. Antonio dopo l'arrivo di nuove tastiere CASIO rende il prodotto nuovamente disponibile.
3. Antonio effettua il login sul sito www.musicparadise.com
4. Seleziona il prodotto tastiera CASIO (dalla lista prodotti) da rendere nuovamente disponibile
5. inserisce la quantità del prodotto ora disponibile
6. Conferma e visualizza responso di avvenuta modifica
7. Scenario concluso

Nome Scenario: Eliminazione di un prodotto dal catalogo

Attori Partecipanti: Alessandro: Gestore prodotti

Flusso Eventi:

1. Alessandro vuole eliminare dal sito il prodotto: chitarra Fender, in quanto non più prodotto.
2. Alessandro effettua il login al sito www.musicparadise.com.
3. Seleziona dalla lista dei prodotti chitarra Fender.
4. Cancella il prodotto
5. Conferma
6. Scenario concluso

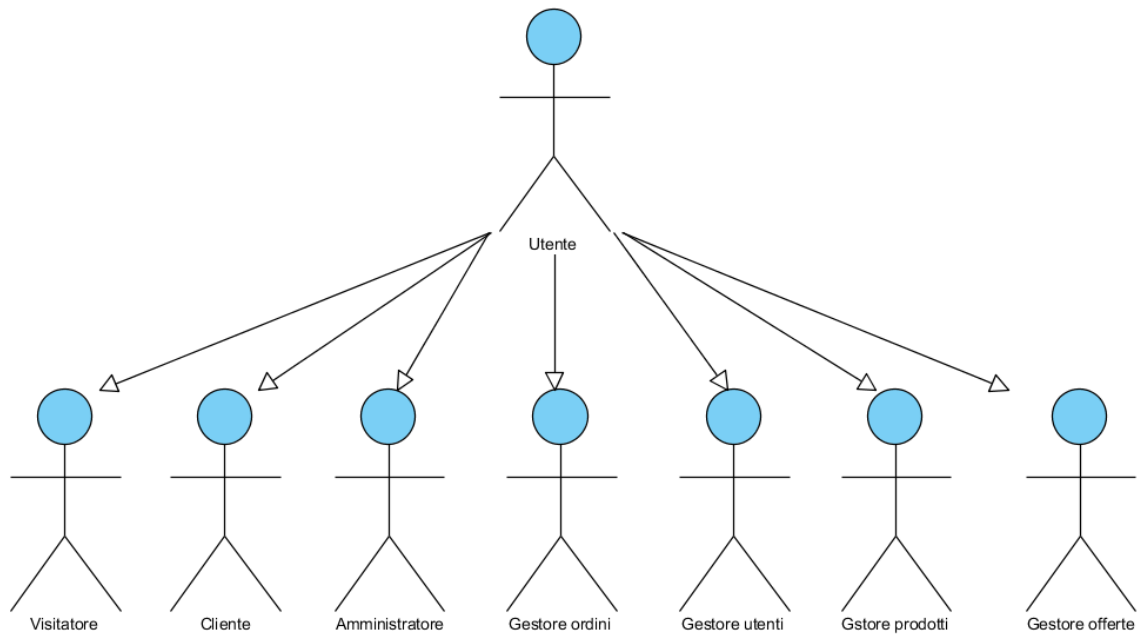
Nome Scenario: Cambio di stato di un ordine

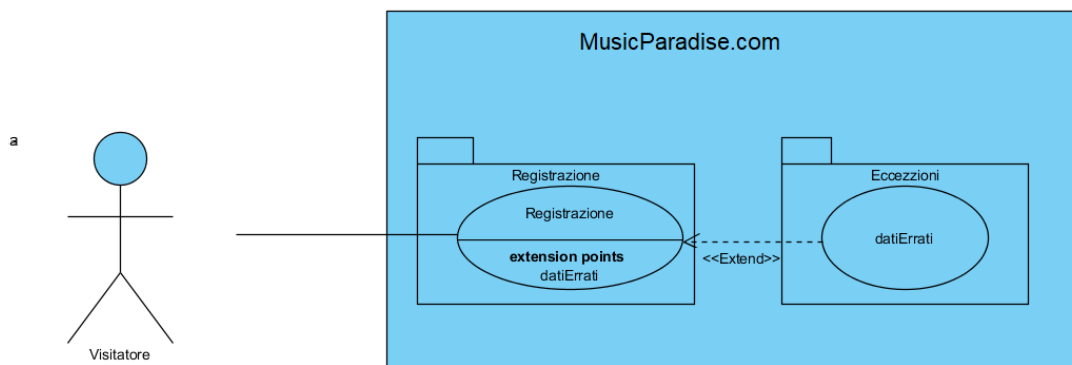
Attori Partecipanti: Alessia: gestore ordini

Flusso Eventi:

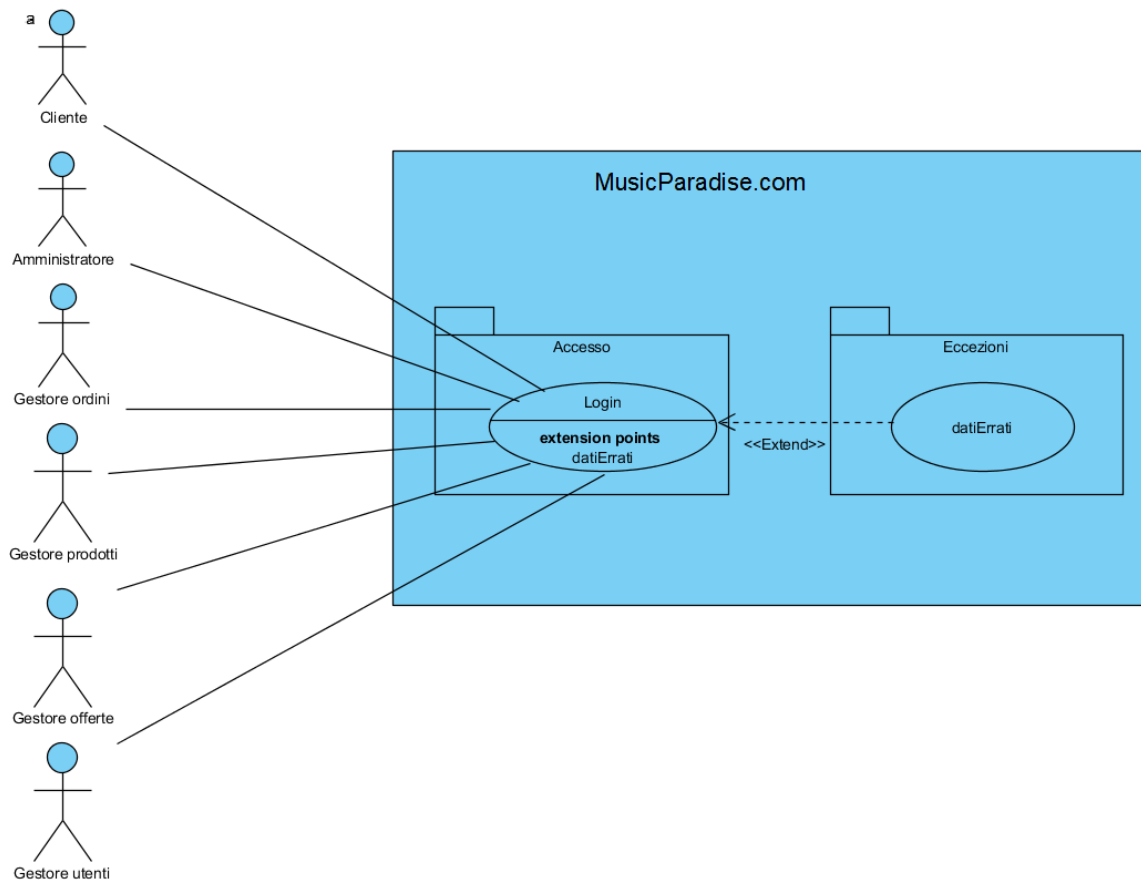
1. Alessia deve gestire gli ordini nel giorno corrente
2. Accede sul sito musicparadise.com
3. Alessia deve cambiare lo stato dell'ordine N°100, effettuato il giorno precedente
4. Pertanto cambia lo stato da "in preparazione" a "spedito"

3.4.2. MODELLO CASI D'USO

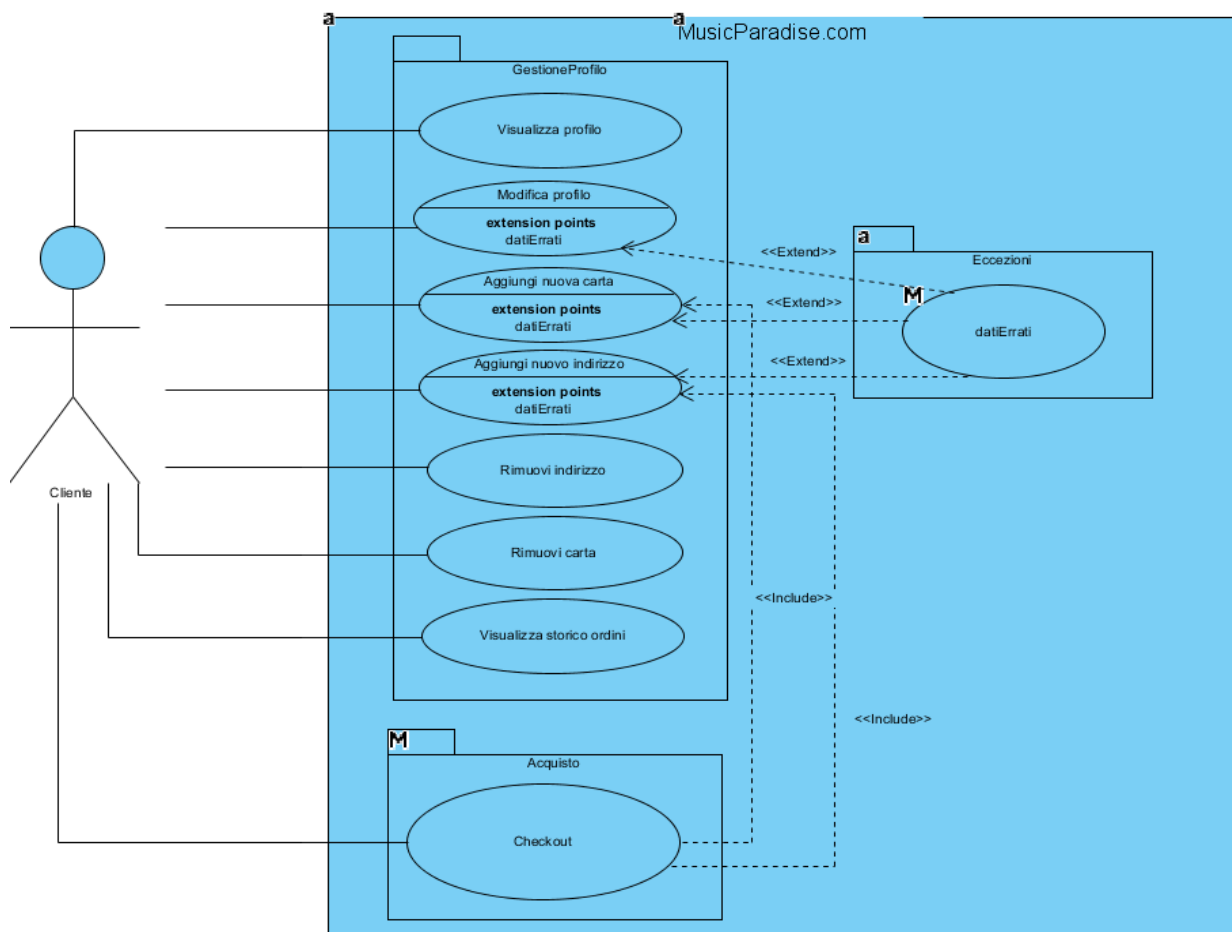




Nome:	Registrazione
ID:	UC01
Partecipante:	Visitatore
Flusso degli eventi:	<p>A. Il sistema mostra una form che richiede l’inserimento di:</p> <ul style="list-style-type: none"> • Nome (obbligatorio) • Cognome (obbligatorio) • Nickname (obbligatorio) • Email (obbligatorio) • Password (obbligatorio) <p>B. Il visitatore compila il modulo per la registrazione e lo sottomette al sistema.</p> <p>C. Il sistema verifica che:</p> <ul style="list-style-type: none"> • Tutti i campi obbligatori sono stati compilati. • Il nickname non sia già in uso. • La password sia lunga almeno 8 caratteri. <p>D. Il sistema se corretti, salva i dati nel database.</p> <p>E. Il sistema mostra una schermata che informa l’utente della registrazione avvenuta con successo.</p>
Condizione di entrata:	il visitatore si trova nell’home del sistema e clicca sul bottone registrati.
Condizione di uscita:	Il sistema notifica l’avvenuta registrazione. L’utente è registrato nel sistema.
Estensione:	<p>il visitatore annulla il login (Caso d’uso annulla)</p> <p>Variante A.1</p> <ul style="list-style-type: none"> • Il visitatore clicca sul pulsante annulla. <p>I dati inseriti non sono corretti. (Caso d’uso datiErrati)</p>



Nome:	Login
ID:	UC02
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il sistema visualizza una form che richiede l'inserimento di:</p> <ul style="list-style-type: none"> Nickname/email Password <p>B. Il cliente inserisce le sue credenziali e sottomette la form appena compilata al sistema.</p> <p>C. Il sistema verifica che:</p> <ul style="list-style-type: none"> Tutti i campi obbligatori sono stati compilati Il sistema riscontra le credenziali inserite con quelle salvate nel database. <p>D. Il sistema mostra una schermata che informa l'utente dell'avvenuto login al sistema se corretti.</p>
Condizione di entrata:	il cliente clicca sul bottone login
Condizione di uscita:	il cliente è loggato al sistema.
Estensione:	<p>il cliente annulla il login (Caso d'uso annulla)</p> <p>Variante B.1</p> <ul style="list-style-type: none"> Il cliente clicca sul pulsante annulla. <p>I dati inseriti non sono corretti. (Caso d'uso datiErrati)</p>



Nome:	Visualizza profilo
ID:	UC03
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il cliente accede alla sezione il mio profilo</p> <p>B. Il sistema visualizza le informazioni del cliente</p>
Condizione di entrata:	Il cliente clicca sul bottone login ed effettua il login
Condizione di uscita:	I dati del profilo sono visibili al cliente.
Estensione:	

Nome:	Visualizza storico ordini
ID:	UC04
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il cliente accede alla sezione i miei ordini</p> <p>B. Il sistema visualizza una tabella contenente gli ordini del cliente</p>
Condizione di entrata:	Il cliente clicca sul bottone login ed effettua il login
Condizione di uscita:	Gli ordini sono visibili al cliente.
Estensione:	

Nome:	Modifica informazioni profilo
ID:	UC05
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il cliente accede alla sezione modifica profilo</p> <p>B. Il sistema visualizza una form che contiene le seguenti informazioni:</p> <ul style="list-style-type: none"> • Nome (obbligatorio) • Cognome (obbligatorio) • Nickname (obbligatorio) • Email (obbligatorio) • Password (obbligatorio) <p>C. Il cliente modifica le informazioni relative al suo profilo e sottomette la form appena compilata al sistema.</p> <p>D. Il sistema verifica che:</p> <ul style="list-style-type: none"> • Tutti i campi obbligatori sono stati compilati <p>E. Il sistema mostra una schermata che informa l'utente dell'avvenuto modifica dei dati.</p>
Condizione di entrata:	Il cliente clicca sul bottone login
Condizione di uscita:	I dati del cliente sono stati modificati.
Estensione:	I dati inseriti non sono corretti. (Caso d'uso datiErrati)

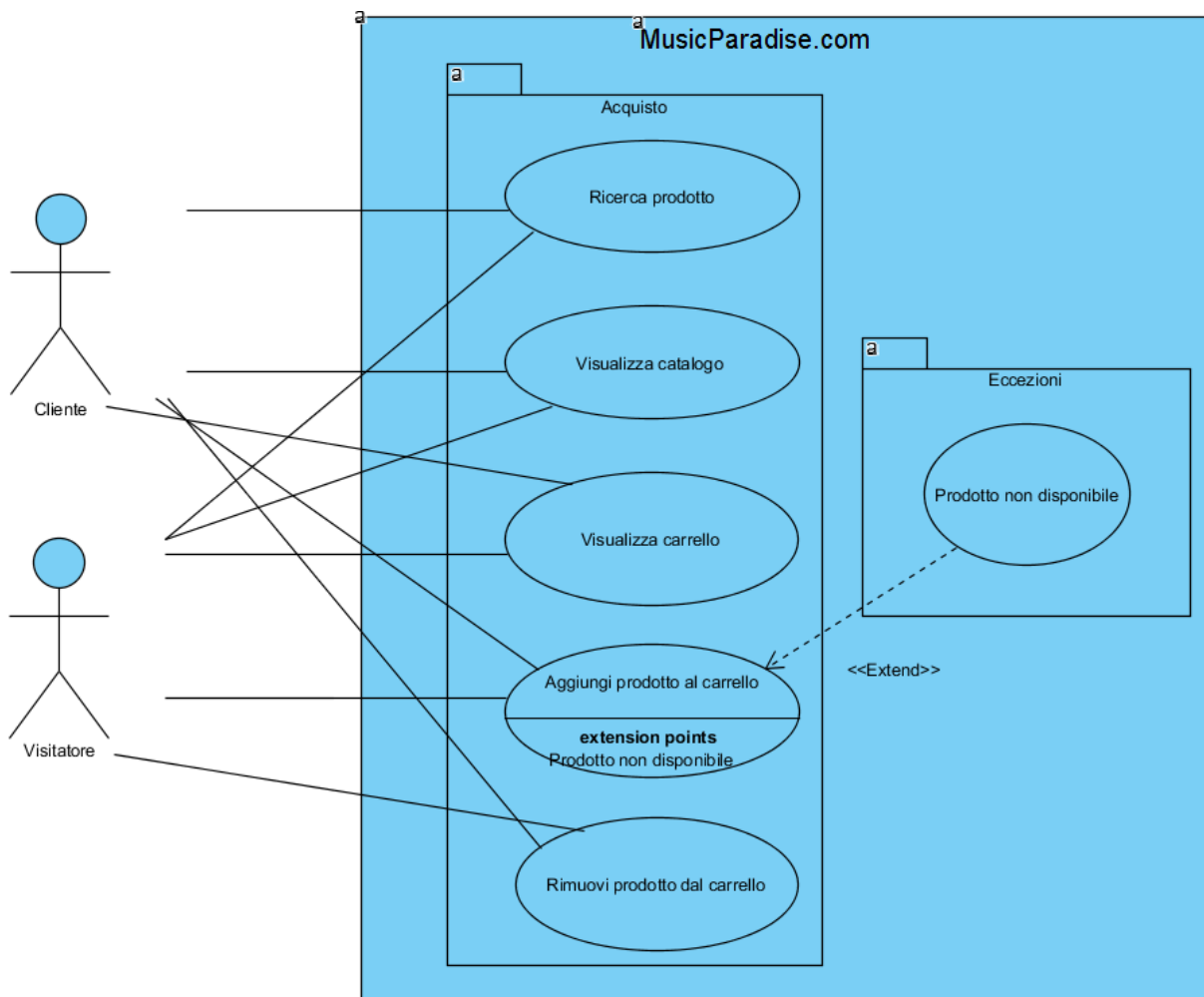
Nome:	Aggiungi nuovo indirizzo
ID:	UC06
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il sistema visualizza una form che richiede l'inserimento di:</p> <ul style="list-style-type: none"> • Nome (obbligatorio) • Cognome (obbligatorio) • Indirizzo (obbligatorio) • Città (obbligatorio) • Cap (obbligatorio) • Telefono (obbligatorio) <p>B. Il cliente riempie i campi e sottomette la form appena compilata al sistema.</p> <p>C. Il sistema verifica che:</p> <ul style="list-style-type: none"> • Tutti i campi obbligatori sono stati compilati in modo corretto
Condizione di entrata:	Il cliente clicca sul bottone aggiungi indirizzo
Condizione di uscita:	Il nuovo indirizzo è stato salvato
Estensione:	I dati inseriti non sono corretti. (Caso d'uso datiErrati)

Nome:	Aggiungi nuova carta
ID:	UC07
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il sistema visualizza una form che richiede l'inserimento di:</p> <ul style="list-style-type: none"> • Numero carta (obbligatorio) • Nome proprietario (obbligatorio) • Scadenza (obbligatorio) • CVC (obbligatorio) <p>B. Il cliente riempie i campi e sottomette la form appena compilata al sistema.</p> <p>C. Il sistema verifica che:</p> <ul style="list-style-type: none"> • Tutti i campi obbligatori sono stati compilati in modo corretto
Condizione di entrata:	Il cliente clicca sul bottone aggiungi indirizzo
Condizione di uscita:	Il nuovo indirizzo è stato salvato
Estensione:	I dati inseriti non sono corretti. (Caso d'uso datiErrati)

Nome:	Rimuovi indirizzo di spedizione
ID:	UC08
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il cliente clicca sul bottone rimuovi associato all'indirizzo</p> <p>B. Il sistema visualizza la tabella con gli indirizzi ancora associati al cliente</p>
Condizione di entrata:	Il cliente effettua il login al sistema e clicca sulla sezione il mio profilo
Condizione di uscita:	l'indirizzo viene rimosso dal database
Estensione:	

Nome:	Rimuovi carta di credito
ID:	UC09
Partecipante:	Cliente
Flusso degli eventi:	<p>A. Il cliente clicca sul bottone rimuovi associato alla carta</p> <p>B. Il sistema visualizza la tabella con le carte ancora associate al cliente</p>
Condizione di entrata:	Il cliente effettua il login al sistema e clicca sulla sezione il mio profilo
Condizione di uscita:	La carta viene rimossa dal database
Estensione:	

Nome:	Checkout
ID:	UC10
Partecipante:	Cliente
Flusso degli eventi:	<ul style="list-style-type: none"> A. Il cliente si sposta nella sezione carrello. B. Il sistema mostra i prodotti nel carrello. C. Il cliente clicca sul bottone "Checkout". D. Il cliente seleziona un indirizzo di spedizione tra quelli memorizzati oppure ne inserisce uno nuovo (caso d'uso Aggiungi nuovo Indirizzo) E. Il cliente seleziona una carta di credito tra quelle memorizzate, oppure ne inserisce una nuova (caso d'uso Aggiungi nuova carta) A. Il cliente clicca sul bottone "conferma ordine". B. Il sistema verifica che: <ul style="list-style-type: none"> • Tutti i campi sono riempiti in modo corretto. C. Il sistema notifica l'avvenuto acquisto.
Condizione di entrata:	<p>Il cliente effettua il login al sistema.</p> <p>Il cliente deve avere almeno un prodotto nel carrello.</p> <p>Il cliente deve trovarsi nella pagina del carrello</p>
Condizione di uscita:	Il sistema notifica l'avvenuto acquisto del prodotto e reindirizza alla home del sito.
Estensione:	<p>I dati inseriti non sono corretti.</p> <p>(Caso d'uso dati errati)</p>



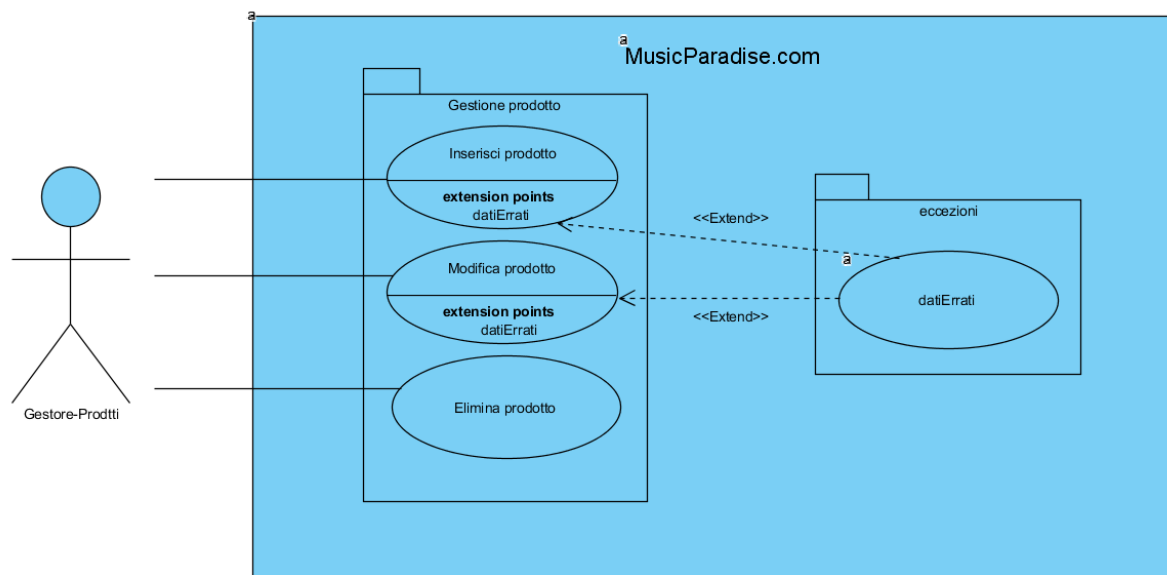
Nome:	Ricerca prodotto
ID:	UC11
Partecipante:	Cliente/Visitatore
Flusso degli eventi:	D. Il cliente/visitatore inserisce nella barra delle ricerche una keyword per la ricerca E. Il sistema mostra a schermo una lista di prodotti correlati alla ricerca. F. Il cliente/visitatore sceglie il prodotto che ha ricercato e clicca sul bottone visualizza prodotto. G. Il sistema mostra la pagina del prodotto.
Condizione di entrata:	Il cliente/visitatore si posiziona nell'home del sistema.
Condizione di uscita:	Il cliente visualizza la pagina del prodotto ricercato.
Estensione:	

Nome:	Aggiungi prodotto al carrello
ID:	UC12
Partecipante:	Cliente/Visitatore
Flusso degli eventi:	A. Include caso d'uso Ricerca prodotto B. Il cliente/visitatore clicca sul bottone aggiungi al carrello. C. Il sistema notifica che nel carrello è stato inserito il prodotto.
Condizione di entrata:	Il cliente/visitatore si posizione nell'home del sistema.
Condizione di uscita:	Il sistema notifica l'avvenuto aggiunta del prodotto al carrello.
Estensione:	Il sistema notifica che il prodotto non è disponibile. (Caso d'uso prodotto non disponibile) A. Se il prodotto non è disponibile: <ul style="list-style-type: none"> Il sistema non permette di aggiungere il prodotto al carrello.

Nome:	Rimuovi prodotto dal carrello
ID:	UC013
Partecipante:	Cliente/visitatore
Flusso degli eventi:	A. Il cliente/visitatore clicca sul bottone rimuovi del prodotto B. Il sistema notifica la rimozione del prodotto dal carrello e visualizza il nuovo totale
Condizione di entrata:	Il cliente/visitatore clicca sulla sezione carrello
Condizione di uscita:	Il prodotto è stato eliminato dal carrello
Estensione:	

Nome:	Visualizza Catalogo
ID:	UC14
Partecipante:	Cliente/visitatore
Flusso degli eventi:	A. Il cliente/visitatore clicca sul bottone Catalogo B. Il sistema mostra a schermo tutti i prodotti presenti nel catalogo che il cliente può acquistare.
Condizione di entrata:	Il cliente/visitatore clicca sulla sezione catalogo
Condizione di uscita:	Il cliente/visitatore visualizza il catalogo
Estensione:	

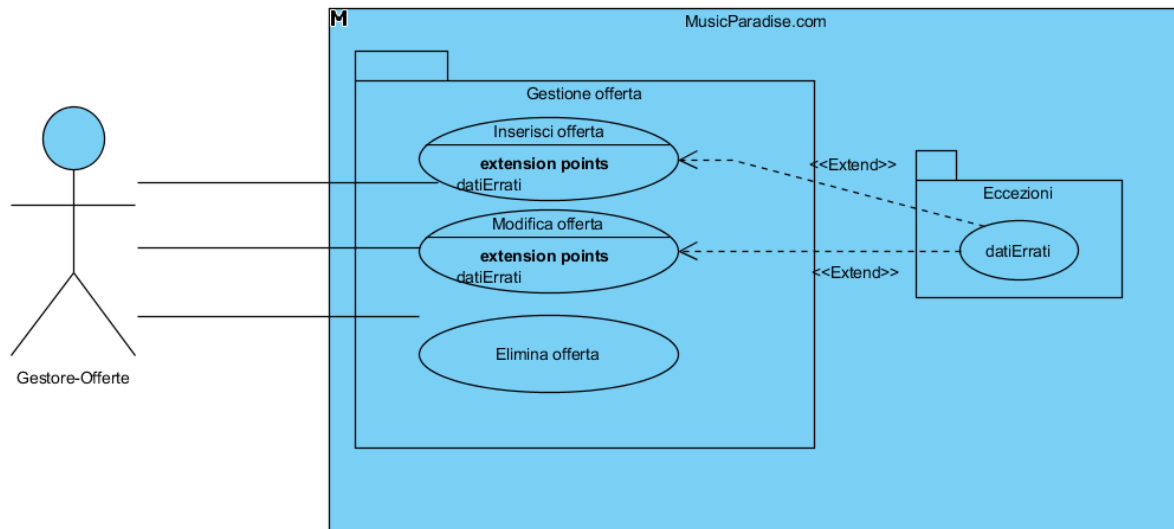
Nome:	Visualizza carrello
ID:	UC15
Partecipante:	Cliente/visitatore
Flusso degli eventi:	A. Il sistema mostra a schermo lo stato del carrello con i relativi prodotti se presenti.
Condizione di entrata:	Il cliente/visitatore clicca sull'icona del carrello
Condizione di uscita:	Il cliente/visitatore visualizza il carrello
Estensione:	



Nome:	Aggiungi prodotto
ID:	UC16
Partecipante:	Gestore-prodotti
Flusso degli eventi:	<p>A. Il gestore-prodotti clicca sulla sezione aggiungi prodotto.</p> <p>B. Il sistema visualizza una form che richiede l'inserimento di:</p> <ul style="list-style-type: none"> • Nome prodotto (obbligatorio) • Marca prodotto (obbligatorio) • Tipo di prodotto (obbligatorio) • Prezzo del prodotto (obbligatorio) • Descrizione (obbligatorio) <p>C. Il gestore-prodotti inserisce le informazioni relative al prodotto da aggiungere e sottomette la form appena compilata al sistema.</p> <p>D. Il sistema verifica che:</p> <p>a. Tutti i campi obbligatori sono stati compilati</p> <p>E. Il sistema salva i dati del prodotto nel database.</p> <p>Il sistema mostra una schermata che informa il gestore dell'avvenuta aggiunta del prodotto.</p>
Condizione di entrata:	il gestore-prodotti effettua il login al sistema.
Condizione di uscita:	Il prodotto è aggiunto al catalogo.
Estensione:	I dati inseriti non sono corretti. (Caso d'uso dati errati)

Nome:	Modifica prodotto
ID:	UC17
Partecipante:	Gestore-prodotti
Flusso degli eventi:	<p>A. Il sistema visualizza la tabella dei prodotti.</p> <p>B. Il gestore-prodotti clicca sul bottone modifica relativo al prodotto da modificare.</p> <p>C. Il sistema visualizza una form che contiene:</p> <ul style="list-style-type: none"> • Nome prodotto • Marca prodotto • Tipo di prodotto • Prezzo del prodotto • Descrizione <p>D. Il gestore-prodotti modifica le informazioni relative al prodotto e sottomette la form appena compilata al sistema.</p> <p>E. Il sistema verifica che:</p> <ul style="list-style-type: none"> • Tutti i campi modificati sono stati compilati nel modo corretto. <p>F. Il sistema salva i dati del prodotto nel database se corretti.</p> <p>G. Il sistema mostra una schermata che informa il gestore dell'avvenuta modifica del prodotto.</p>
Condizione di entrata:	il gestore-prodotti effettua il login al sistema.
Condizione di uscita:	Le informazioni del prodotto sono modificate.
Estensione:	I dati inseriti non sono corretti. (Caso d'uso dati errati)

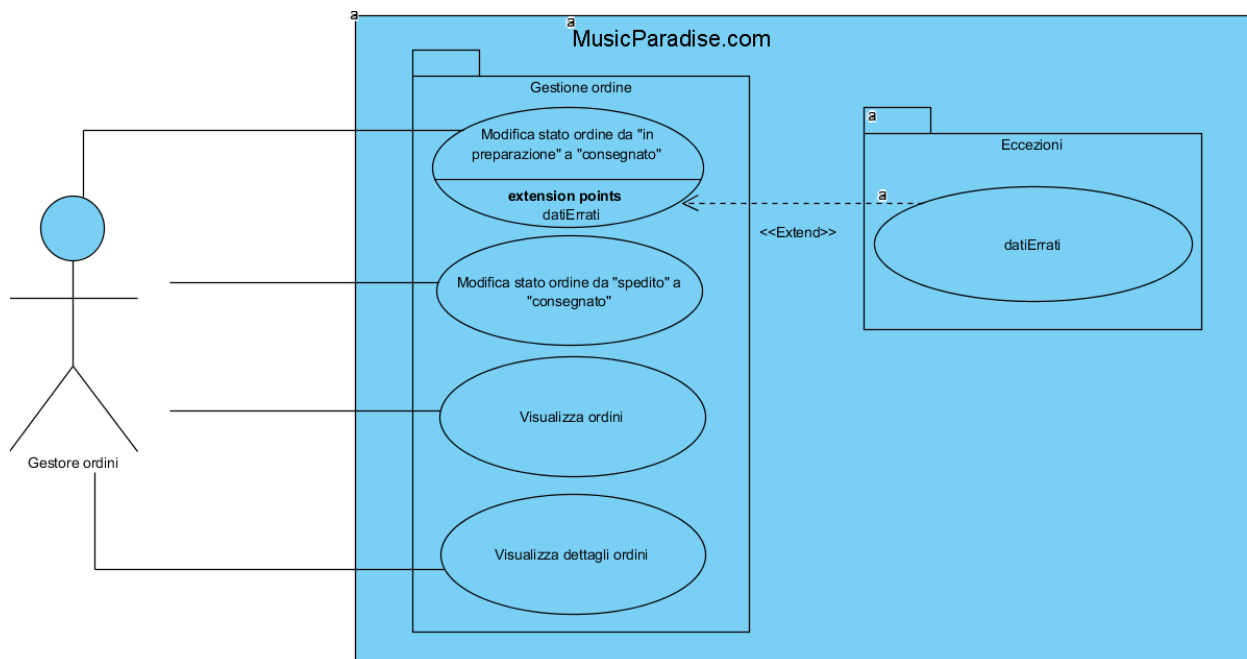
Nome:	Elimina prodotto
ID:	UC18
Partecipante:	Gestore-prodotti
Flusso degli eventi:	<p>A. Il gestore-prodotti clicca sulla sezione elimina prodotto.</p> <p>B. Il sistema visualizza una tabella contenente i prodotti.</p> <p>C. Il gestore-prodotti sceglie il prodotto da eliminare</p> <p>D. Il sistema aggiorna il database.</p> <p>E. Il sistema mostra una schermata che informa il gestore dell'avvenuta rimozione del prodotto.</p>
Condizione di entrata:	il gestore-prodotti effettua il login al sistema.
Condizione di uscita:	Il prodotto è eliminato dal catalogo.



Nome:	Inserisci offerta
ID:	UC02
Partecipante:	Gestore-offerte
Flusso degli eventi:	<p>A. Il gestore-offerte clicca sulla sezione aggiungi offerta.</p> <p>B. Il sistema mostra una tabella con tutti i prodotti disponibili.</p> <p>C. Il gestore-offerte seleziona il prodotto su cui inserire l'offerta.</p> <p>D. Il sistema visualizza una form che richiede l'inserimento di:</p> <ul style="list-style-type: none"> • Percentuale di sconto (obbligatorio) • Data di inizio e di fine dell'offerta (obbligatorio) <p>E. Il sistema verifica che:</p> <ul style="list-style-type: none"> • I campi obbligatori sono stati compilati <p>F. Il sistema se corretti, salva i dati nel database</p> <p>G. Il sistema mostra una schermata che informa al gestore del corretto inserimento dell'offerta.</p>
Condizione di entrata:	il gestore-offerte effettua il login al sistema
Condizione di uscita:	L'offerta sul prodotto è attiva.
Estensione:	I dati inseriti non sono corretti. (Caso d'uso dati errati)

Nome:	Elimina offerta
ID:	UC20
Partecipante:	Gestore-offerte
Flusso degli eventi:	<p>A. Il gestore-offerte clicca sulla sezione offerte.</p> <p>B. Il sistema visualizza una tabella contenente le offerte.</p> <p>C. Il gestore-offerte sceglie l'offerta da eliminare</p> <p>D. Il sistema aggiorna il database.</p> <p>E. Il sistema mostra una schermata che informa il gestore dell'avvenuta rimozione dell'offerta.</p>
Condizione di entrata:	il gestore-offerte effettua il login al sistema.
Condizione di uscita:	L'offerta è eliminato dal sistema.

Nome:	Visualizza lista offerte
ID:	UC021
Partecipante:	Gestore-offerte
Flusso degli eventi:	A. Il gestore-offerte clicca sulla sezione lista offerte. B. Il sistema visualizza una tabella contenente le offerte create.
Condizione di entrata:	il gestore-offerte effettua il login al sistema.
Condizione di uscita:	Il sistema visualizza una tabella contenente le offerte create.

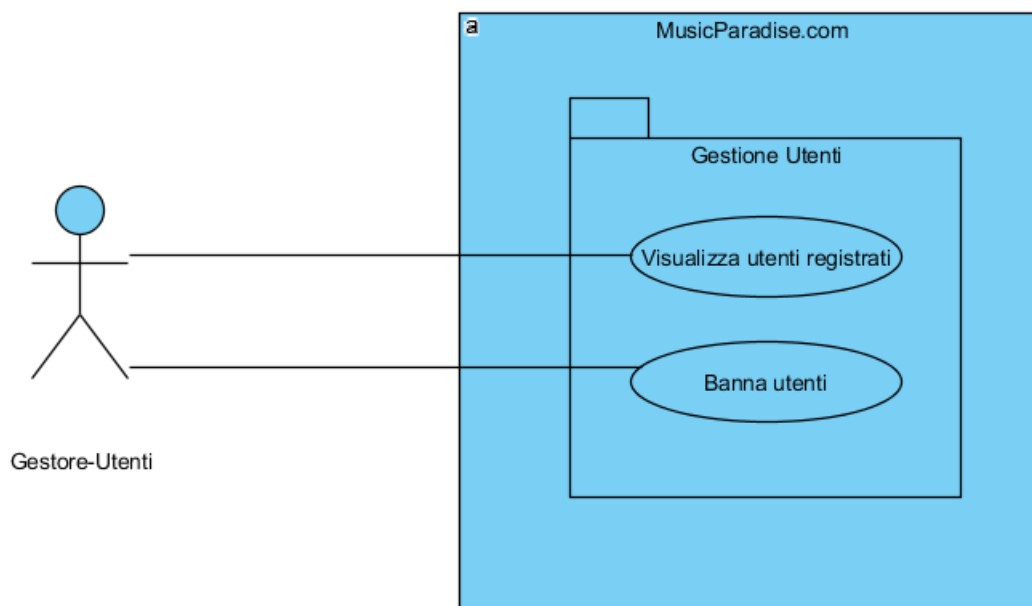


Nome:	Visualizza ordini clienti
ID:	UC22
Partecipante:	Gestore-ordini
Flusso degli eventi:	A. Il sistema visualizza tre tabella contenente gli ordini dei clienti, quelli con lo stato "in preparazione", con lo stato "spedito" e con lo stato "consegnato"
Condizione di entrata:	il gestore-ordini effettua il login al sistema.
Condizione di uscita:	Il sistema visualizza le tabelle contenente gli ordini dei clienti.

Nome:	modifica stato ordine (da in preparazione a spedito)
ID:	UC23
Partecipante:	Gestore-ordine
Flusso degli eventi:	<p>A. Il gestore-ordini clicca sulla tabella ordini “in preparazione”.</p> <p>B. Il sistema visualizza una tabella contenente gli ordini effettuati dai clienti.</p> <p>C. Il gestore-ordini clicca su modifica stato</p> <p>D. Il sistema richiede l’inserimento di:</p> <ul style="list-style-type: none"> • Nome corriere • Codice tracking • Data consegna <p>E. Il gestore-ordini inserisce le informazioni e sottomette la form appena compilata al sistema.</p> <p>F. Il sistema verifica che:</p> <p>a. Tutti i campi modificati sono stati compilati nel modo corretto.</p> <p>G. Il sistema salva i dati nel database se corretti.</p> <p>H. Il sistema mostra una tabella presenti gli ordini spediti.</p>
Condizione di entrata:	il gestore-ordini effettua il login al sistema.
Condizione di uscita:	L’ordine è stato spedito

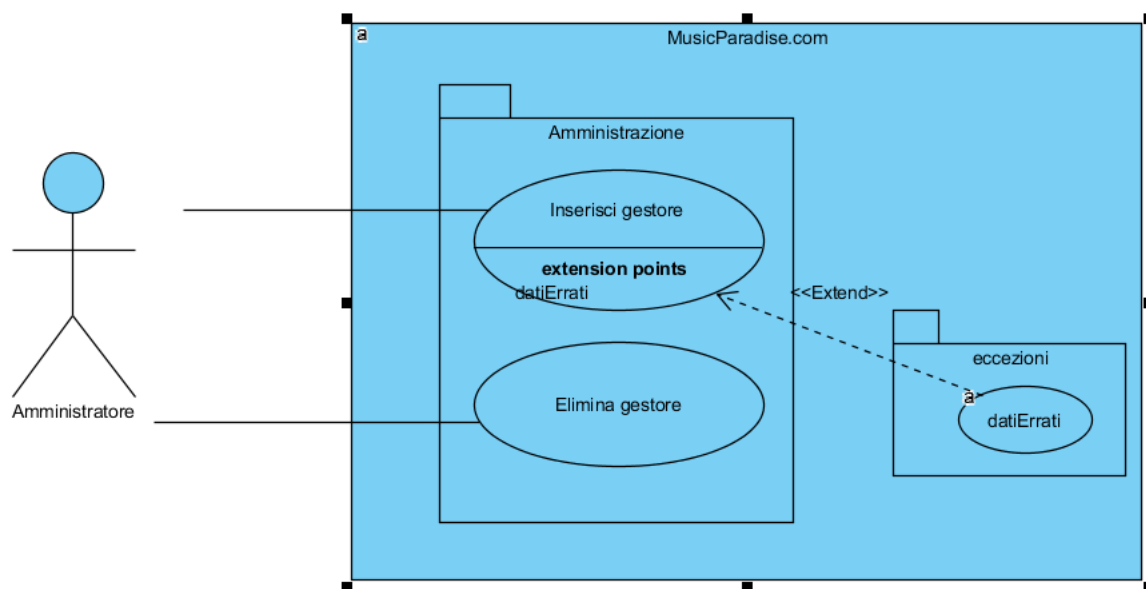
Nome:	modifica stato ordine (da in spedito a consegnato)
ID:	UC24
Partecipante:	Gestore-ordine
Flusso degli eventi:	<p>A. Il gestore-ordini clicca sulla tabella ordini “spediti”</p> <p>B. Il sistema visualizza una tabella contenente gli ordini effettuati dai clienti con lo stato “spedito”.</p> <p>C. Il gestore-ordini clicca sul bottone consegnato</p> <p>D. Il sistema aggiorna lo stato dell’ordine e mostra una schermata dove ci sono gli ordini consegnati</p>
Condizione di entrata:	il gestore-ordini effettua il login al sistema.
Condizione di uscita:	L’ordine è stato consegnato

Nome:	Visualizza dettagli ordini clienti
ID:	UC25
Partecipante:	Gestore-ordini
Flusso degli eventi:	<p>A. Il sistema visualizza tre tabella contenente gli ordini dei clienti, quelli con lo stato “in preparazione”, con lo stato “spedito” e con lo stato “consegnato”</p> <p>B. Il gestore clicca sul bottone dettagli</p> <p>C. Il sistema visualizza una pagina dove ci sono i dettagli dell’ordine.</p>
Condizione di entrata:	il gestore-ordini effettua il login al sistema.
Condizione di uscita:	Il sistema visualizza i prodotti dell’ordine



Nome:	Visualizza utenti registrati
ID:	UC26
Partecipante:	Gestore-utenti
Flusso degli eventi:	A. Il gestore-utenti clicca sulla sezione utenti. B. Il sistema visualizza una tabella contenente gli utenti registrati al sistema
Condizione di entrata:	il gestore-utenti effettua il login al sistema.
Condizione di uscita:	Il sistema visualizza una tabella contenente gli utenti registrati.

Nome:	Banna utenti
ID:	UC27
Partecipante:	Gestore-utenti
Flusso degli eventi:	A. Il gestore-utenti clicca sulla sezione utenti. B. Il sistema visualizza una tabella contenente gli utenti registrati al sistema C. Il gestore-utenti clicca sul bottone "banna" dell'utente da bannare D. Il sistema banna l'utente.
Condizione di entrata:	il gestore-utenti effettua il login al sistema.
Condizione di uscita:	Il cliente è stato bannato



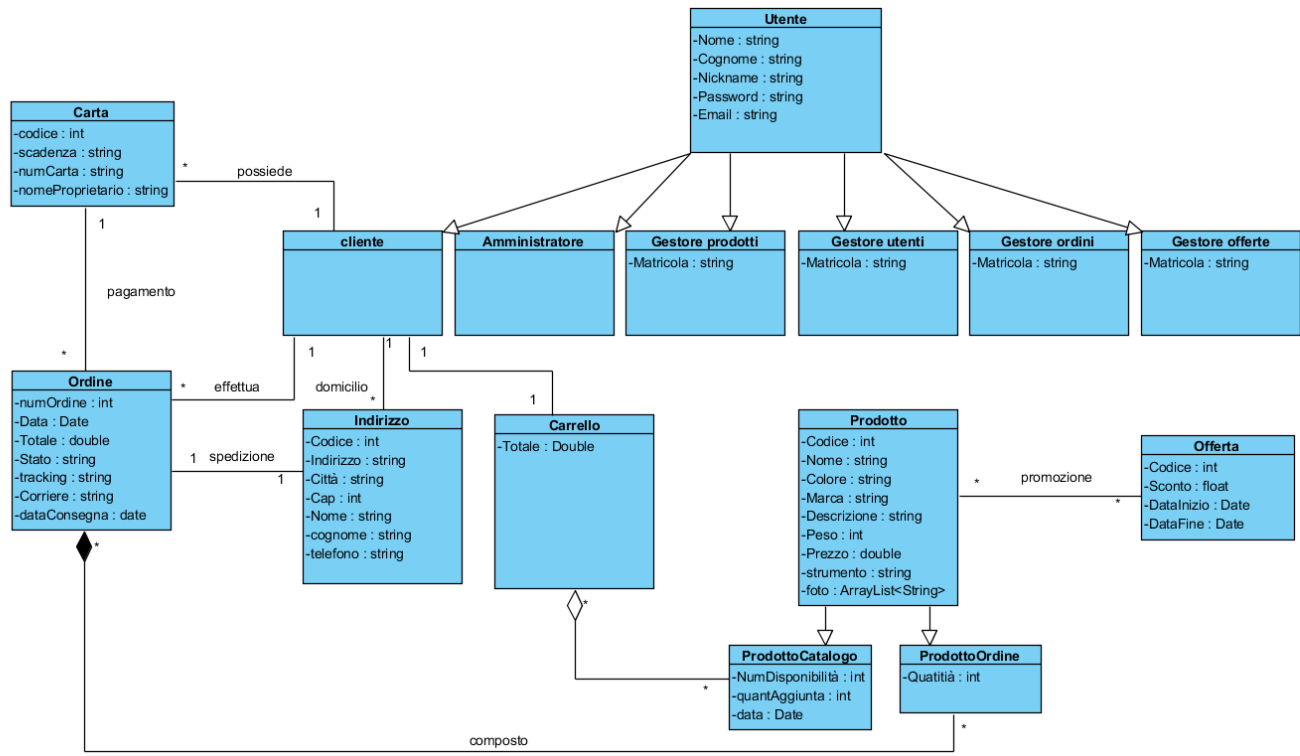
Nome:	Aggiungi gestore
ID:	UC28
Partecipante:	Amministratore
Flusso degli eventi:	<p>A. L'amministratore clicca sul bottone aggiungi gestore.</p> <p>B. Il sistema visualizza un form che richiede l'inserimento di;</p> <ul style="list-style-type: none"> • Nome (obbligatorio) • Cognome (obbligatorio) • Matricola (obbligatorio) • Email (obbligatorio) • Area gestione (obbligatorio) • Nickname (obbligatorio) • Password (obbligatorio) <p>C. Il gestore inserisce le informazioni relative al gestore da aggiungere e sottomette la form appena compilata al sistema.</p> <p>D. Il sistema verifica che:</p> <ul style="list-style-type: none"> • Tutti i campi obbligatori sono stati compilati <p>E. Il sistema salva i dati del gestore nel database.</p> <p>F. Il sistema mostra una schermata che informa l'amministratore dell'avvenuta aggiunta del gestore..</p>
Condizione di entrata:	Amministratore effettua il login al sistema
Condizione di uscita:	Il gestore è aggiunto al sistema
Estensione:	I dati inseriti non sono corretti. (Caso d'uso dati errati)

Nome:	Elimina gestore
ID:	UC29
Partecipante:	Amministratore
Flusso degli eventi:	<ul style="list-style-type: none"> A. L'amministratore clicca sulla sezione gestori B. L'amministratore clicca sul bottone elimina gestore. C. Il sistema mostra una schermata che informa l'amministratore dell'avvenuta rimozione del gestore..
Condizione di entrata:	Amministratore effettua il login al sistema
Condizione di uscita:	Il gestore è eliminato dal sistema
Estensione:	

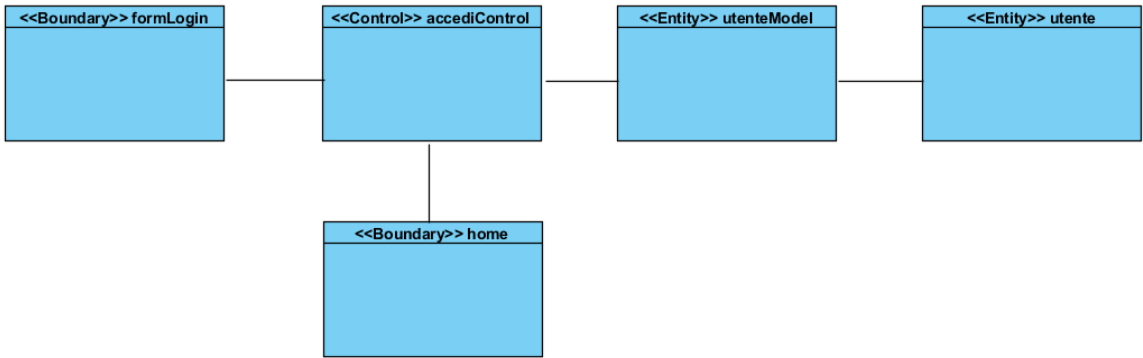
Nome:	dati errati
ID:	UC30
Partecipante:	Sistema
Flusso degli eventi:	<p>Variante 1</p> <ul style="list-style-type: none"> • Sistema: visualizza un messaggio di errore che segnala all'utente che non ha inserito tutti i campi obbligatori. <p>Variante 2</p> <ul style="list-style-type: none"> • Sistema: visualizza un messaggio di errore che segnala all'utente che il nickname è già in uso. <p>Variante 3</p> <ul style="list-style-type: none"> • Sistema: visualizza un messaggio di errore che segnala all'utente che la password deve essere almeno di 8 caratteri. <p>Variante 4</p> <ul style="list-style-type: none"> • Sistema: visualizza un messaggio di errore che segnala all'utente che non ha inserito le credenziali giuste.
Condizione di entrata:	il sistema verifica che i campi obbligatori sono stati compilati in modo corretto
Condizione di uscita:	<ul style="list-style-type: none"> • Sistema: Resta in attesa di una nuova sottomissione della form.

3.4.3. Object model

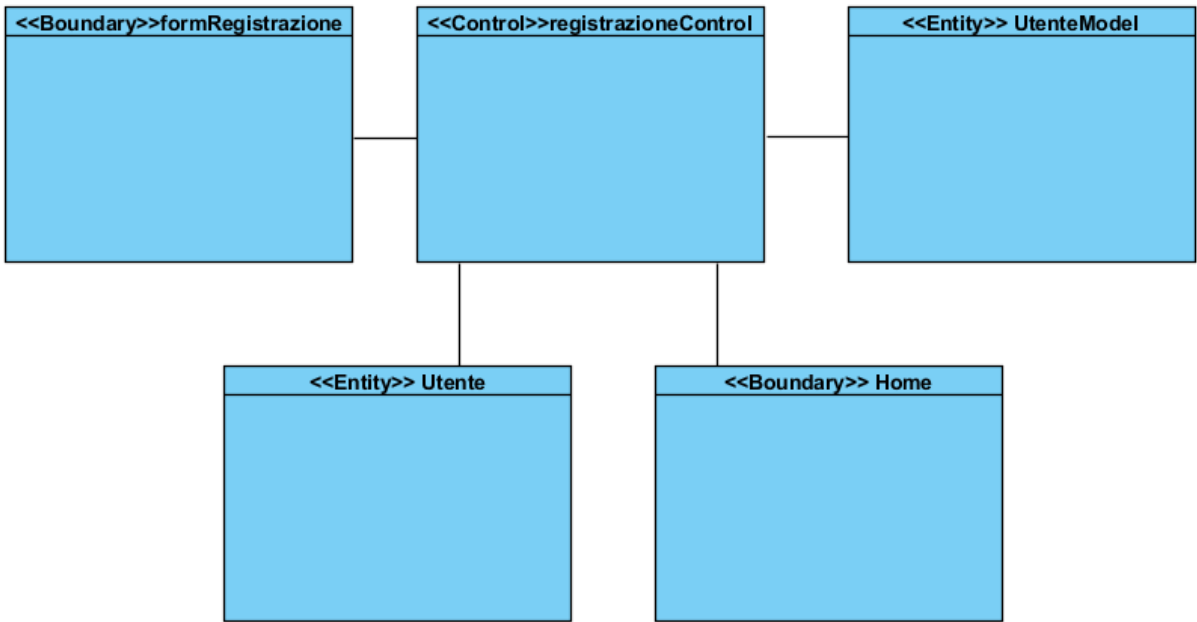
3.4.3.1 Class Diagrams



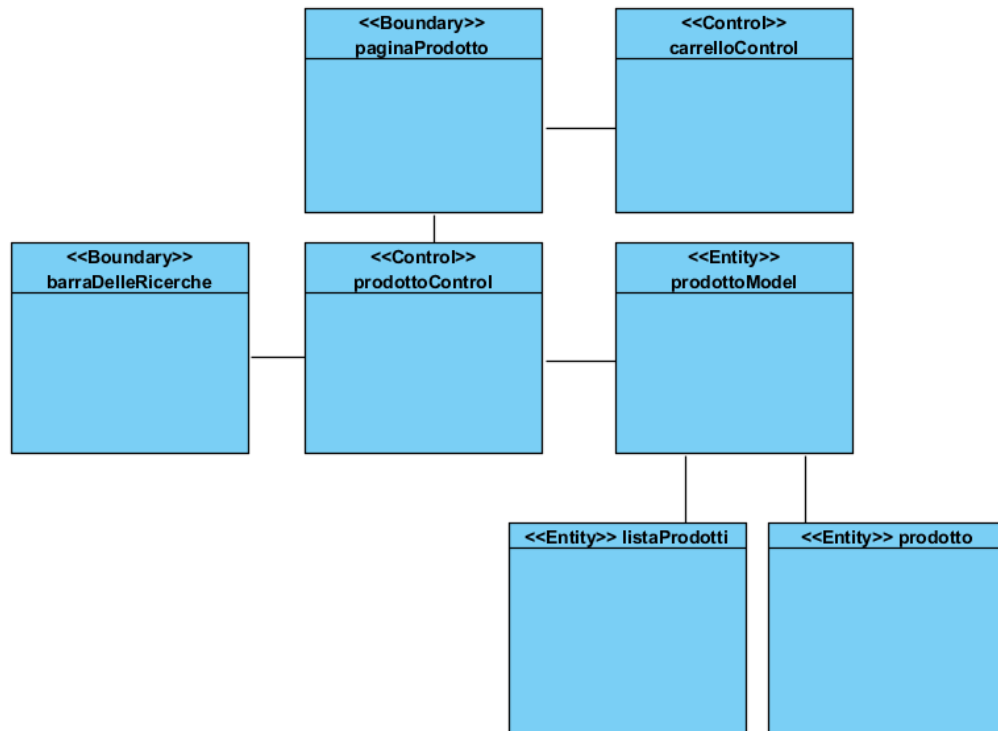
3.4.3.2. Class diagrams Login



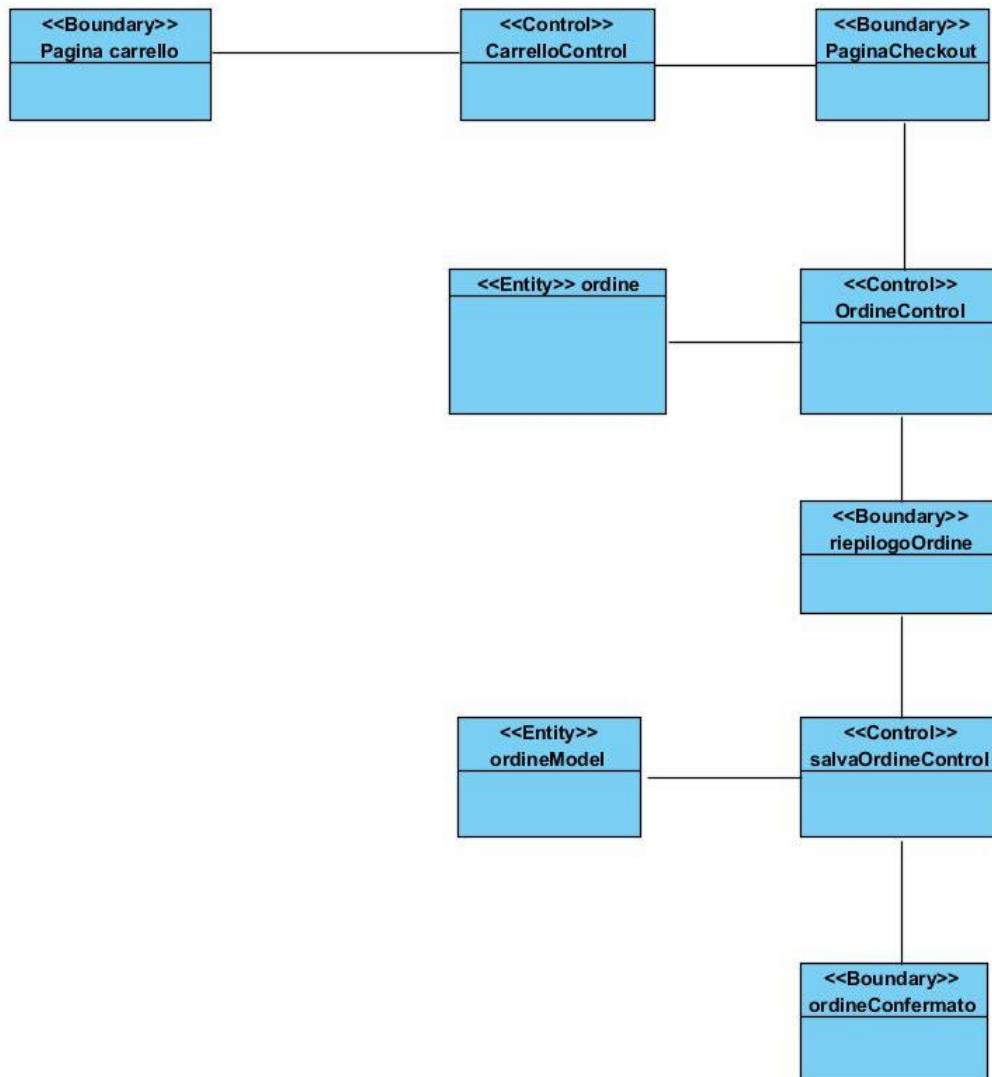
3.4.3.3. Class Diagrams Registrazione



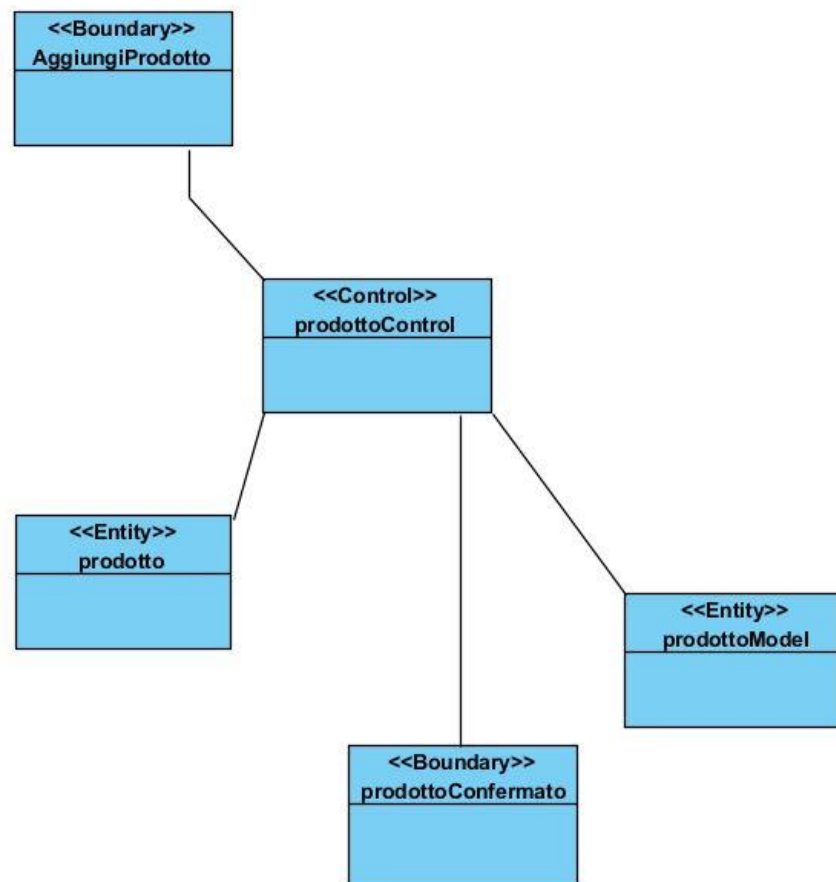
3.4.3.4. Class Diagrams Aggiungi prodotto al carrello



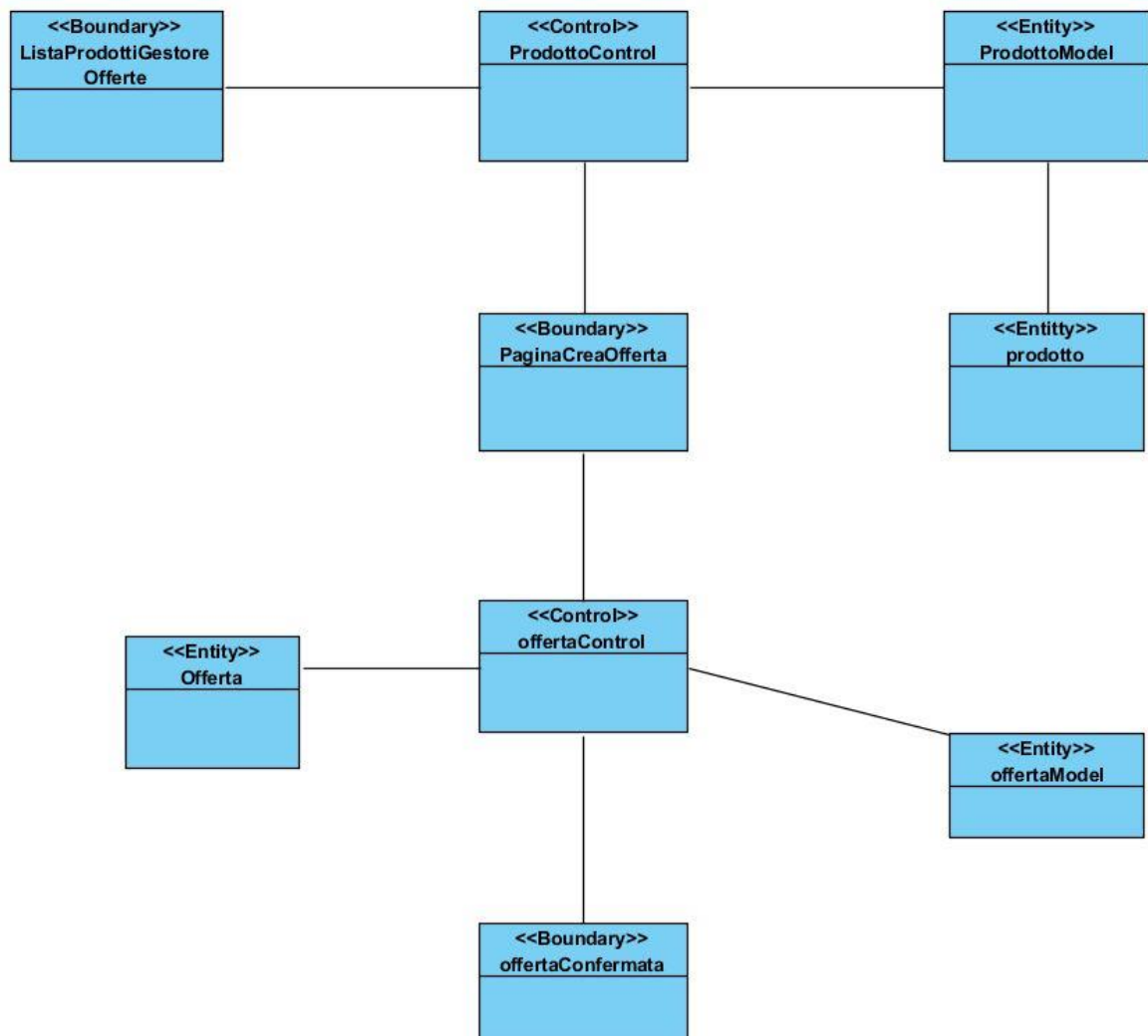
3.4.3.5. Class Diagrams Acquista prodotto



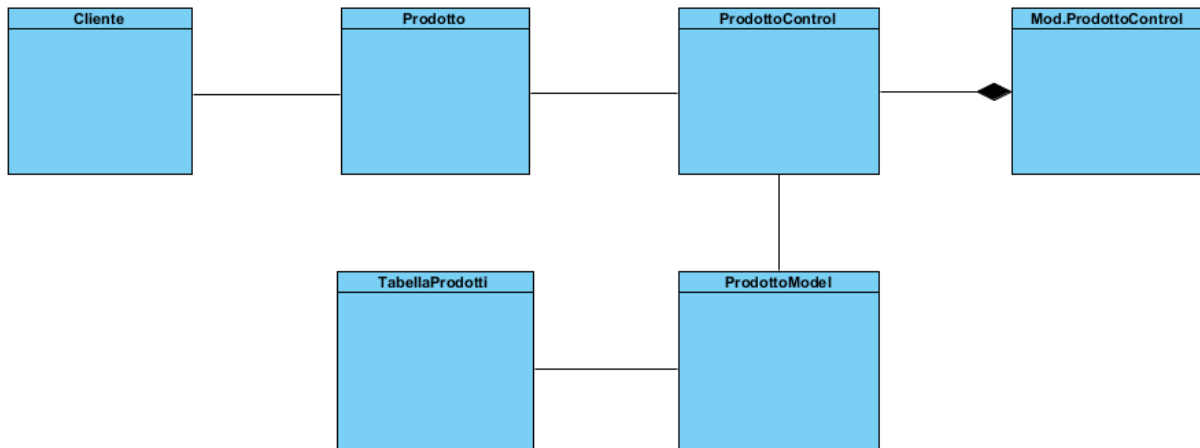
3.4.3.6. Class Diagrams Aggiungi prodotto al catalogo



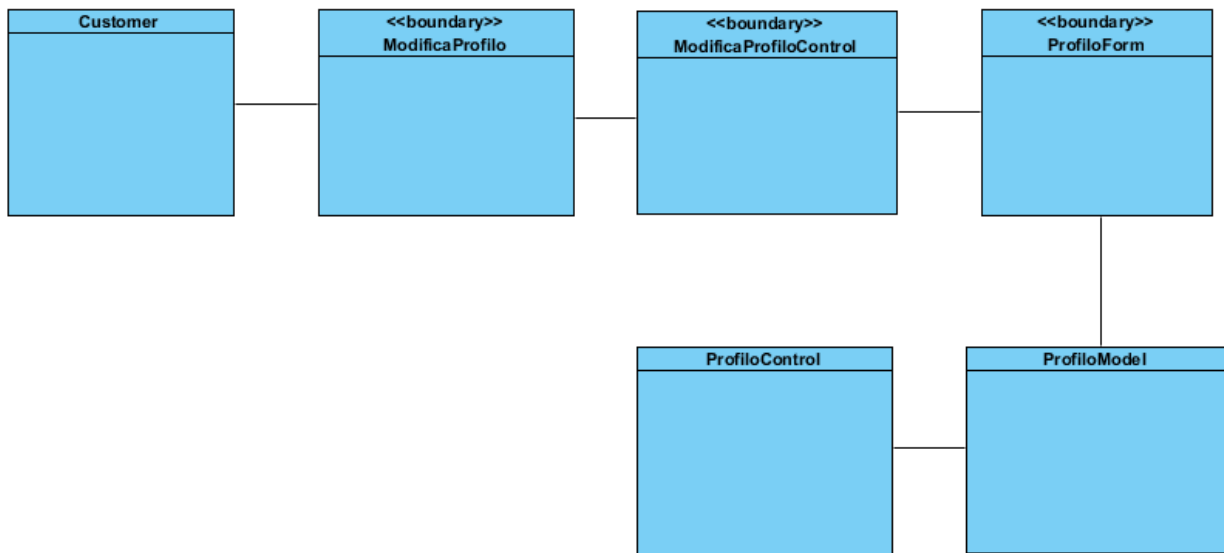
3.4.3.7. Aggiungi offerta prodotto



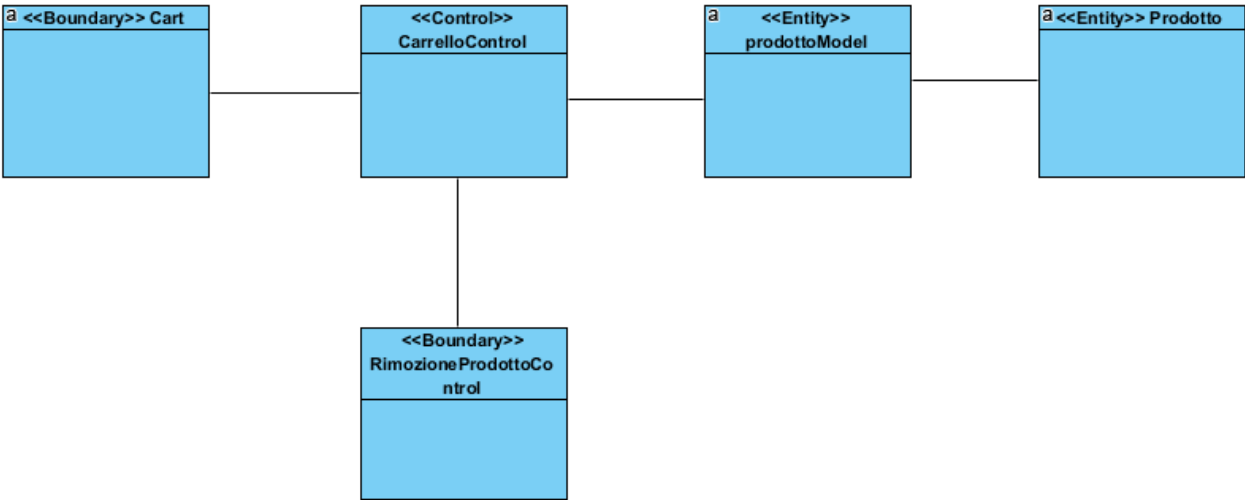
3.4.3.8. Class Diagrams Modifica prodotto



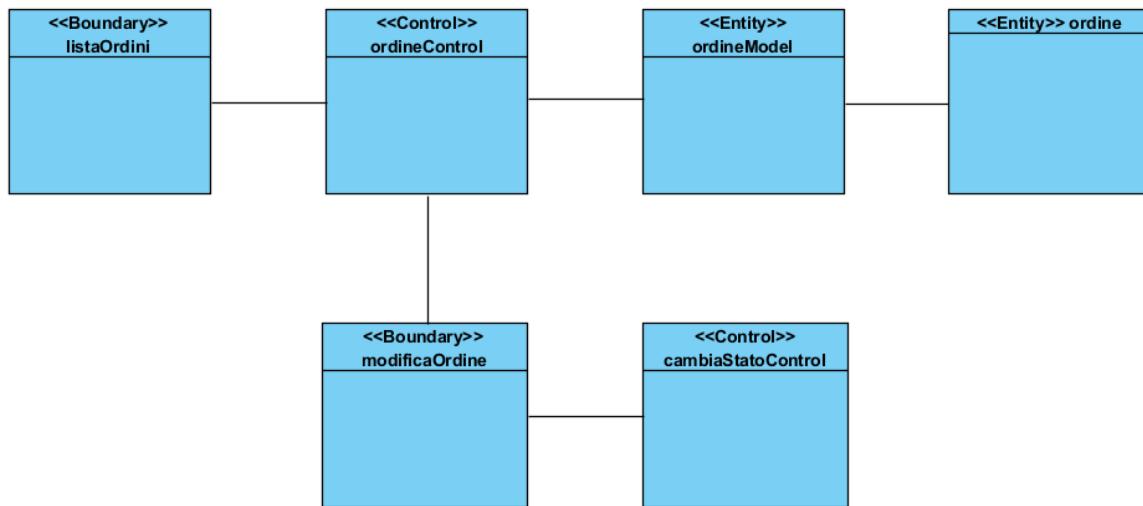
3.4.3.9. Class Diagrams Modifica dati cliente



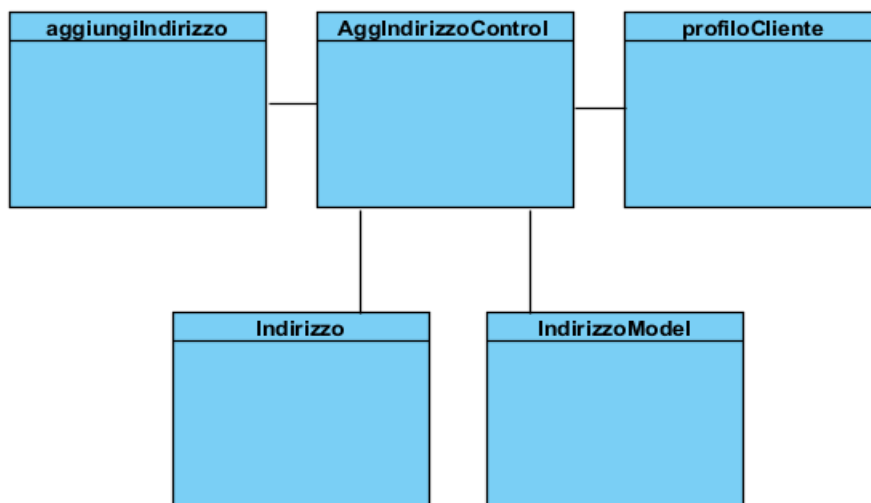
3.4.3.10. Class Diagrams Rimuovi prodotto dal carrello



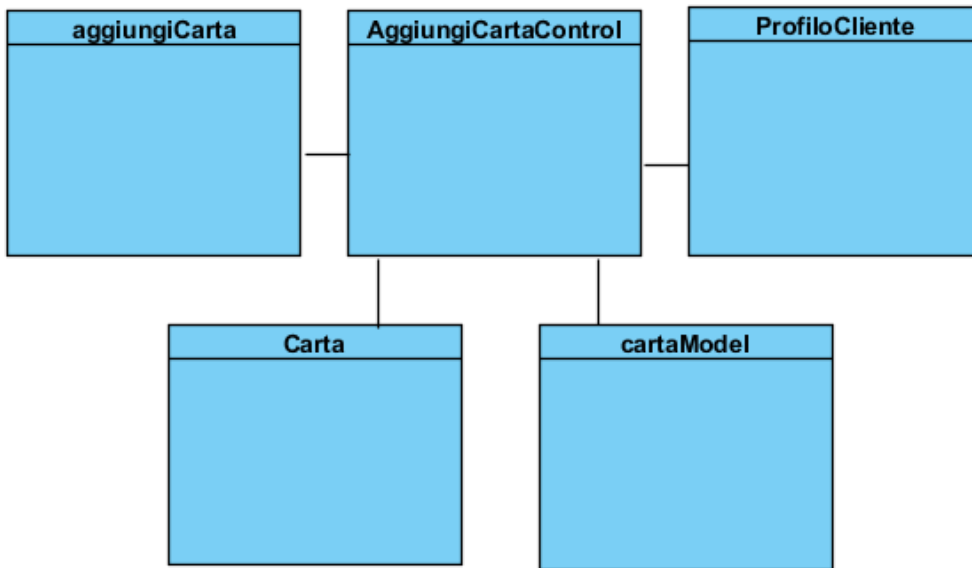
3.4.3.11. Class Diagrams Modifica stato ordine



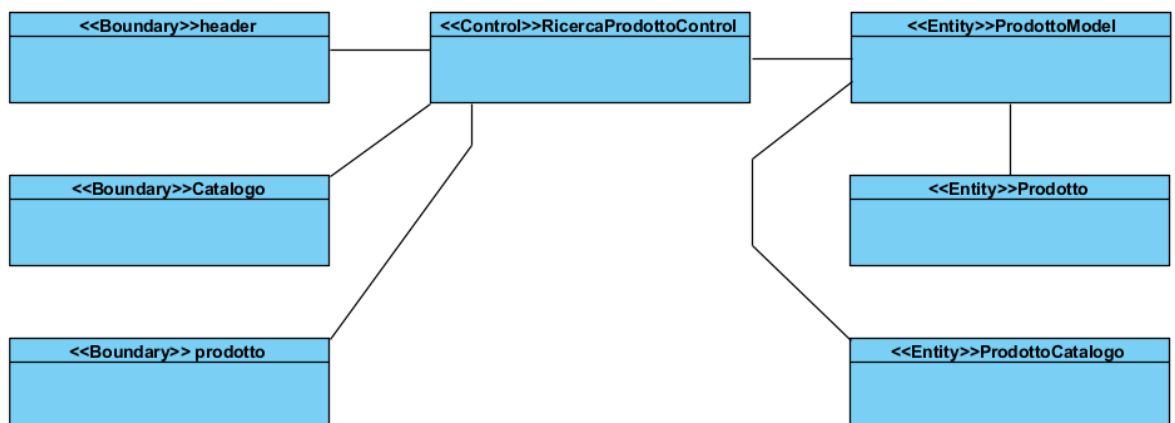
3.4.4.12. Class Diagram Aggiungi indirizzo



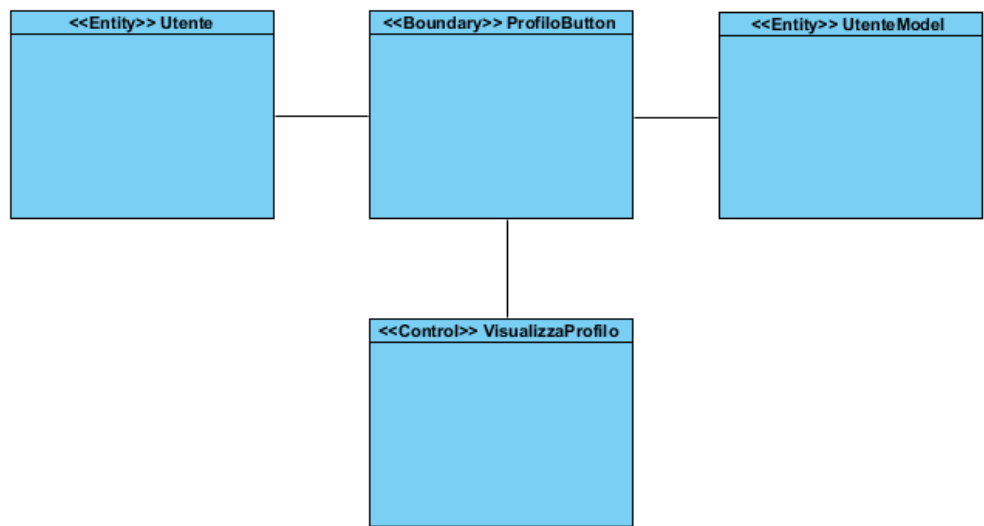
3.4.4.12. Class Diagrams Aggiungi carta



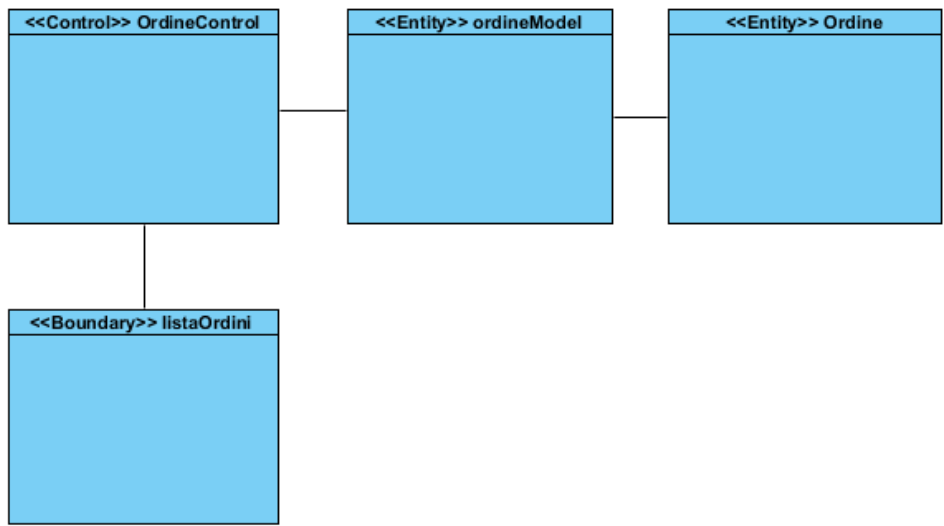
3.4.4.13. Class Diagram Ricerca Prodotto



3.4.4.14. Visualizza Profilo

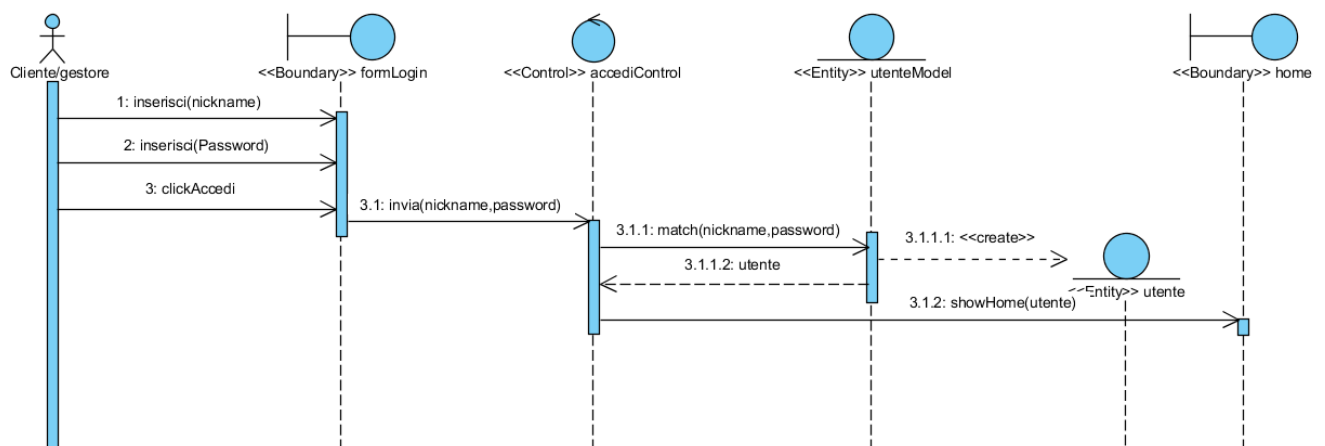


3.4.4.15 Visualizza Ordini



3.4.5. *Dynamic model*

3.4.5.12. Login



Attore:

- **Cliente/Gestore** che inizia il caso d'uso

Boundary object:

- **formLogin**, permette di inserire le credenziali per accedere
- **home**, è la pagina che permette di visualizzare l'home del sistema.

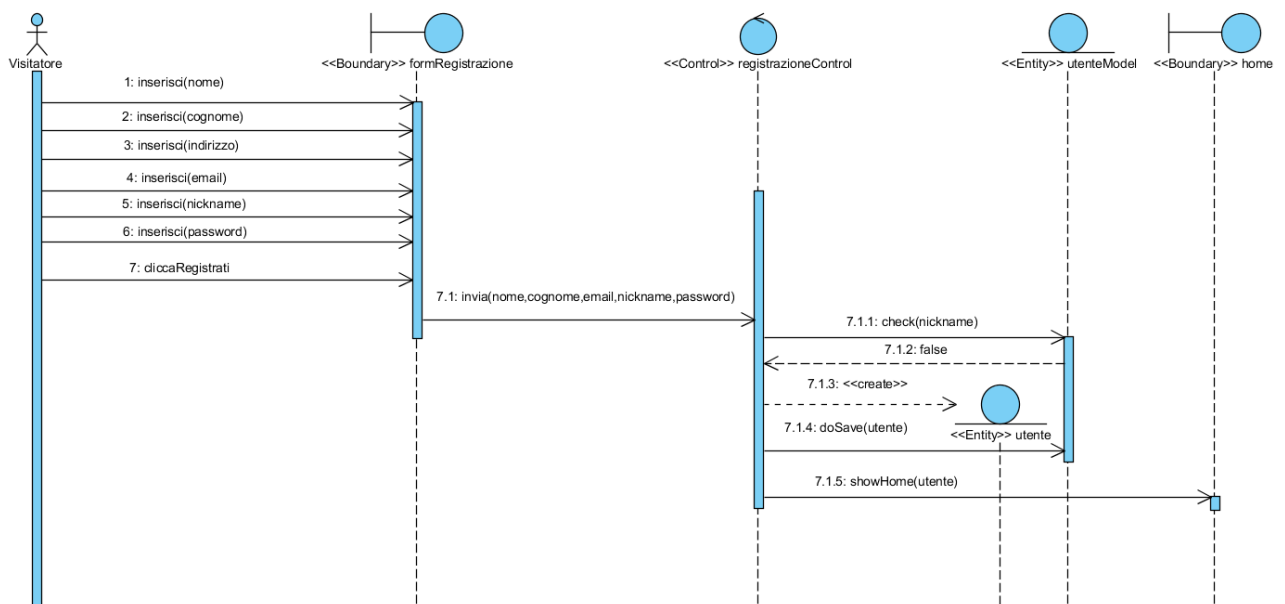
Control object:

- **accediControl**, raccoglie dal <<Boundary>> formLogin e passa le informazioni all'entity, per controllare che le credenziali sono corrette.

Entity object:

- **utenteModel** controlla se l'utente è registrato al sistema.
- **utente**, rappresenta l'informazione nel sistema relative all'utente

3.4.5.13. Registrazione



Attore:

- **Visittore** che inizia il caso d'uso

Boundary object:

- **formRegistrazione**, permette di inserire il nome del prodotto da ricercare
- **home**, è la pagina che permette di visualizzare tutti i prodotti risultati dalla ricerca

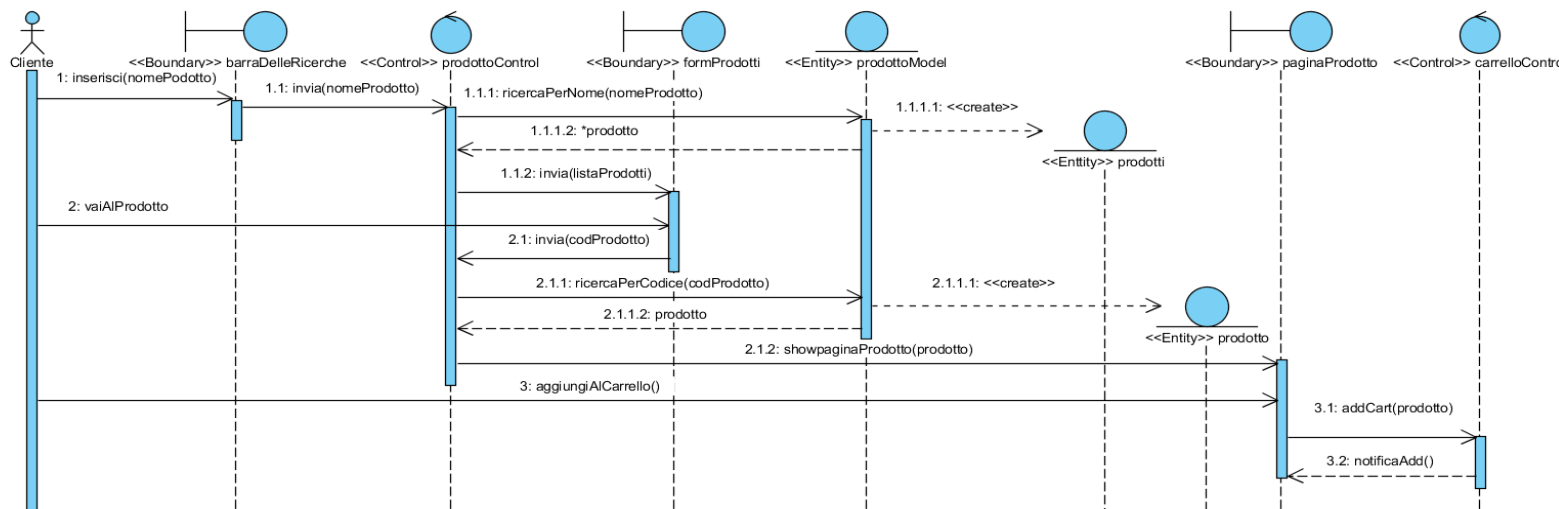
Control object:

- **registrazioneControl**, raccoglie dal <<Boundary>> formRegistrazione le informazioni inserite dal visitatore e crea l'oggetto utente. Il control passa l'oggetto utente all'etity utenteModel.

Entity object:

- **utenteModel** salva l'utente nel database.
- **utente**, rappresenta l'informazione nel sistema relative all'utente.

3.4.4.3. Aggiungi prodotto al carrello



Attore:

- **Cliente** che inizia il caso d'uso

Boundary object:

- **Barra delle ricerche**, permette di inserire il nome del prodotto da ricercare
- **formProdotti**, è la pagina che permette di visualizzare tutti i prodotti risultati dalla ricerca
- **paginaProdotto**, contiene le informazioni del prodotto selezionato e da aggiungere al carrello

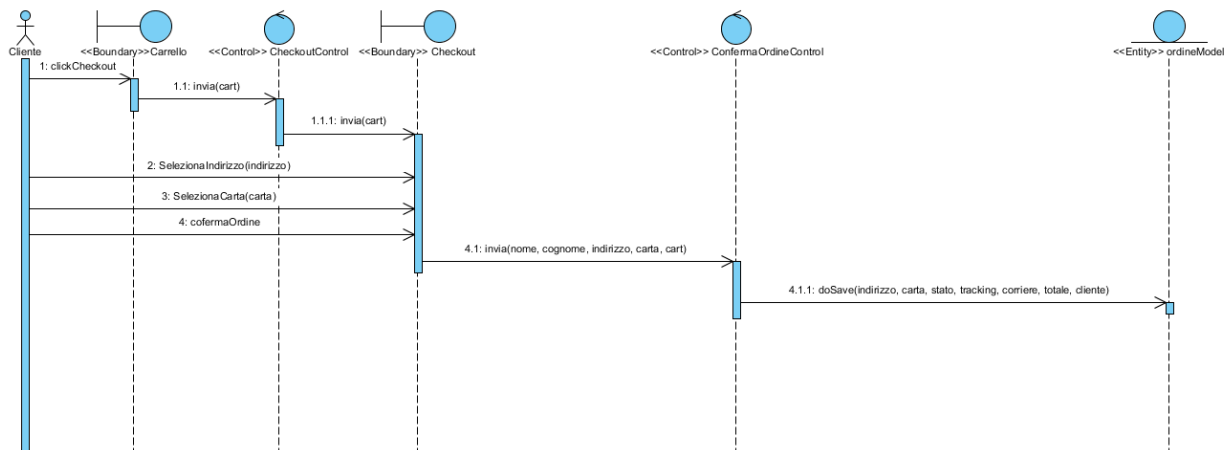
Control object:

- **prodottoControl**, raccoglie dal <<Boundary>> barra delle ricerche il nome del prodotto e passa le informazioni all'entity, da cui riceve la lista dei prodotti. Passa la lista dei prodotti al <<boundary>> formProdotti.
- **carrelloControl** riceve l'oggetto prodotto, aggiunge il prodotto al carrello e notifica l'inserimento del prodotto.

Entity object:

- **prodottoModel** ricerca il prodotto all'interno del database e lo passa al control

3.4.4.4. Acquista prodotto



Attore:

- **Cliente** che inizia il caso d'uso

Boundary object:

- **Carrello**, permette di visualizzare i prodotti presenti nel carrello
- **Checkout**, pagina che permette all'utente di selezionare l'indirizzo di spedizione e la carta di credito, tra quelli memorizzati. Inoltre fa visualizzare il totale dell'ordine.

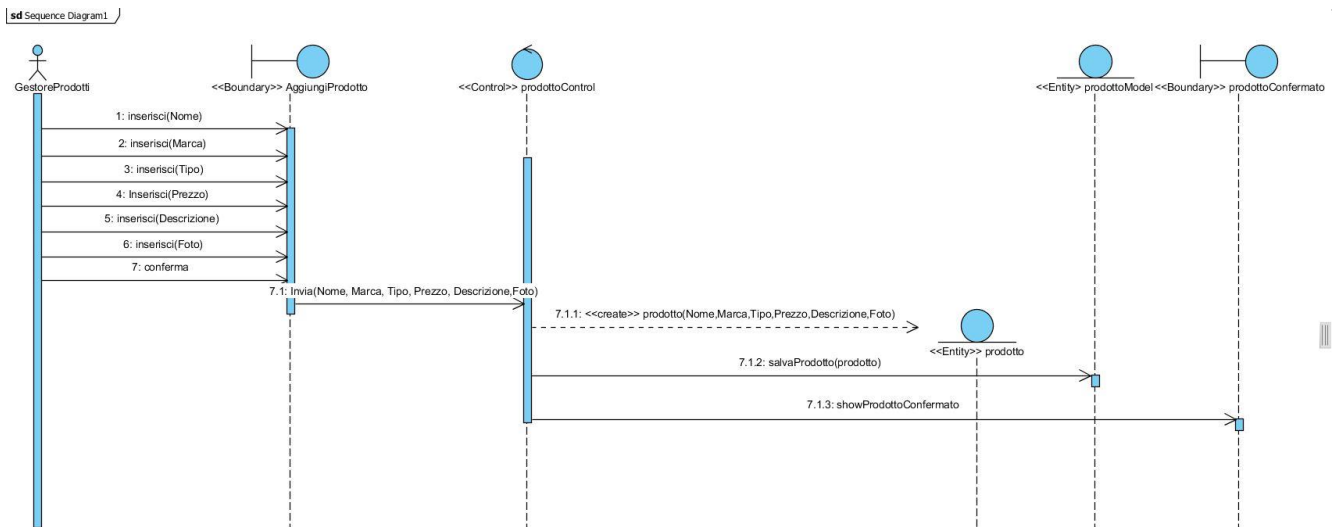
Control object:

- **carrelloControl** invia le informazioni al <<boundary>> Checkout
- **ConfermaOrdineControl** invia il messaggio salvaOrdine(dati) all'entity ordineModel

Entity object:

- **ordineModel** salva l'ordine nel database

3.4.4.5. Gestore Prodotti – Aggiungi Prodotto



Attore:

- **GestoreProdotti**, utilizza il sistema per inserire un Prodotto.

Entity:

- **prodottoModel** gestisce la comunicazione del database.
- **Prodotto**, informazioni sul prodotto.

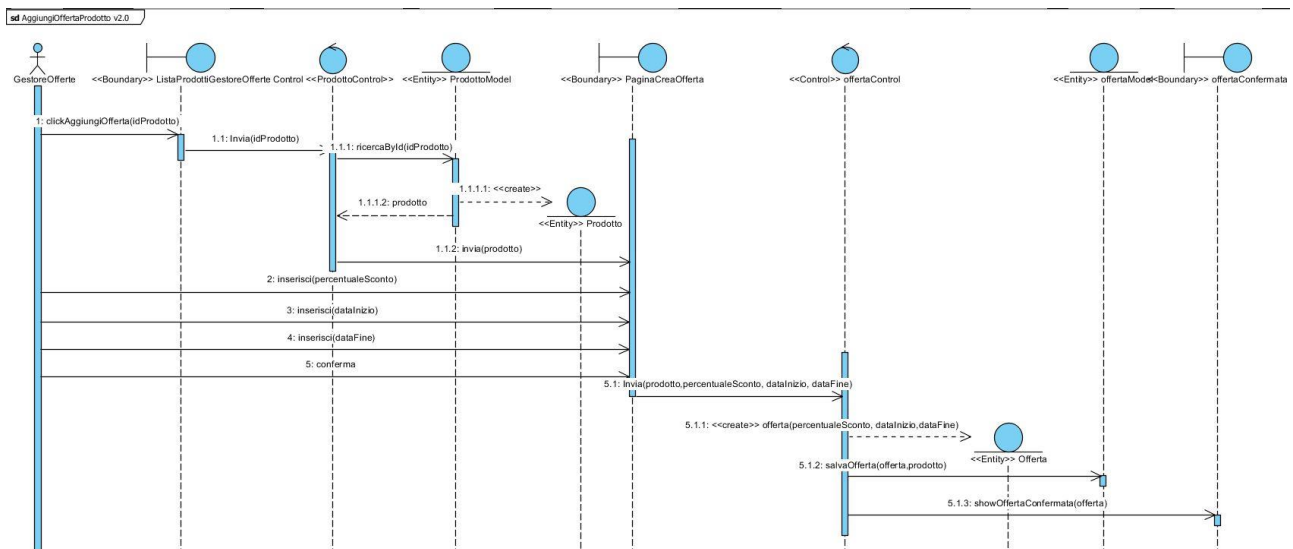
Boundary:

- **AggiungiProdotto**, form usato per l'input dei dati per inserire un nuovo prodotto, Nome, Marca, Tipo, Prezzo, Descrizione, Foto. In fine un bottone che permette la sottomissione del form.
- **prodottoConfermato**, schermata che mostra che l'aggiunta del prodotto al catalogo è avvenuta correttamente.

Control:

- **prodottoControl** crea l'oggetto prodotto, lo passa all'entity prodottoModel e mostra la schermate che il prodotto è stato inserito correttamente.

3.4.4.6. Gestore Offerte– Aggiungi Offerta Prodotto



Attore:

- **GestoreOfferte**, utilizza il sistema per inserire un Offerta.

Boundary:

- **PaginaCreaOfferta**: Form dove il gestore delle offerte riempie i campi dell'offerta.
- **ListaProdottiGestoreOfferte**, lista contenente i prodotti presenti nel sistema, con il bottone aggiungi offerta.
- **OffertaConfermata**, finestra che serve per notificare l'avvenuto inserimento dell'offerta.

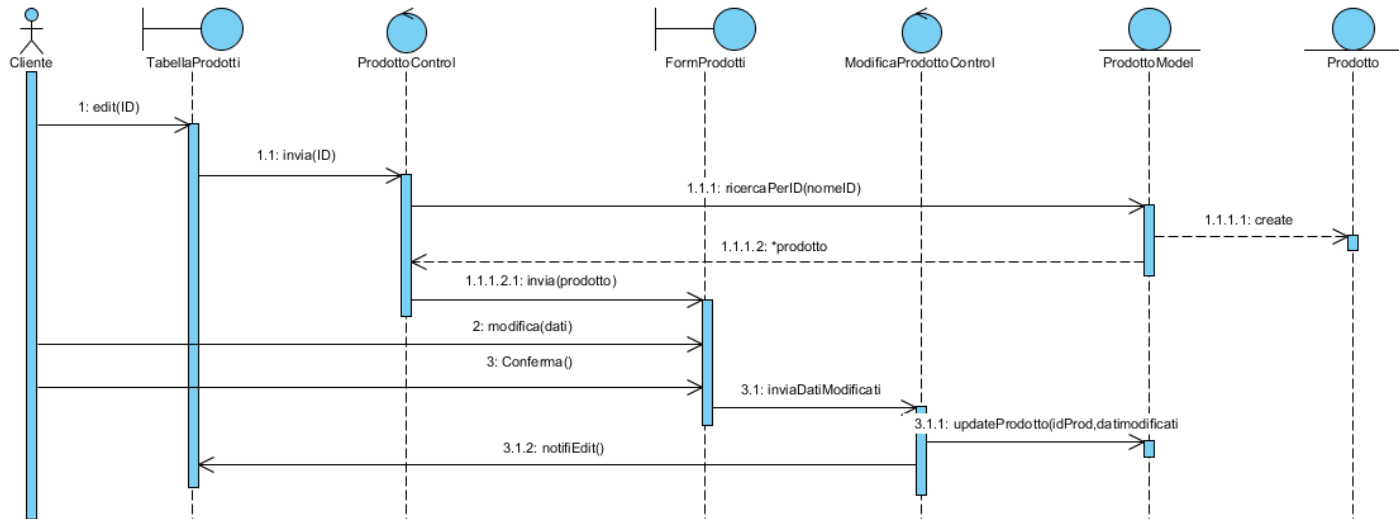
Control:

- **ProdottoControl**: passa l'ID del prodotto da ricercare al database, appena il database gli passa il prodotto, lo invia alla pagina PaginaCreaOfferta.
- **OffertaControl**: riceve il prodotto a cui inserire l'offerta e i dati dell'offerta e crea l'entità offerta, la salva nel database e chiama offertaConfermata.

Entity :

- **Offerta**, informazioni sull'offerta;
- **OffertaModel**: gestisce la comunicazione del database per le offerte.
- **ProdottoModel**: gestisce la comunicazione del database per i prodotti.
- **Prodotto**: crea l'entità prodotto ricavata dal database.

3.4.4.7. Modifica prodotto



Attore:

- **GestoreProdotti** che inizia il caso d'uso

Boundary object:

- **modificaProdotto**, form usata per l'input per modificare le informazioni del prodotto
- **prodottoModificato**, form usata per visualizzare la modifica

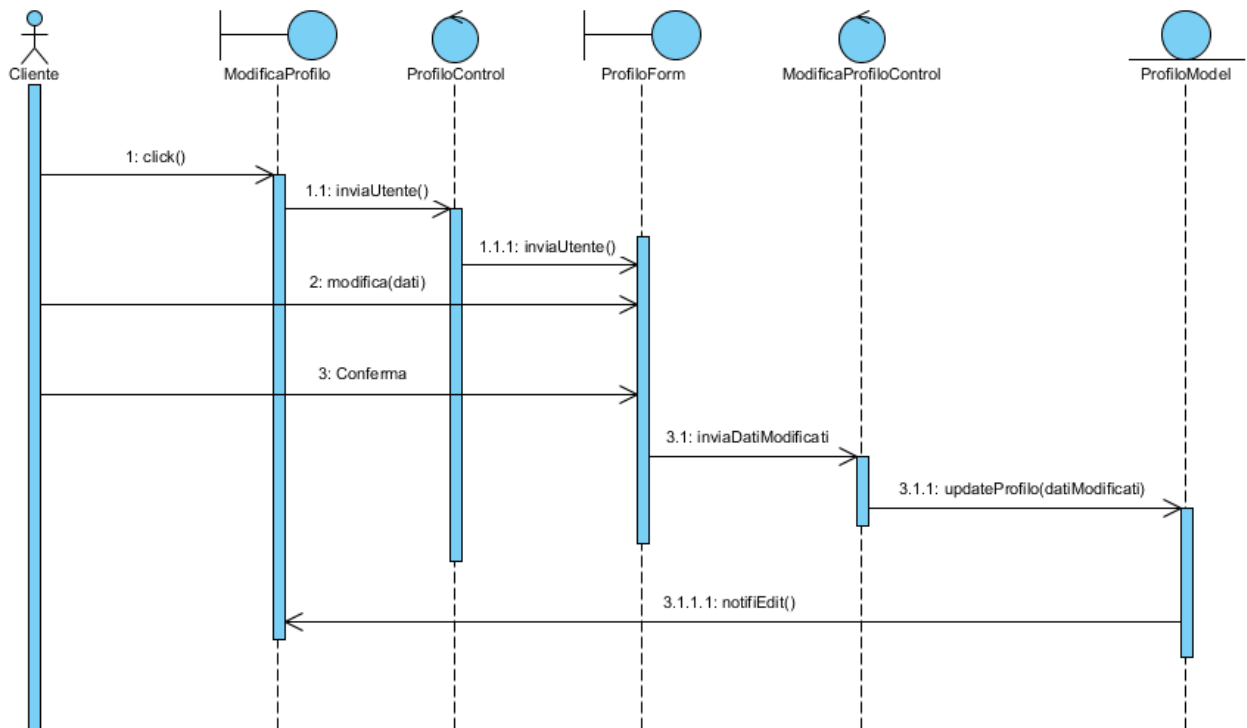
Control object:

- **prodottoControl**, riceve i dati dal <<Boundary>>modificaProdotto e salva le informazioni nel database.

Entity object:

- **prodottoModel** aggiorna le informazioni del prodotto modificato
- **prodotto**, rappresenta l'informazione all'interno del sistema relative al prodotto

3.4.4.8. Modifica dati utente



Attore:

- cliente che inizia il caso d'uso

Boundary object:

- **modificaProfilo**, form usata per l'input per modificare le informazioni del prodotto
- **profiloForm**, form usata per visualizzare la modifica

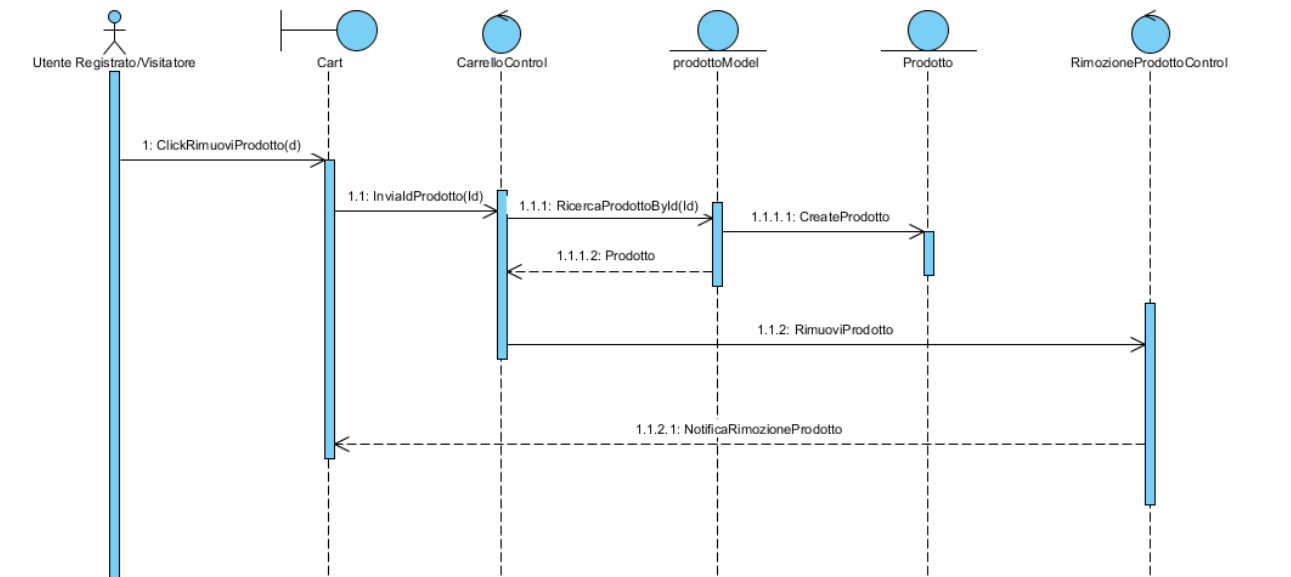
Control object:

- **profiloControl**, riceve i dati dal <<Boundary>>modificaProdotto e salva le informazioni nel database.
- **modificaProfiloControl**

Entity object:

- **profiloModel** aggiorna le informazioni del prodotto modificato

3.4.4.9. Rimuovi prodotto dal carrello



Attore:

- **Cliente/Visitatore** che inizia il caso d'uso

Boundary object:

- **cart**, form usata per visualizzare i prodotti nel carrello

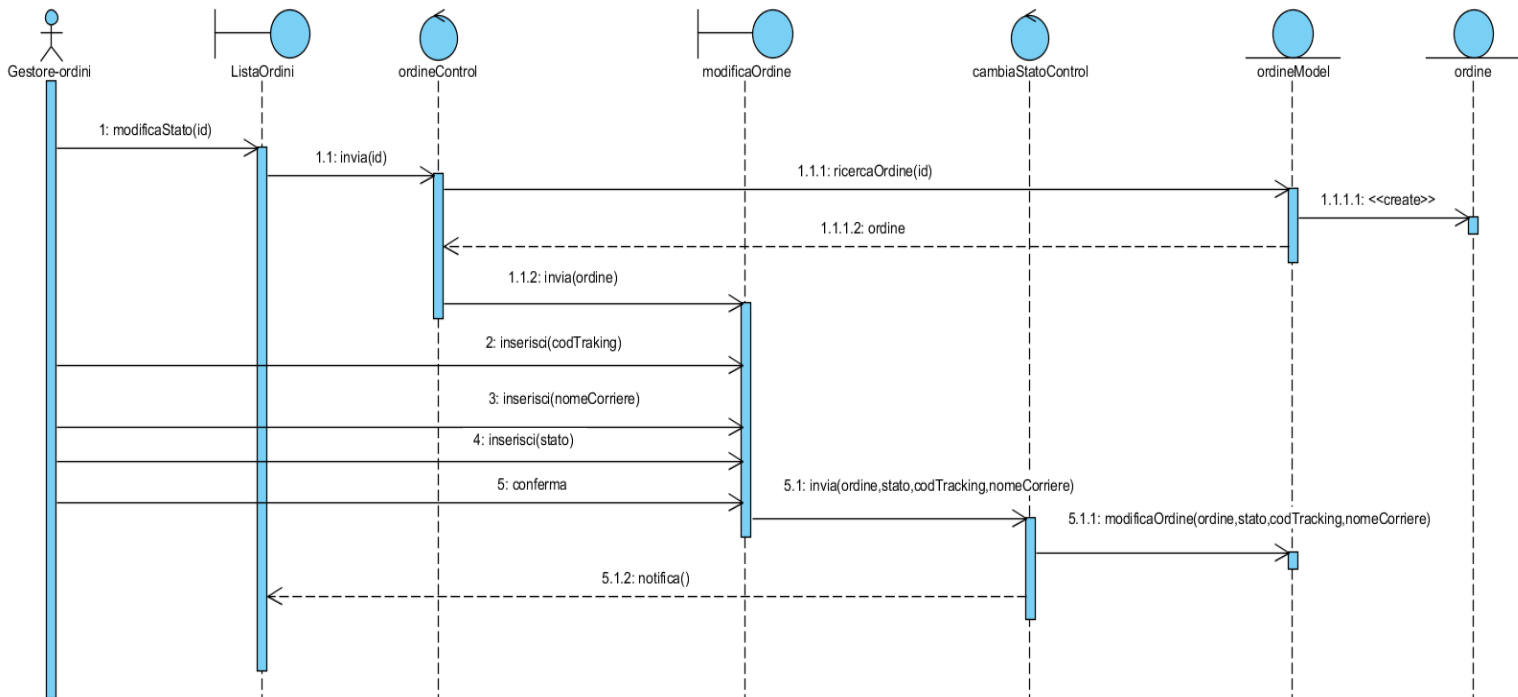
Control object:

- **carrelloControl**, riceve i dati dal <<Boundary>>cart e li invia all'<<entity>> prodottoModel.
- **rimozioneProdottoControl**

Entity object:

- **prodottoModel** ricerca il prodotto nel database tramite l'id ricevuto dal control
- **prodotto**, rappresenta l'informazione all'interno del sistema relative al prodotto

3.4.4.10. Modifica stato ordine



Attore:

- **Gestore-ordini** che inizia il caso d'uso

Boundary object:

- **ListaOrdini**, tabella contenente gli ordini con lo stato "in preparazione"
- **modificaOrdine** form che richiede l'inserimento dello stato, codTracking e nomCorriere

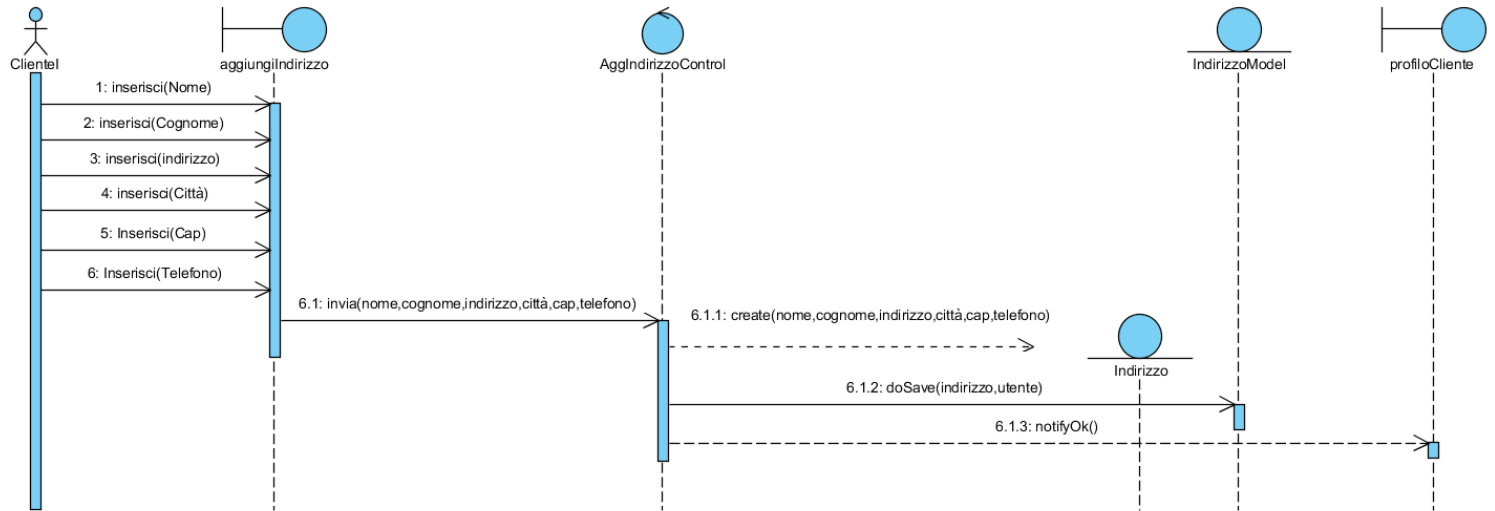
Control object:

- **ordienControl**, riceve i dati dal <<Boundary>> listaOrdini e li invia all'<<entity>> ordineModel per ricercare l'ordine.
- **cambiaStatoControl** si occupa di ricevere i dati dal <<boundary>> modificaOrdine e invia i dati all'<<Entity>> ordineModel per modificare lo stato dell'ordine.

Entity object:

- **ordineModel** ricerca l'ordine nel database tramite l'id ricevuto dal control e si occupa di salvare le modifiche relative all'ordine.
- **ordine**, rappresenta l'informazione all'interno del sistema relative all'ordine

3.4.4.11. Aggiungi indirizzo



Attore:

- **Cliente** che inizia il caso d'uso

Boundary object:

- **aggiungiIndirizzo** form che richiede l'inserimento dei dati per l'inserimento del nuovo indirizzo
- **profiloCliente** permette di visualizzare il profilo del cliente

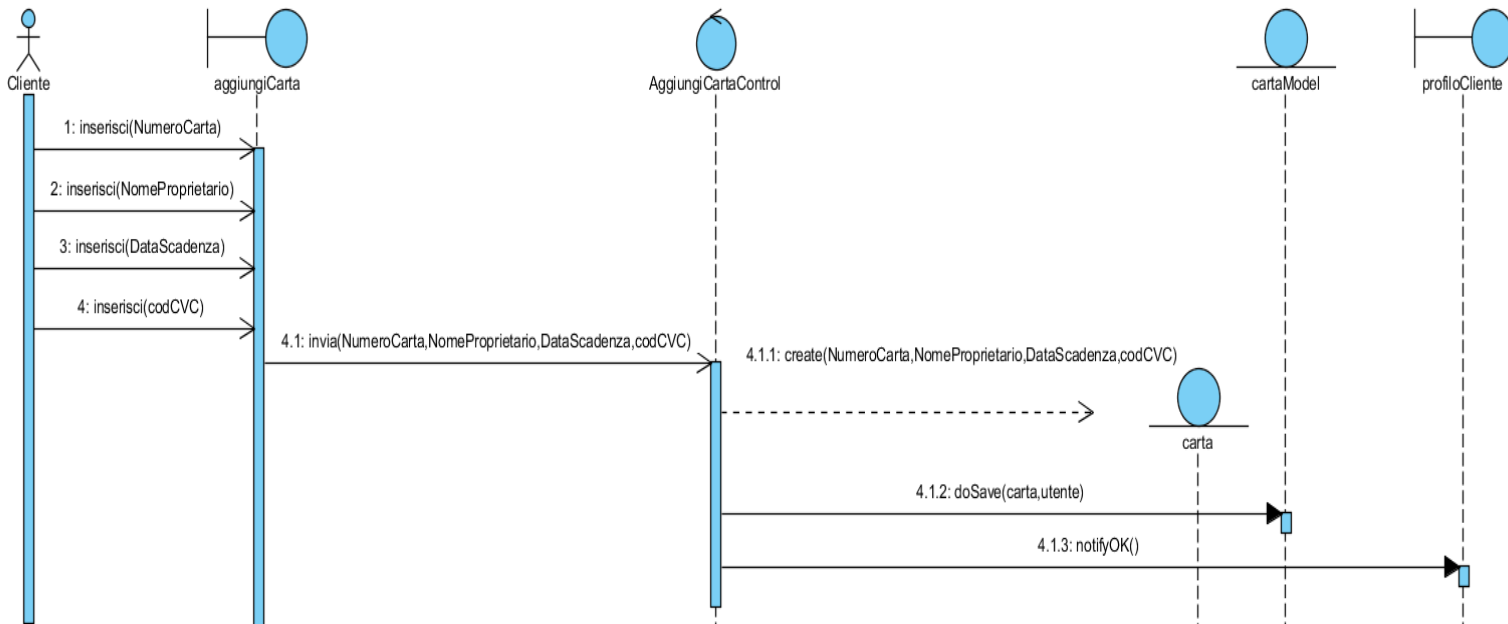
Control object:

- **AggIndirizzoControl**, riceve i dati dal <<Boundary>>aggiungiIndirizzo crea l'oggetto indirizzo e lo invia all'<<entity>> IndirizzoModel

Entity object:

- **indirizzoModel** salva l'indirizzo nel database
- **indirizzo**, rappresenta l'informazione all'interno del sistema relative all'indirizzo

3.4.4.12. Aggiungi carta



Attore:

- **Cliente** che inizia il caso d'uso

Boundary object:

- **aggiungiCarta** form che richiede l'inserimento dei dati per l'inserimento della nuova carta
- **profiloCliente** permette di visualizzare il profilo del cliente

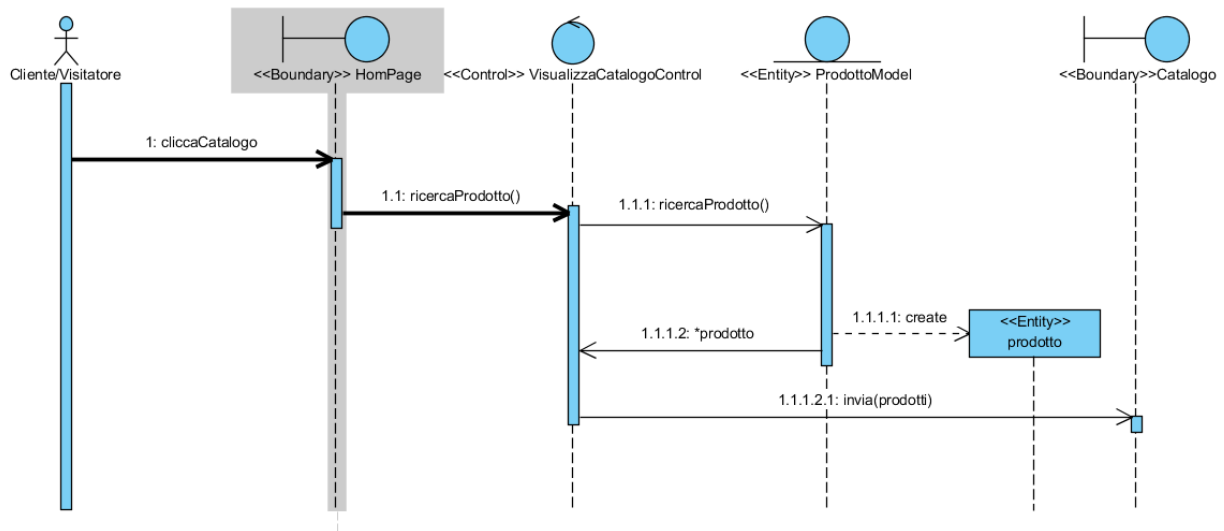
Control object:

- **AggiungiCartaControl**, riceve i dati dal <<Boundary>>aggiungiCarta crea l'oggetto carta e lo invia all'<<entity>> CartaModel

Entity object:

- **cartaModel** salva la carta nel database
- **indirizzo**, rappresenta l'informazione all'interno del sistema relativa alla carta

3.4.4.13. Visualizza catalogo



Attore:

- **Cliente/Visitatore** che inizia il caso d'uso

Boundary

- **HomePage** pagina iniziale del sito web
- **Catalogo** pagina che mostra la lista dei prodotti

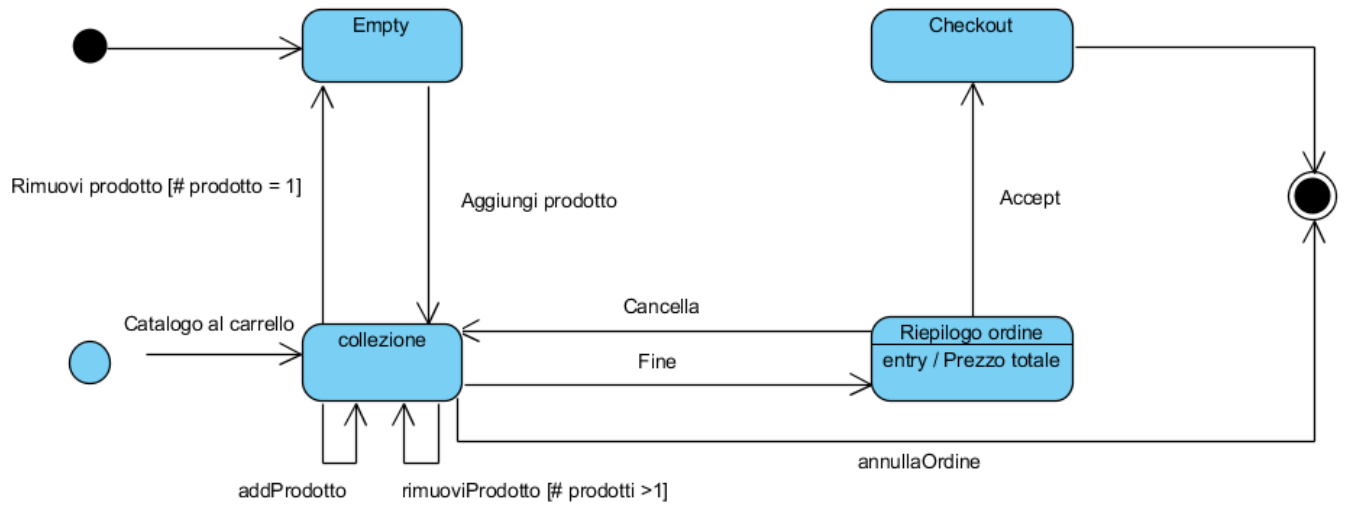
Control

- **VisualizzaCatalogoControl** riceve i dati dal boundary e ricerca i prodotti, infine invia i prodotti restituiti dall'entity al boundary Catalogo

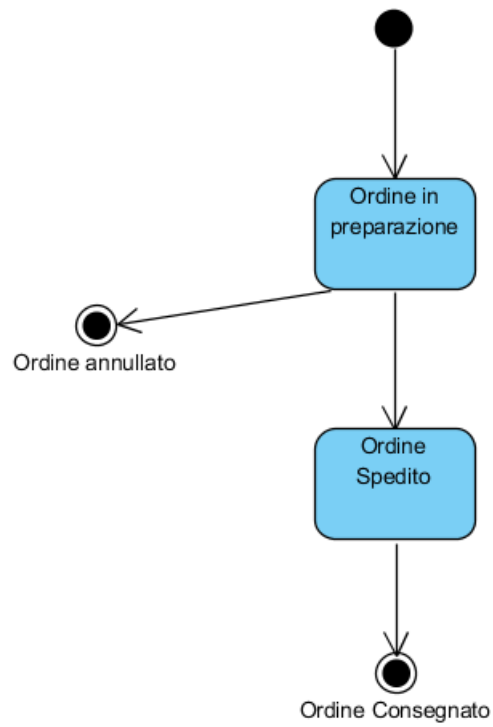
Entity

- **ProdottoModel** funzione che ricerca i prodotti all'interno del database
- **Prodotto** entity creata dal prodotto model

3.4.4.14. Diagramma di stato carrello

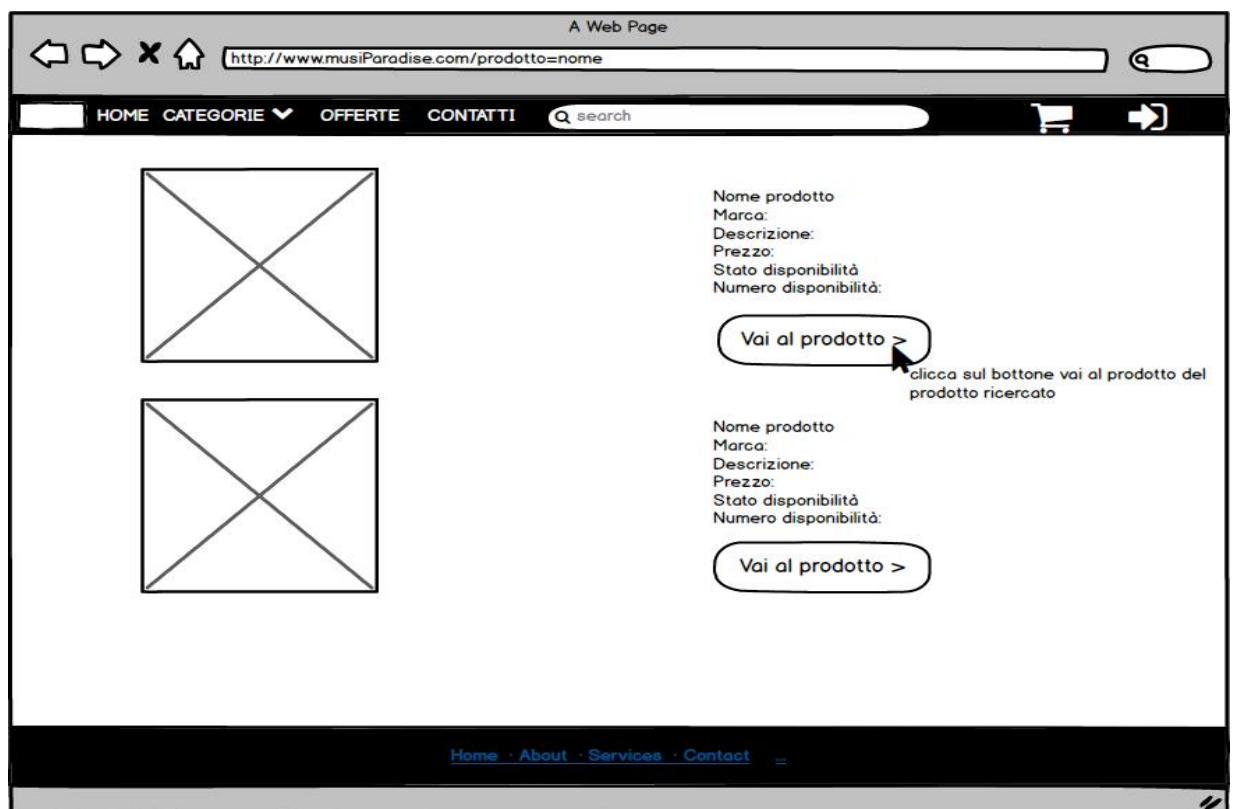
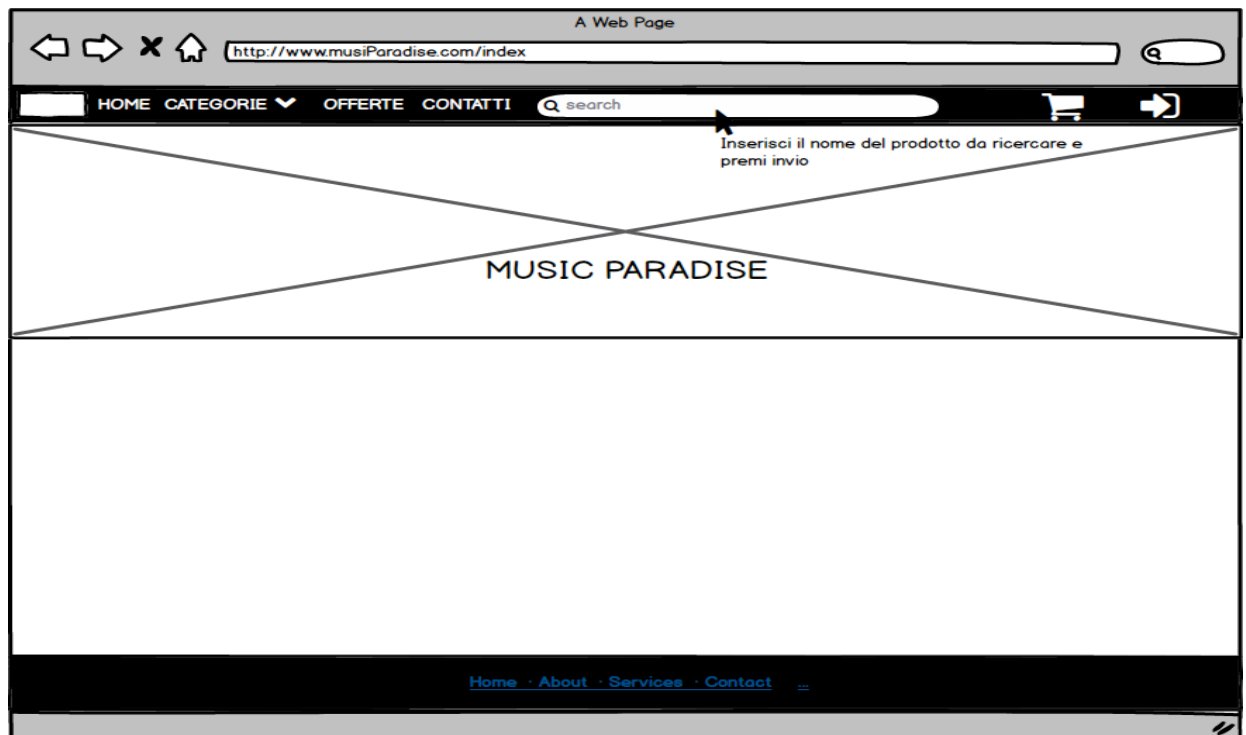


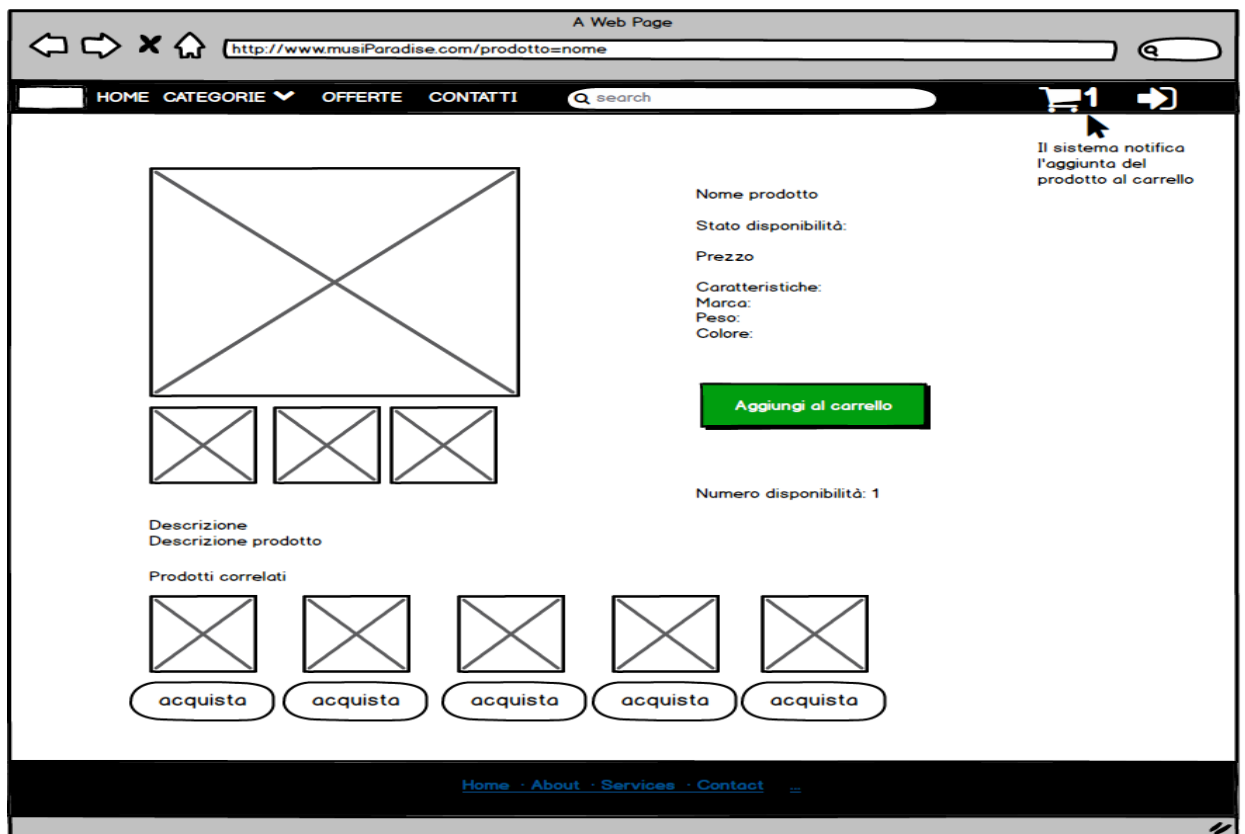
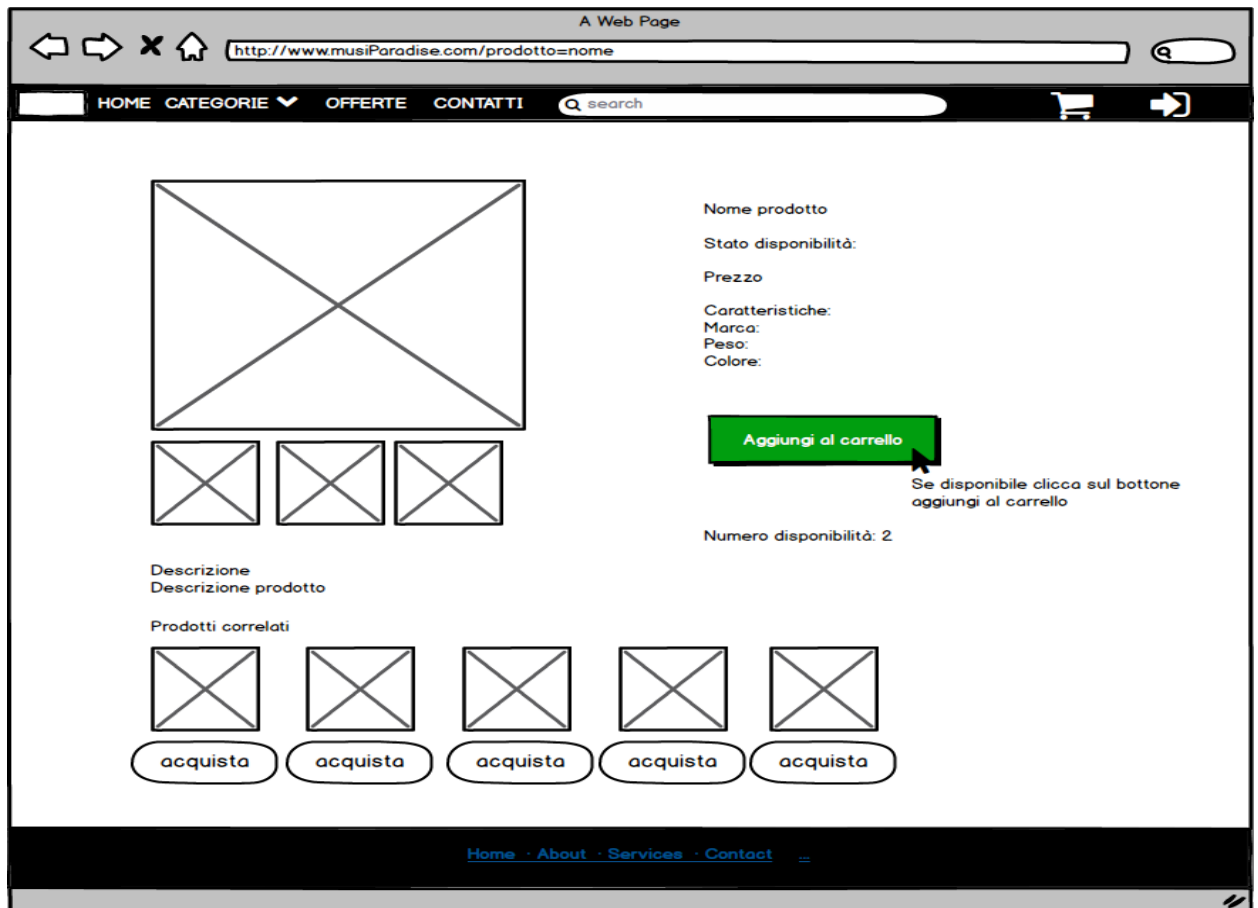
3.4.4.15. Diagramma di stato dell'ordine

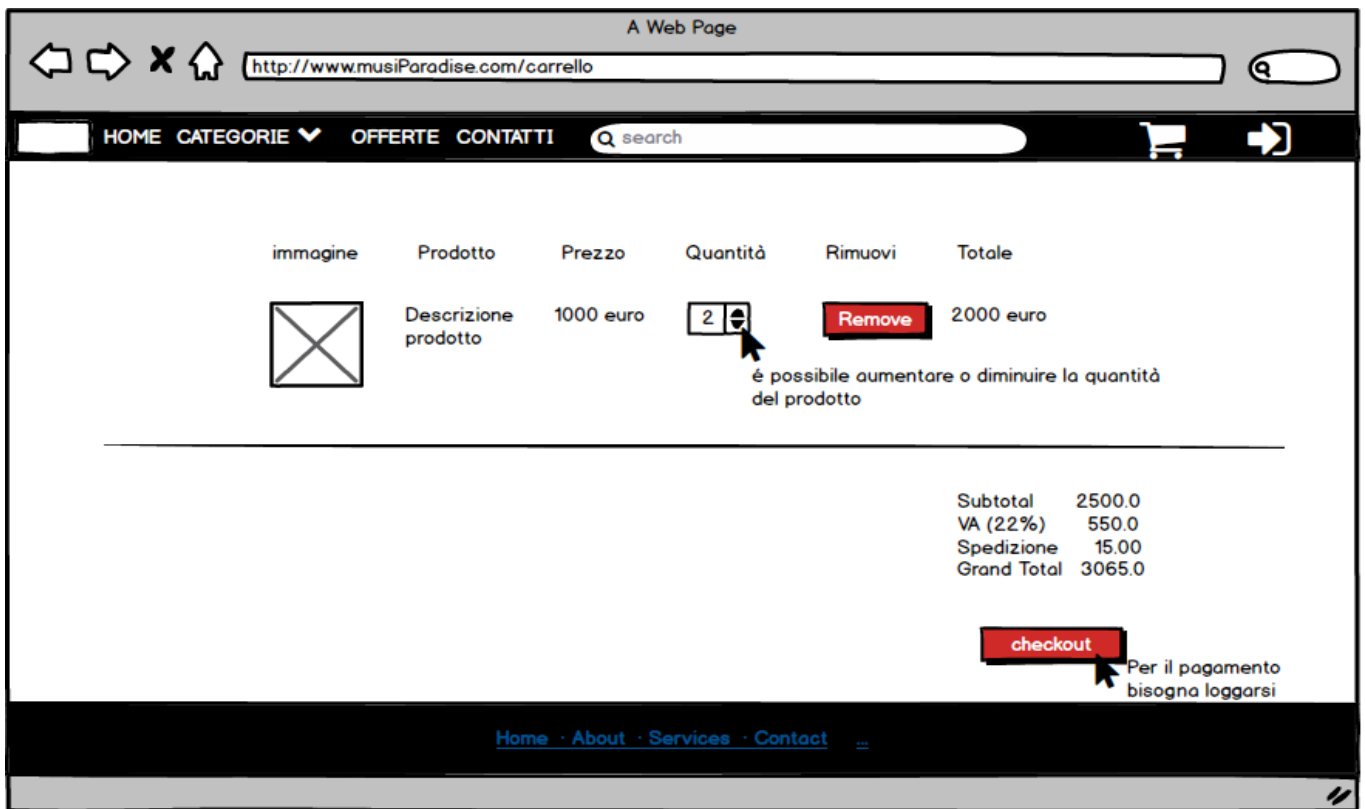


3.4.5.MOCKUP

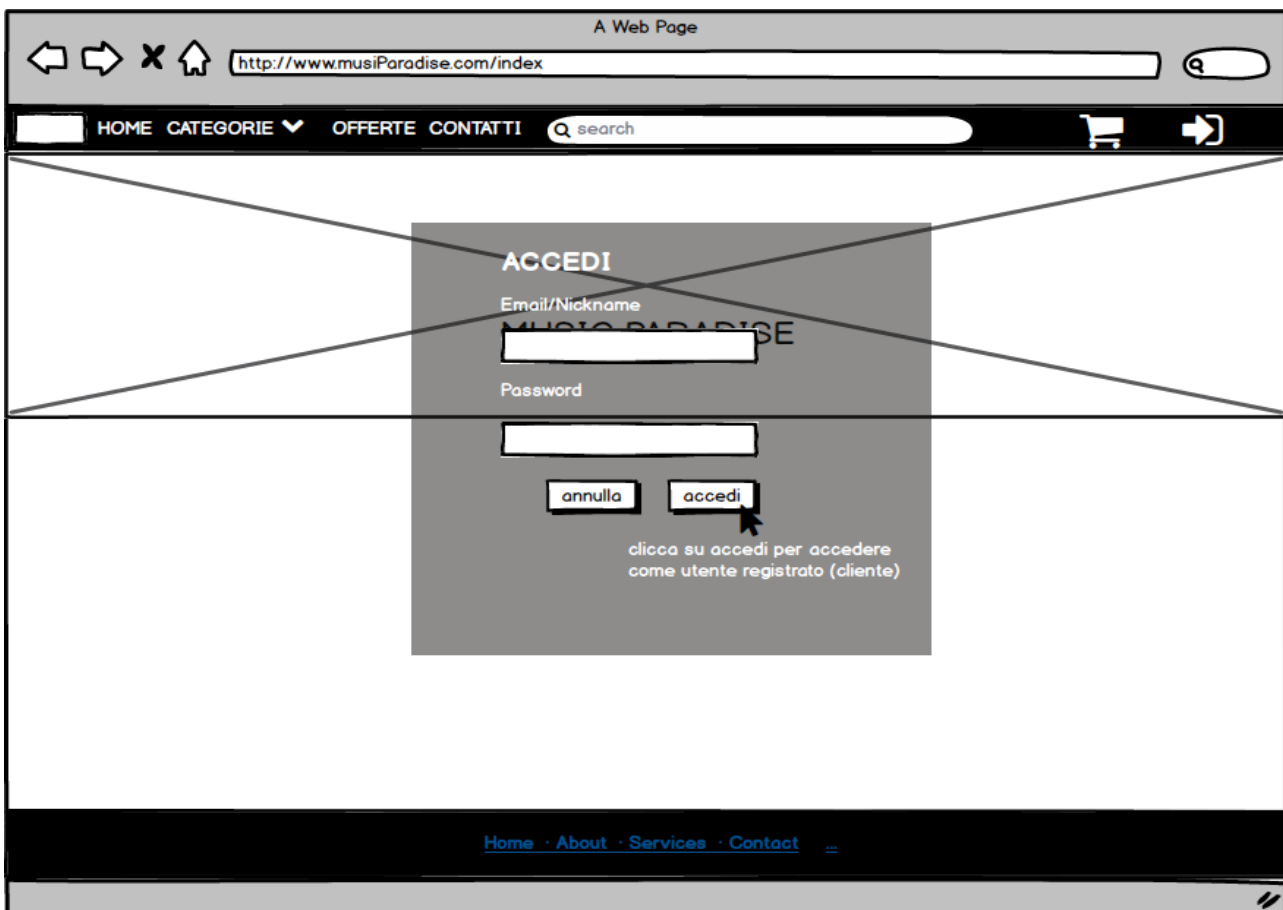
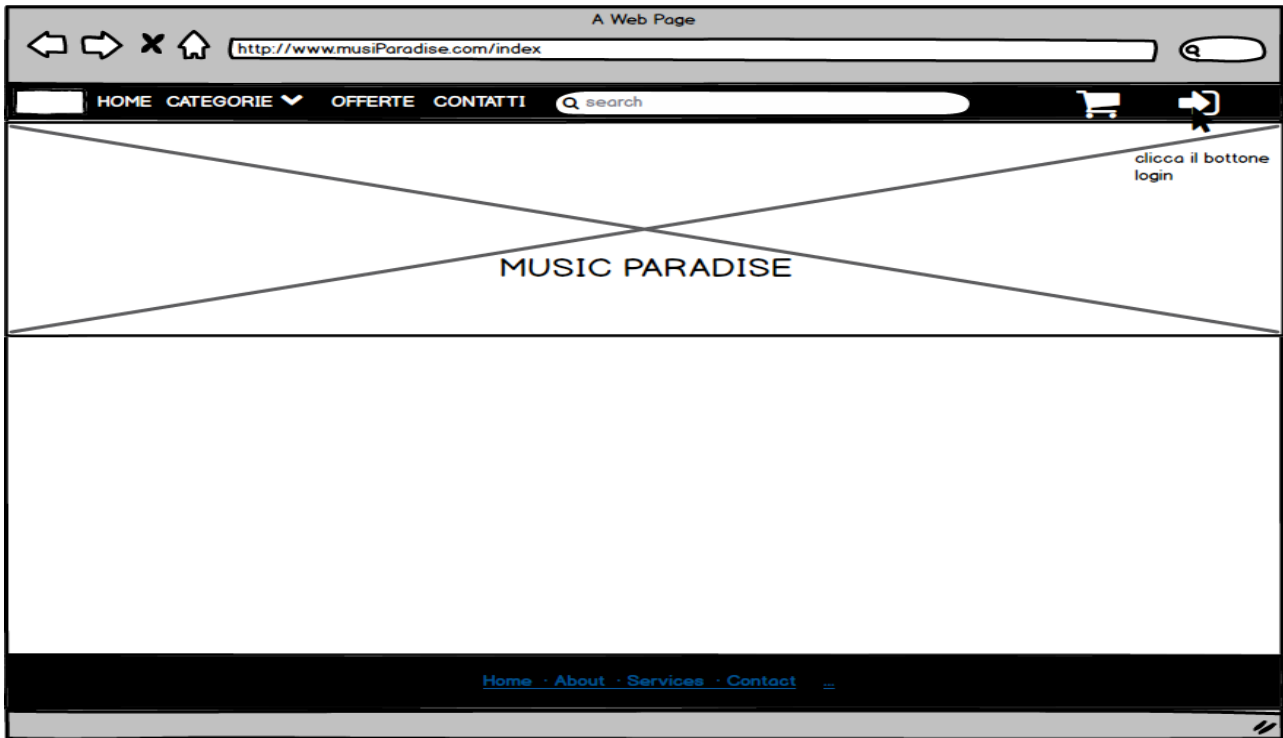
3.4.5.1. Aggiungi prodotto al carrello

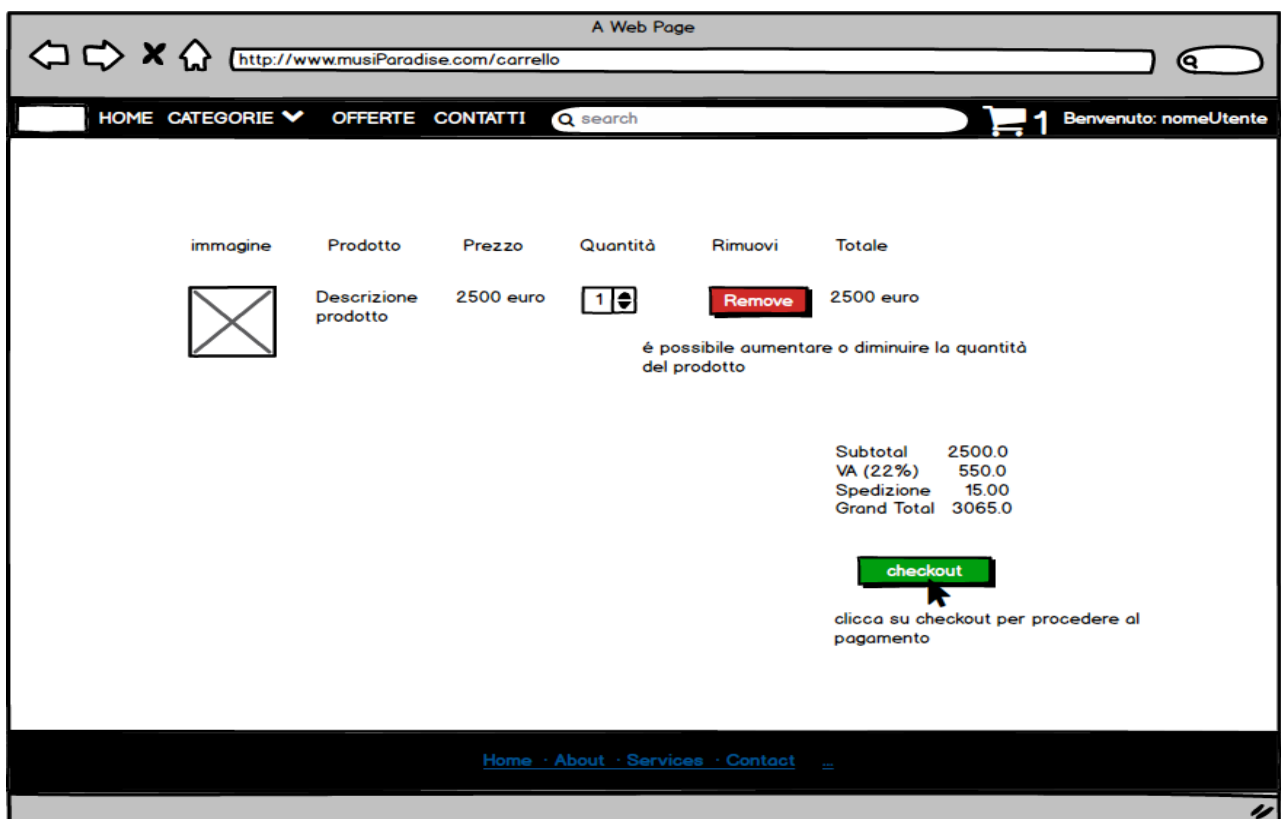
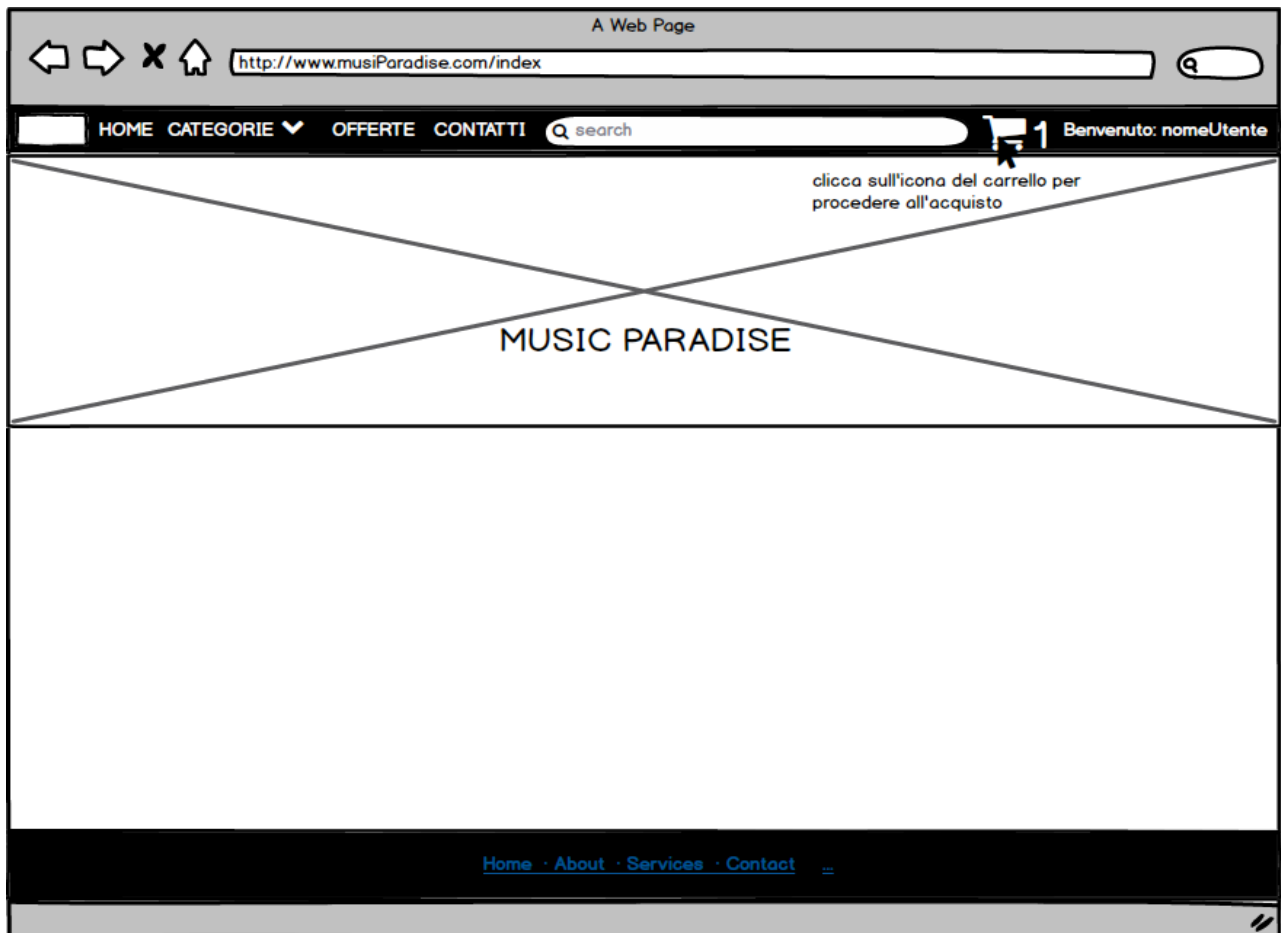






3.4.5.2. ACQUISTA PRODOTTO





A Web Page

http://www.musiParadise.com/pagamento

HOME CATEGORIE OFFERTE search 1 Benvenuto: nomeUtente

Nome
 Cognome
 Città
 CAP
 Nazione
 Indirizzo
 Telefono

N° CARTA
 CVC Scadenza


annulla procedi clicca procedi per effettuare l'ordine

[Home](#) · [About](#) · [Services](#) · [Contact](#) ...

A Web Page

http://www.musiParadise.com/ordine

HOME CATEGORIE OFFERTE CONTATTI search 1 Benvenuto: nomeUtente



Riepilogo ordine

ORDINE N° X

Music Paradise S.p.A
 Informazioni azienda

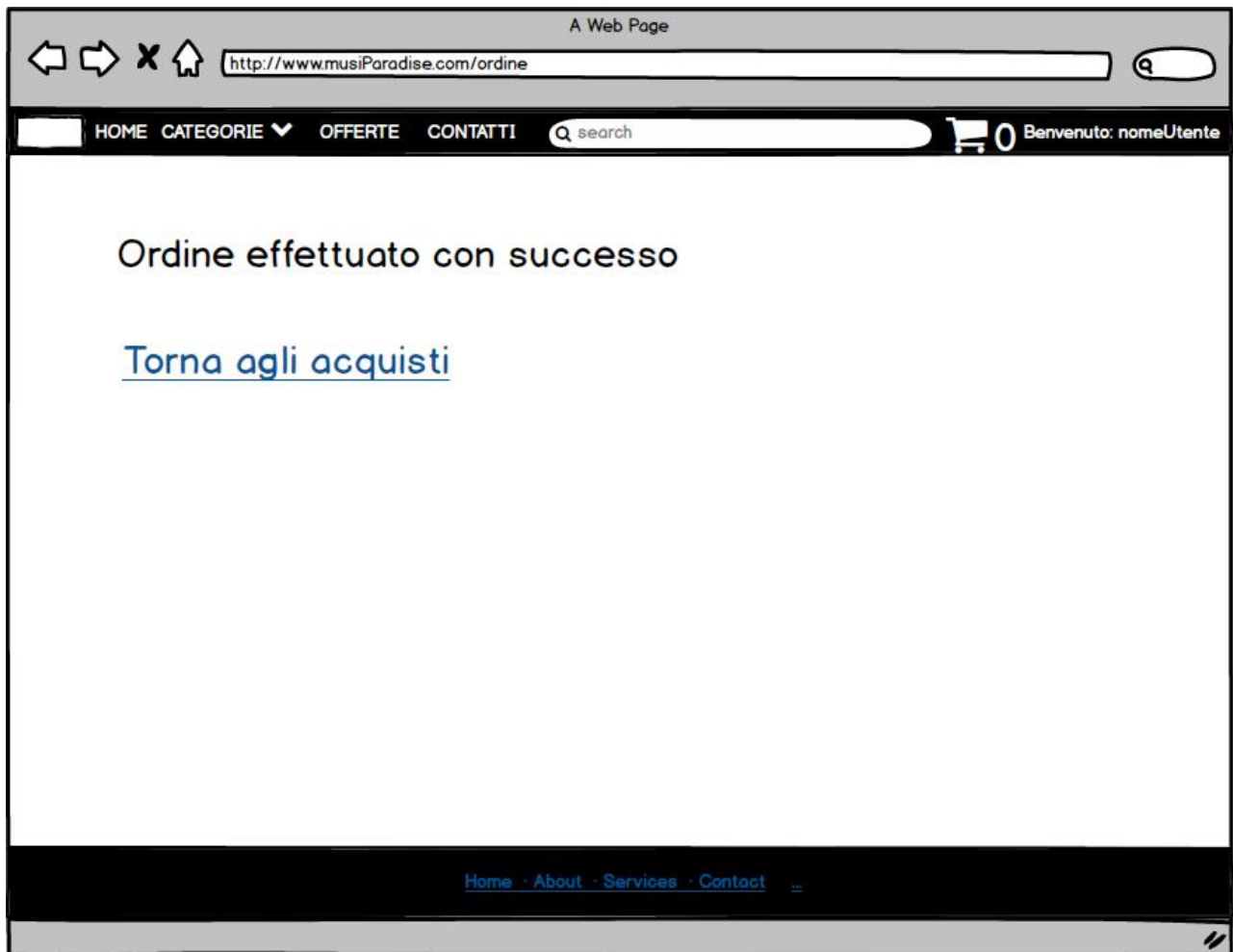
CLIENTE: Cognome Nome
 INDIRIZZO:
 CAP:
 CITTA':
 NAZIONE:

Prodotto	Marca	Prezzo	Quantità	Totale
Subtotal: IVA (22%): Spedizione: Totale ordine:				

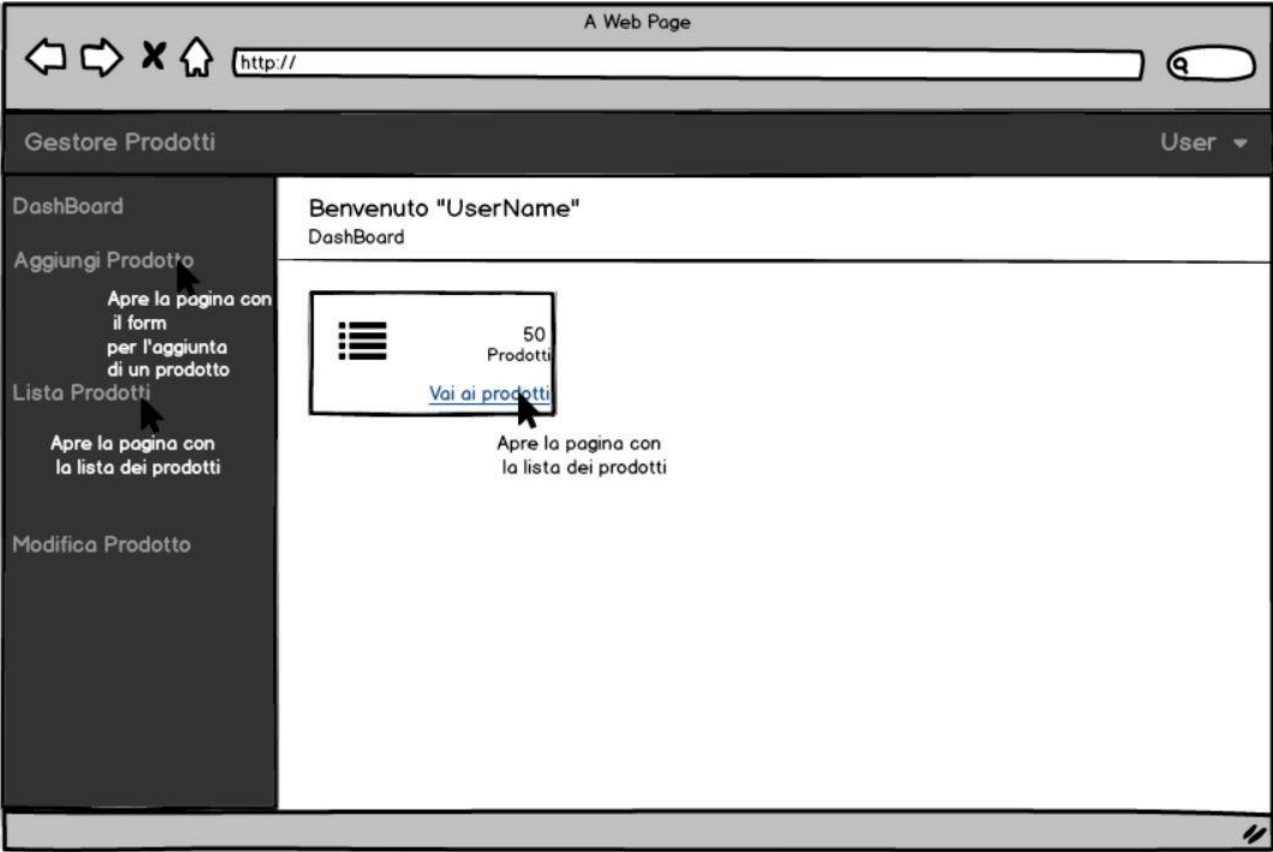
annulla indietro procedi all'acquisto

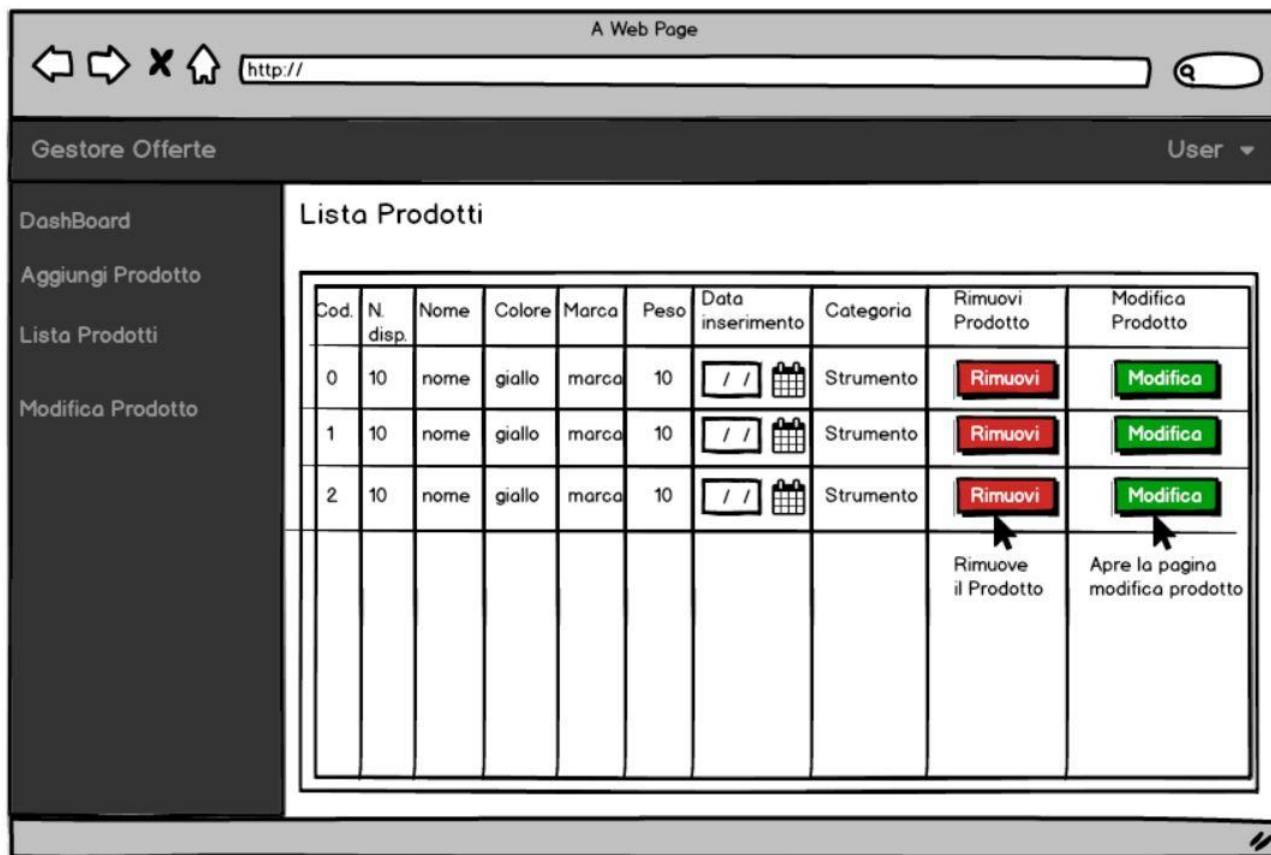
annula l'ordine modifica informazioni ordine Conferma ordine

[Home](#) · [About](#) · [Services](#) · [Contact](#) ...



3.4.5.3. Gestore Prodotti





A Web Page
http://

Gestore Offerte

Dashboard
Aggiungi Prodotto
Lista Prodotti
Modifica Prodotto

Aggiungi Prodotto

Categoria

Categoria del prodotto

Nome

Marca

Numero pezzi disponibili

Numero pezzi disponibili

Descrizione

Peso

Peso del prodotto in KG

Colore

Foto

Carica foto da file

Sezione Offerta (Opzionale)

☐ Aggiungi offerta

☒ Non aggiungere offerta

Sconto (%)

Data Inizio

/ /

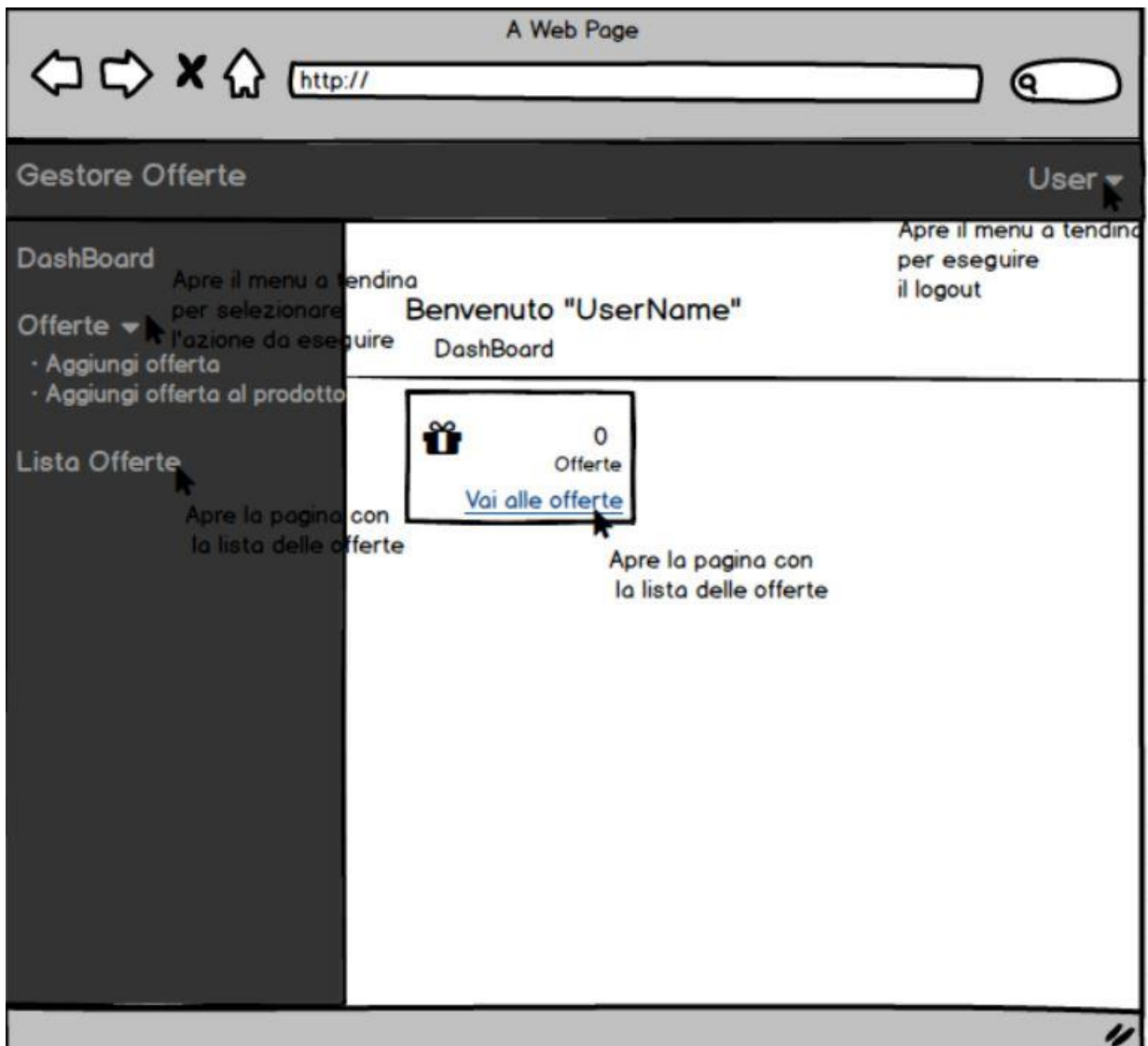
Data Fine

/ /

Aggiungi Prodotto

Aggiunge il prodotto alla lista dei prodotti disponibili

3.4.5.4. Gestore Offerte



A Web Page

http://

Gestore Offerte

User ▾

DashBoard

Offerte ▾

- Aggiungi offerta
- Aggiungi offerta al prodotto

Lista Offerte

Aggiungi offerta

Sconto (%)

Data inizio

/ /

Data fine

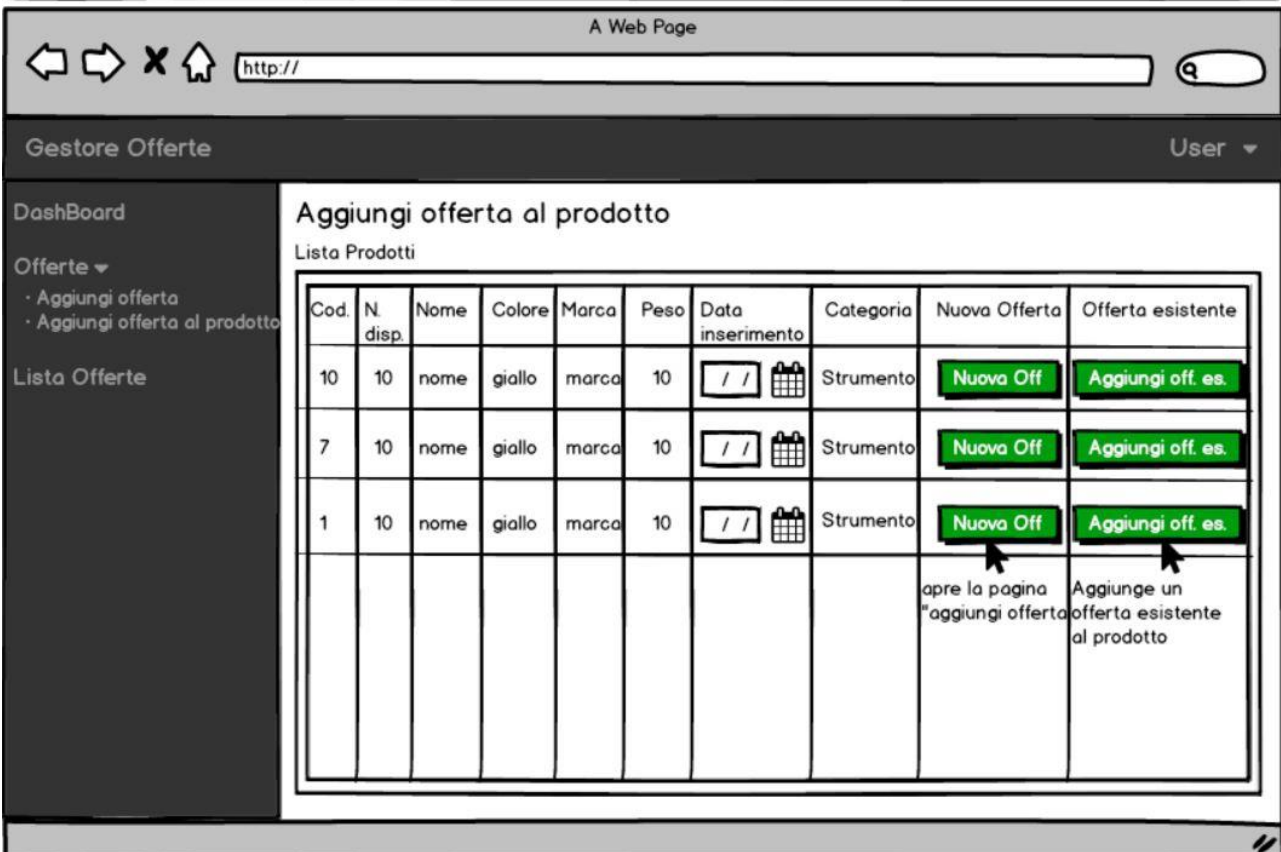
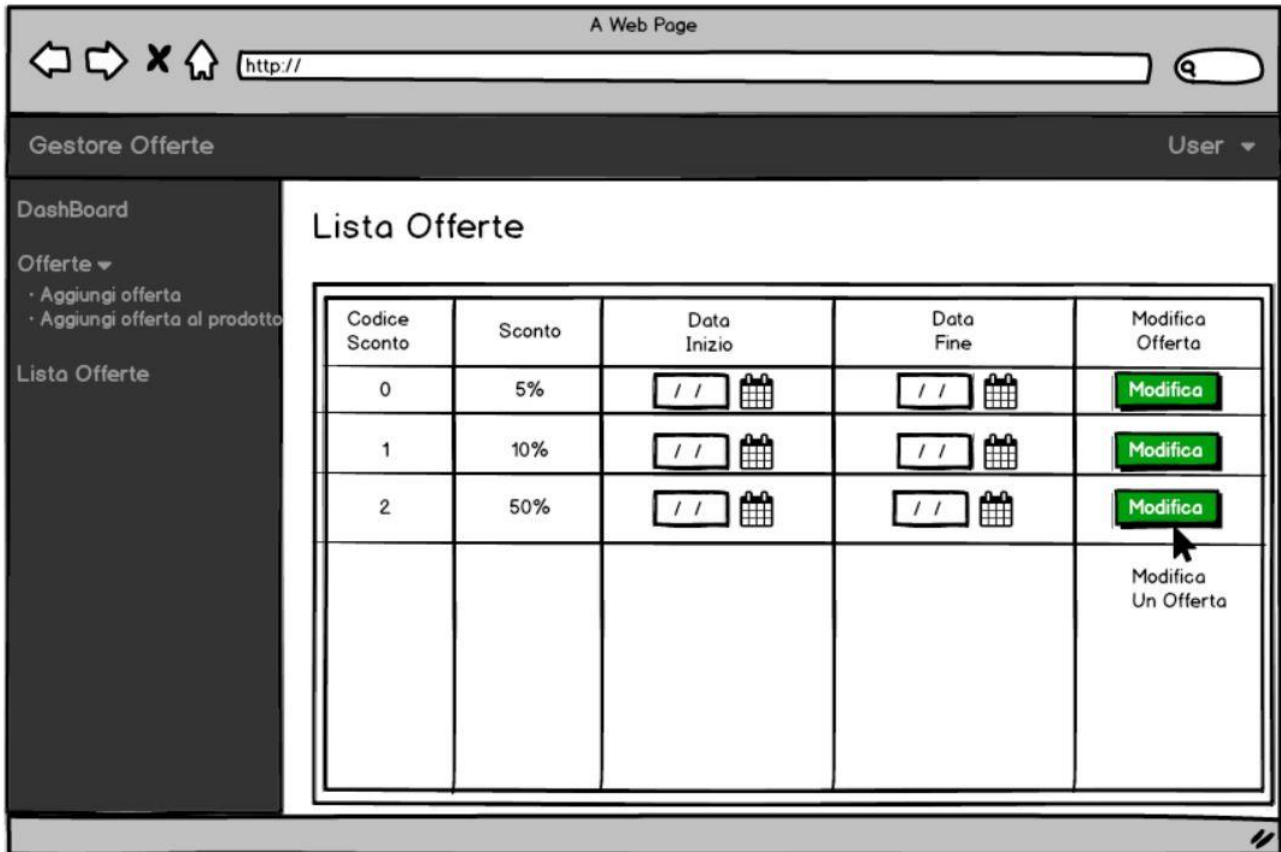
/ /

Submit

Premento il bottone, aggiunge l'offerta

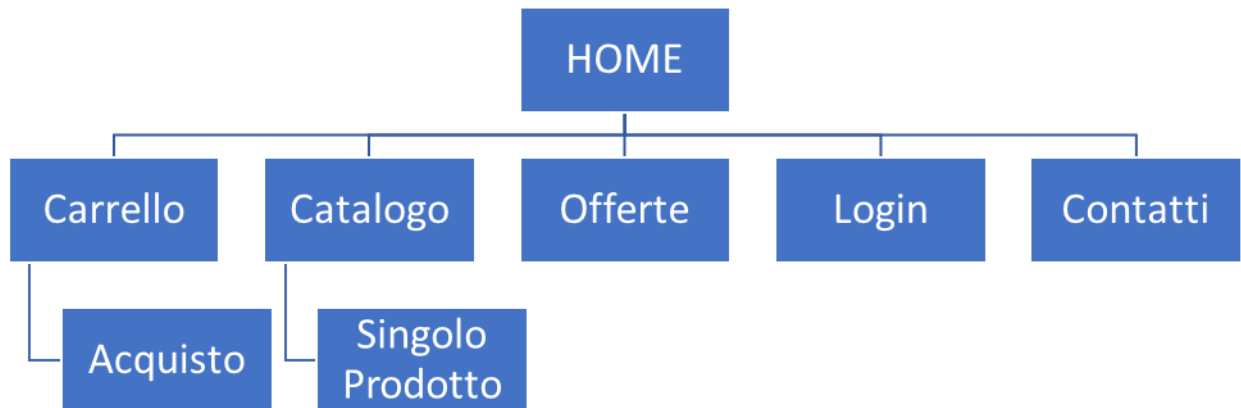
Ingegneria del Software

Pagina 65 di 71

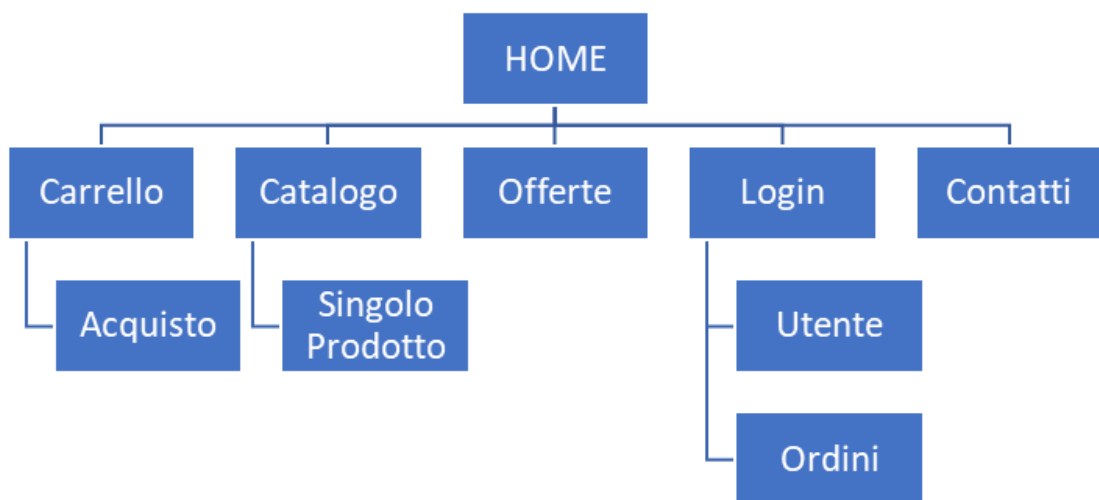


3.4.6. Path navigazionali

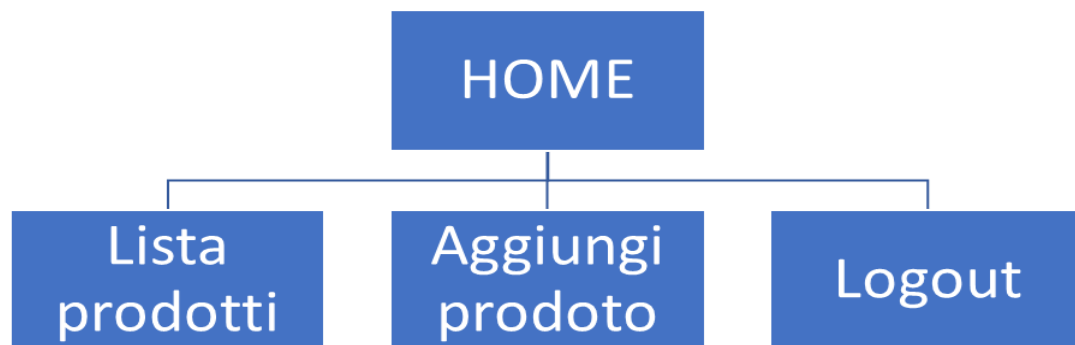
3.4.6.1. Visitatore



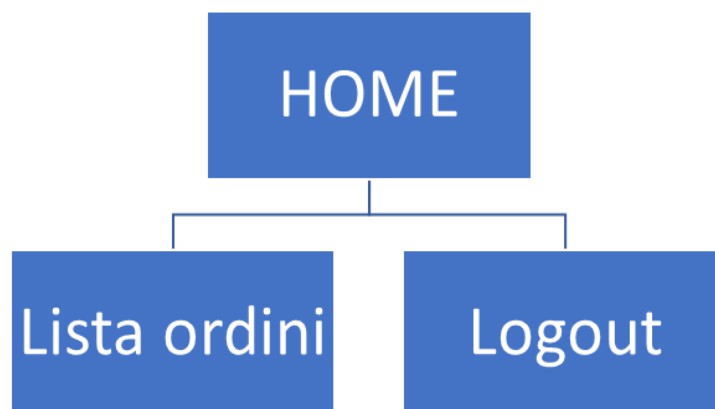
3.4.6.2. Cliente



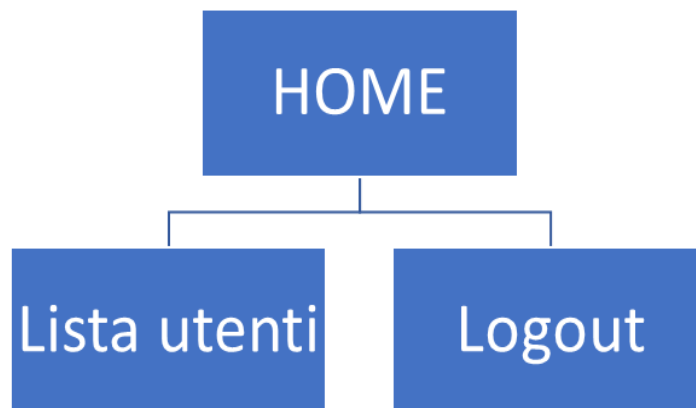
3.4.6.3. Gestore prodotti



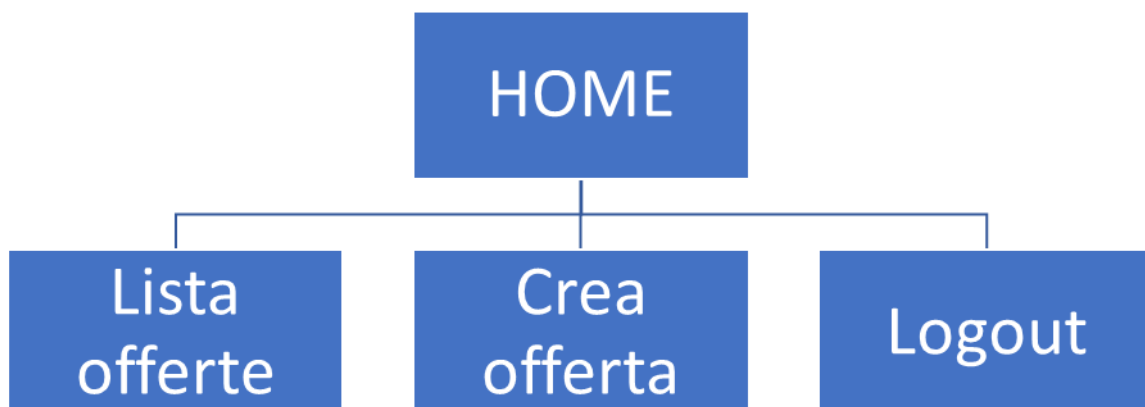
3.4.6.4. Gestore ordini



3.4.6.5. Gestore utenti



3.4.6.6. Gestore offerte



3.5. **GLOSSARIO**

A.

Amministratore

Amministratore di un applicazione desktop che si occupa delle operazioni sugli account di tipo Gestore;

C.

Class diagram

Diagramma delle classi, descrive gli elementi statici di un sistema software;

Cliente

Utente registrato nel sistema;

G.

GestoreOfferte

Gestore che si occupa delle operazioni sulle offerte;

GestoreOrdini

Gestore che si occupa delle operazione sugli ordini;

GestoreProdotti

Gestore che si occupa delle operazioni sui prodotti;

GestoreUtenti

Gestore che si occupa delle operazioni sugli account dei clienti;

R.

RAD

Documento di Analisi dei Requisiti;

S.

Scenario

Istanza di un use case;

Sequence Diagram

Diagramma di sequenza, descrive il comportamento dinamico di un sistema software;

State-Chart Diagram

Diagramma degli stati, descrive il comportamento dinamico di un sistema;

U.

Use Case Diagram

Diagramma dei casi d'uso, descrive il comportamento funzionale del sistema software;

V.

Visitatore

Utente non registrato al sito web;