**REQUISITI E CASI D’USO**

**RAD\_EmmeShop**

Sommario

[**1)** **Introduction** 4](#_Toc31565831)

[**1.1)** **Proposta di sistema** 4](#_Toc31565832)

[**1.2)** **Scopo del sistema** 4](#_Toc31565833)

[**1.3)** **Obiettivi e criteri di successo** 4](#_Toc31565834)

[**1.4)** **Riferimenti** 5](#_Toc31565835)

[**2)** **Current system** 5](#_Toc31565836)

[**3)** **Proposed System** 5](#_Toc31565837)

[**3.1) Overview** 5](#_Toc31565838)

[**3.2) Functional Requirements** 5](#_Toc31565839)

[**3.2.4) Cercare prodotti di un determinato negozio** 6](#_Toc31565840)

[**4)** **Nonfunctional requirements** 8](#_Toc31565841)

[**4.1) Usability:** 8](#_Toc31565842)

[**4.2) Reliability** 9](#_Toc31565843)

[**4.3) Performance** 9](#_Toc31565844)

[**4.4) Supportability** 9](#_Toc31565845)

[**4.5) Implementation** 9](#_Toc31565846)

[**4.6) Interface** 9](#_Toc31565847)

[**4.7) Packaging** 9](#_Toc31565848)

[**4.8) Legal** 9](#_Toc31565849)

[**5)** **System models** 10](#_Toc31565850)

[**5.1) Scenari** 10](#_Toc31565851)

[**5.1.1) Creazione del negozio** 10](#_Toc31565852)

[**5.1.2) Acquistare da un negozio** 10](#_Toc31565853)

[**5.2) Use Case Model** 10](#_Toc31565854)

[**5.2.1) Attori:** 10](#_Toc31565855)

[ **Use Case Diagram** 12](#_Toc31565856)

[5.2.2) Registrazione venditore 15](#_Toc31565857)

[***5.2.2.1 checkRegistrazione*** 15](#_Toc31565858)

[5.2.3) Login Venditore 17](#_Toc31565859)

[***5.2.3.1*** ***)errorLogin*** 17](#_Toc31565860)

[18](#_Toc31565861)

[5.2.4) Creazione del negozio 19](#_Toc31565862)

[***5.2.4.1 checkCampi*** 19](#_Toc31565863)

[5.2.5) Inserisci categoria 21](#_Toc31565864)

[***5.2.5.1 checkCampi*** 21](#_Toc31565865)

[5.2.6) Inserisci prodotto 23](#_Toc31565866)

[***5.2.6.1 checkCampi*** 23](#_Toc31565867)

[5.2.7) Modifiche categorie 25](#_Toc31565868)

[***5.2.7.1 checkCampi*** 25](#_Toc31565869)

[26](#_Toc31565870)

[5.2.8) Modifiche prodotto 27](#_Toc31565871)

[***5.2.8.1 notUploadImage*** 27](#_Toc31565872)

[5.2.9) Cancellazione prodotto 29](#_Toc31565873)

[5.2.10) Gestione degli ordini 30](#_Toc31565874)

[30](#_Toc31565875)

[5.2.11) Registrazione Cliente 31](#_Toc31565876)

[**5.2.11.1 checkCampi** 31](#_Toc31565877)

[5.2.12) Login cliente 33](#_Toc31565878)

[**5.2.12.1 errorLogin** 33](#_Toc31565879)

[5.2.13) Visualizzazione negozio 35](#_Toc31565880)

[5.2.14) Visualizza carello 36](#_Toc31565881)

[5.2.15) Aggiunta prodotto al carello 37](#_Toc31565882)

[***5.2.15.1 pathAlternativo*** 37](#_Toc31565883)

[**5.2.15.2 quantitàNonDisponibile** 37](#_Toc31565884)

[5.2.16) Rimozione prodotto (dal Carrello) 39](#_Toc31565885)

[5.2.18) Visualizzazione categorie 40](#_Toc31565886)

[5.2.19) Visualizzazione prodotti 41](#_Toc31565887)

[5.2.20) Visualizzazione ordini 42](#_Toc31565888)

[5.2.21) Conferma un ordine 43](#_Toc31565889)

[**5.2.17.1 checkOrdine** 43](#_Toc31565890)

[***5.2.17.2 nessunProdottonelCarrello*** 43](#_Toc31565891)

[44](#_Toc31565892)

[5.2.22) Login Amministratore 45](#_Toc31565893)

[**5.2.18.1 errorLogin** 45](#_Toc31565894)

[5.2.23) Visualizza clienti 47](#_Toc31565895)

[5.2.24) Visualizza venditori 48](#_Toc31565896)

[5.2.25) Modifica cliente 49](#_Toc31565897)

[**5.2.21.1 checkCampi** 49](#_Toc31565898)

[5.2.26) Modifica venditore 51](#_Toc31565899)

[**5.2.22.1 checkCampi** 51](#_Toc31565900)

[5.2.27) Cancellazione cliente 53](#_Toc31565901)

[**5.2.23.1 annullaEliminazione** 53](#_Toc31565902)

[5.2.28) Cancellazione venditore 55](#_Toc31565903)

[**5.2.24.1 annullaEliminazione** 55](#_Toc31565904)

[**5.3) Formato dati:** 57](#_Toc31565905)

[**5.4.1 Object Model** 58](#_Toc31565906)

[**5.4.1.1 Categoria** 59](#_Toc31565907)

[**5.4.1.2 Prodotto** 59](#_Toc31565908)

[**5.4.1.3 Carrello** 60](#_Toc31565909)

[**5.4.1.4 Negozio** 60](#_Toc31565910)

[**5.4.1.5 Venditore** 61](#_Toc31565911)

[**5.4.1.6 Utente** 61](#_Toc31565912)

[**5.4.1.7 Gestore Accounting** 61](#_Toc31565913)

[**5.4.1.8 Cliente** 62](#_Toc31565914)

[**5.4.1.9 Fattura** 62](#_Toc31565915)

[**5.4.1.10 Riferimento** 62](#_Toc31565916)

[**5.4.1.11 Gestore spedizioni** 63](#_Toc31565917)

[**5.5 Dynamic model** 63](#_Toc31565918)

[**5.6 User interface, navigatioinal paths and screen mock-ups** 66](#_Toc31565919)

[**5.6.1 Navigational paths** 66](#_Toc31565920)

[**5.6.2 Content map** 66](#_Toc31565921)

[**5.6.3 User Interface and Screen mock-ups** 67](#_Toc31565922)

[67](#_Toc31565923)

[**6)Glossario** 72](#_Toc31565924)

# **Introduction**

## **Proposta di sistema**

Fino a qualche anno fa la mancanza di contatto con il prodotto, l’impossibilità di confrontarsi con un addetto alle vendite, i lunghi tempi di attesa e i costi di spedizione scoraggiavano gli utenti nel compiere acquisti online.

Oggi la situazione è completamente diversa:

trend degli acquisti online sono in costante crescita.

Sempre più persone scelgono di concludere le proprie trattative online, complice anche una maggiore fiducia nel trattamento dei dati personali e nei metodi di pagamento.

## **Scopo del sistema**

Lo scopo del progetto Emme Shop è quello di sviluppare una piattaforma software per la creazione e gestione del proprio sito e-commerce, contemporaneamente permette all’acquirente di trovare in un unico luogo varie proposte di negozi. La piattaforma dovrebbe essere semplice e intuitivo, per utenti inesperti grazie alla guida del sistema.

Per rendere Emme shop raggiungibile dal maggior numero di utenti, deve essere disponibile su un server online, visualizzabile su tutti i browser, sia personal computer che smartphone o smart tv….

## **Obiettivi e criteri di successo**

Gli obiettivi del progetto di Emme Shop sono:

* Fornire una piattaforma semplice e intuitiva, che sia disponibile su tutti i dispositivi.
* Gestore Accounting del negozio, dovrebbe essere in grado di svolgere il lavoro di gestione del proprio sito senza ulteriori conoscenze specifiche (es. aggiunta di un prodotto, modifica delle quantità o prezzi di un prodotto).
* Il cliente dovrebbe essere in grado di acquistare prodotti da diversi negozi nello stesso ordine.

Il progetto è accettato se almeno i seguenti criteri di successo sono soddisfatti:

* La piattaforma è disponibile il 10 gennaio 2020 e soddisfa i requisiti funzionali specificati nella sezione 3.2.
* Il server che ospita Emme Shop è visibile su qualsiasi dispositivo connesso a Internet, ad uno specifico indirizzo IP.
* Il sito deve essere online 24h su 24, eccetto nei giorni prefissati di manutenzione.
* I requisiti non funzionali di navigabilità, tempo di risposta breve, completezza dei contenuti, multilingua, interfaccia intuitiva della sezione 3.3.

## **Riferimenti**

[PS\_EmmeShop.docx](../aDocumentiDefinitivi%20accesso%20solo%20cerangolo/PS_EmmeShop.docx)

# **Current system**

Il progetto Emme Shop è un progetto di ingegneria greenfield. Non esiste un sistema attuale da sostituire e task di Emme Shop non sono state ancore completate.

# **Proposed System**

**3.1) Overview**

Il focus di questo progetto è sviluppare una piattaforma di e-commerce chiamata Emme Shop. Esistono tre tipi di utenti, **cliente**, **venditore** e **gestore Accounting**.

Emme Shop è un sito Web che mette a disposizione vari negozi e permette ai clienti di acquistare prodotti di loro interesse. L’home page presenta i vari negozi disponibili sulla piattaforma e altri campi dove sia il cliente che il venditore possono iscriversi e accedere alla piattaforma.

Il **venditore** effettua l’accesso, dopodiché, può inserire i propri prodotti e il **cliente** può visualizzarli, aggiungerli al carrello ed ‘infine acquistarli.

Il **gestore accounting** controlla i negozi e i clienti iscritti alla piattaforma, può modificare e cancellare alcune informazioni inserite nel momento della registrazione.

## **3.2) Functional Requirements**

Il sito Emme Shop supporta tre tipi di utenti: **cliente**, **venditore** e **gestore accounting**. Ognuno di questi utenti ha dei requisiti funzionali ben precisi.

**Il cliente supporta i seguenti requisiti funzionali:**

**3.2.1) Creare un account**

Il cliente deve registrarsi alla piattaforma inserendo informazioni che saranno utili nel momento in cui dovrà effettuare l’accesso per poter acquistare i vari prodotti.

**3.2.2) Visualizzare i negozi**

Il cliente una volta effettuato l’accesso, tramite la piattaforma, potrà visualizzare i negozi disponibili su di essa.

**3.2.3) Visualizzare categorie dei negozi**

Il cliente oltre a visualizzare i negozi, visionerà le varie categorie inserite dai negozi.

### **3.2.4) Cercare prodotti di un determinato negozio**

il cliente dopo aver effettuato l’accesso è in grado di ricercare un prodotto in base alle sue esigenze, per mezzo delle keyword.

**3.2.5) Selezionare la quantità**

Il cliente visualizzando i prodotti disponibili di un determinato negozio, può decidere la quantità desiderata, anche al momento della conferma dell’ordine.

**3.2.6) Inserire note nei prodotti da acquistare.**

Il cliente dopo aver scelto il prodotto da acquistare ha la possibilità di inserire una nota che si riferiscono a determinate caratteristiche del prodotto. (ad esempio: taglia del pantalone, numero di scarpe)

**3.2.7) Aggiungere i prodotti nel carrello**

il cliente deciso ad acquistare uno o più prodotti potrà aggiungerli al carrello tramite un click “aggiungi al carrello”.

**3.2.8) Acquistare un o più prodotti**

Il cliente una volta nella schermata del carrello può decidere di acquistare o meno i prodotti selezionati ed eventualmente, variarne la quantità.

**3.2.9) Visualizzare ordini**

Il cliente tramite la sezione “I miei ordini”, può visualizzare tutti gli ordini effettuati.

**3.2.10) Scaricare la fattura**

Il cliente in qualsiasi momento, tramite la sezione “I miei ordini”, potrà scaricare la fattura relativa ad un ordine nel formato .pdf, che sarà sempre disponibile permanentemente.

**Il venditore supporta i seguenti requisiti funzionali:**

**3.2.11) Creare un negozio**

Il venditore effettuato l’accesso può creare un negozio personale, scegliendo alcuni aspetti dell’interfaccia utente (layout del menu, colore dei pulsanti) e inserendo informazioni riguardanti al negozio (es: nome, via, partita IVA…).

**3.2.12) Inserire categoria**

Il venditore in qualsiasi momento può inserire una nuova categoria di prodotti inserendo una breve descrizione, che permette al cliente di avere informazioni in anticipo di cosa troverà.

**3.2.13) Inserire un prodotto**

Il venditore ogni volta può aggiungere un nuovo prodotto nella categoria opportuna, inserendo alcune informazioni importanti (prezzo, quantità disponibile, sconto, descrizione).

**3.2.14) Modificare una categoria**

Il venditore può modificare sempre una categoria esistente, ad esempio cambiare la descrizione.

**3.2.15) Modificare un prodotto**

Il venditore può decidere in qualsiasi momento di cambiare informazioni su un prodotto (prezzo, nome prodotto, descrizione, iva, quantità, sconto).

**3.2.16) Cancellare un prodotto dal negozio**

il venditore può decidere di eliminare il prodotto dal proprio catalogo.

**3.2.17) Visualizzare ordini dei clienti**

Il venditore tramite la sezione “controlla ordini”, può visionare l’elenco completo degli ordini effettuati dai propri clienti.

**3.2.18) Gestione delle spedizioni**

Il venditore una volta che il cliente effettua l’ordine, può supporre (motivi di ritardi imprevisti) il tempo d’arrivo dell’ordine e sarà visibile al cliente.

Il **gestore accounting** supporta i seguenti requisiti funzionali:

**3.3.19) Visualizzare i negozi**

Il gestore accounting dopo aver effettuato l’accesso è in grado di visionare tutti i negozi registrati alla piattaforma.

**3.3.20) Modificare un negozio**

Il gestore accounting tramite la sezione “gestione negozi”, può effettuare alcuni cambiamenti di un determinato negozio (colore pagina, design pagina, via, città, Cap, partita IVA e descrizione).

**3.3.21) Cancellare un negozio**

Il gestore accounting tramite la sezione “gestione negozi”, può eliminare dalla piattaforma un determinato negozio.

**3.3.22) Visualizzare i venditori**

Il gestore accounting accedendo alla sezione “gestione venditori”, può visionare l’elenco completo di tutti i venditori registrati alla piattaforma.

**3.3.23) Modificare un venditore**

Il gestore Accounting può modificare alcuni dati del venditore

(E-mail, password, telefono, città).

**3.3.24) Cancellare un venditore**

Il gestore accounting può decidere in qualsiasi momento di eliminare un venditore dalla piattaforma.

**3.3.25) Visualizzare i clienti**

Il gestore accounting tramite l’area “gestione clienti”, può visionare una lista di tutti i clienti registrati alla piattaforma.

**3.3.26) Modificare un cliente**

Il gestore accounting accedendo all’area “gestione clienti”, può effettuare alcune modifiche ai clienti (password, e-mail e dati personali).

**3.3.27) Cancellare un cliente**

Il gestore accounting può decidere di rimuovere un cliente dalla piattaforma tramite un’interfaccia dedicata.

# **Nonfunctional requirements**

## **4.1) Usability:**

**4.1.1) Navigabilità:** Emme Shop deve poter essere navigato e utilizzato agevolmente con qualsiasi tipo di dispositivo: computer, tablet, telefono. Per tanto deve essere realizzato, secondo i criteri del responsive design per renderlo adattabile a ogni tipo di schermo.

**4.1.2) Completezza dei contenuti:** ogni negozio offre informazioni dettagliate su ogni prodotto, permettendo al cliente di trovare immediatamente quello che stava cercando.

**4.1.3) Interfaccia intuitiva:** l’interfaccia di Emme Shop dovrebbe essere semplificata, per permettere all’utente di velocizzare le proprie operazioni. L’interfaccia utente su basa su una finestra principale, che include una barra di campi per accedere con il proprio account (dipende se si è un venditore o cliente), oppure di registrarsi alla piattaforma.

## **4.2) Reliability**

**4.2.1) Efficienza:** Emme shop risposta all’autorizzazione entro 10 secondi, nel 95% dei casi. Qualora ci siano problemi di connettività, il sistema deve comunque garantire la registrazione in locale del pagamento, in modo da permettere il normale acquisto nei negozi. Supporto di transazioni, con caratteristiche ACID per gli acquisti.

## **4.3) Performance**

**4.3.1) Tempi di risposta:** la piattaforma deve essere accessibile in tempo inferiore 15 secondi. Mentre la navigazione tra le pagine deve essere garantita in tempi inferiore ai 6 secondi.

## **4.4) Supportability**

**4.4.1) Multilingua:** nel sito è disponibile una funzionalità che in base alle impostazioni di sistema traduce nella propria lingua.

## **4.5) Implementation**

**4.5.1) Linguaggio di programmazione:** la piattaforma deve essere realizzata nel modello MVC model view control, con iterazione di vari linguaggi, Java, JQUERY, AJAX, HTML, ed altri per realizzare un’interfaccia più dettagliata.

## **4.6) Interface**

**4.6.1) Interface:** Emme Shop si interfaccia con un DBMS configurato su MySQL, per supportare in modo permanente i dati. Per quanto riguarda il web server si utilizza Apache Tomcat versione 8.5.

## **4.7) Packaging**

La piattaforma Emme Shop deve essere disponibile sul Web ad un link specificato. Visualizzabile in tutti i browser consigliato Chrome.

## **4.8) Legal**

**4.8.1) Costo:** la piattaforma ha un costo minimo per chi vuole gestire un proprio negozio online, per permettere agli sviluppatori di tenere la piattaforma sempre aggiornata con aggiunte di funzionalità.

# **System models**

## **5.1) Scenari**

**5.1.1) Creazione del negozio**

Angelo possiede un negozio di abbigliamento. Crede che espandendo il proprio negozio online, trarrebbe maggiore vantaggio economico.

Consigliato da un amico, sceglie la piattaforma Emme Shop.

Quando visita la piattaforma, deve registrarsi con i propri dati personali, dopodiché gli verrà chiesto se creare in questo momento il negozio o al prossimo accesso. Sceglie di crearlo subito. Gli vengono richiesti dati sul negozio P.IVA, indirizzo, breve descrizione del negozio, scegliendo temi e logo del proprio negozio.

Dopo aver confermato la creazione del negozio verrà reindirizzato alla home del negozio, dove con una semplice e intuitiva grafica inserisce i propri articoli con tutte le informazioni del prodotto.

Nel negozio di Angelo sono arrivati i capi della nuova linea di abbigliamento invernale. Desidera aggiungerli al proprio sito, togliendo i capi dello scorso anno. Va sul sito Emme Shop, accedendo con il proprio account, vengono mostrati i propri prodotti, con appositi pulsanti decide di eliminarli dai propri prodotti. Per far posto ad una nuova categoria per le nuove uscite invernali.

**5.1.2) Acquistare da un negozio**

Luca sui social network vede un nuovo post di un nuovo negozio aggiunto ad’ Emme Shop. Visita il sito, accede con le proprie credenziali, e si trova di fronte a tutti i negozi presenti sulla piattaforma. Con una breve ricerca trova il negozio di Angelo su abbigliamento, cliccando visualizza i prodotti venduti dal negozio.

Luca aggiunge al carrello i prodotti di suo piacimento, dopodiché prende visione nel carrello dei possibili acquisti, e procede all’ordine.

Angelo confermerà l’ordine, che sarà trasmesso a Marco che si occuperà della spedizione.

## **5.2) Use Case Model**

**5.2.1) Attori:**

Identifichiamo i seguenti attori:

**Cliente:**

Il cliente rappresenta colui che va in cerca di prodotti da acquistare.

**Venditore:**

Il venditore e colui che possiede un negozio iscritto alla piattaforma. Gestisce il proprio catalogo di categorie e prodotti presenti nel negozio.

**Gestore Accounting:**

Amministratore è il gestore dell’intera piattaforma. Gestisce gli utenti iscritti, sia Clienti che Venditori. Ha delle funzionalità specifiche per ciascuno di essi, sono descritte di seguito.

**Gestore delle spedizioni:**

Colui che compie tutte le operazioni dalla presa incarico del ordine, alla spedizione, e la consegna.

## **Use Case Diagram**

Immagine che contiene mappa, testo

Descrizione generata automaticamente

Immagine che contiene testo

Descrizione generata automaticamente

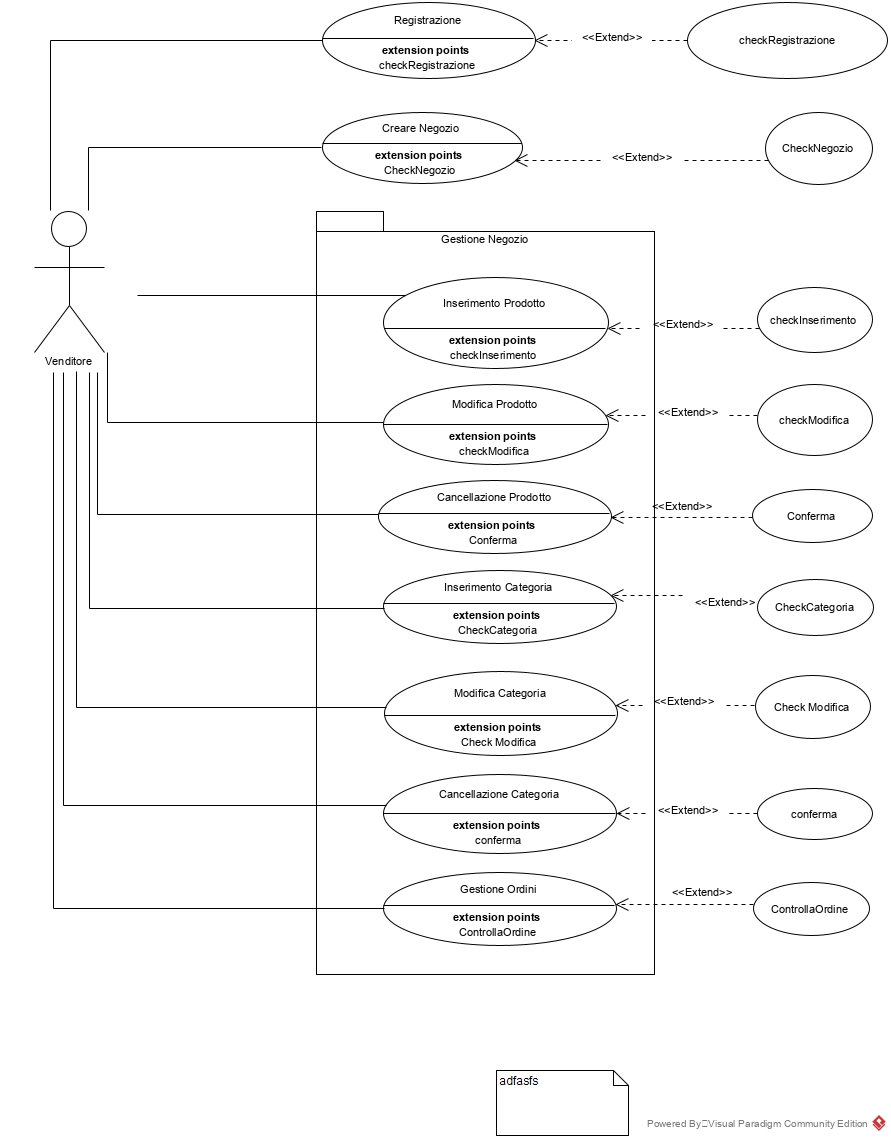
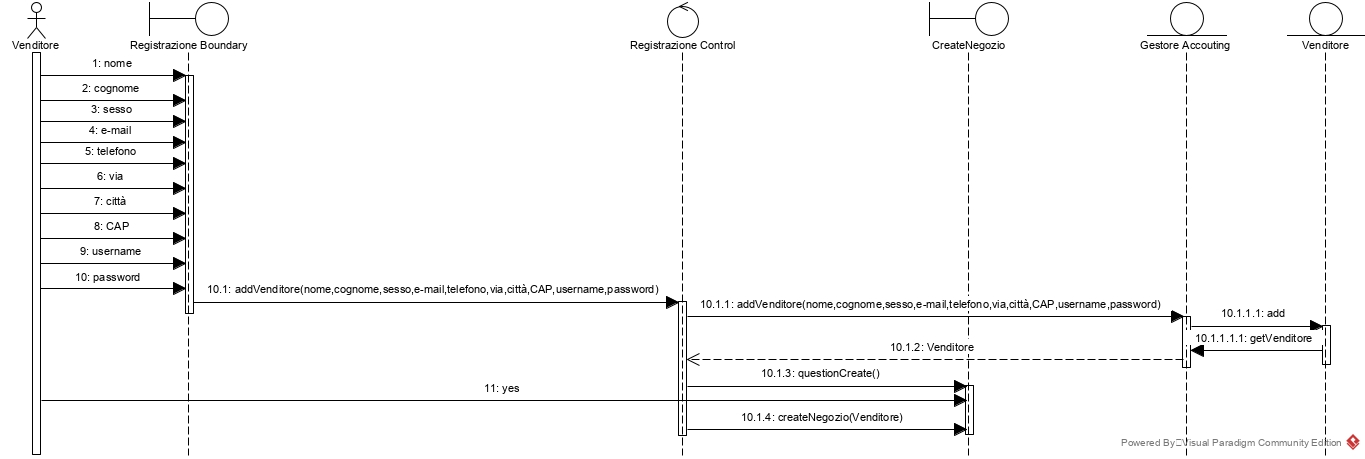


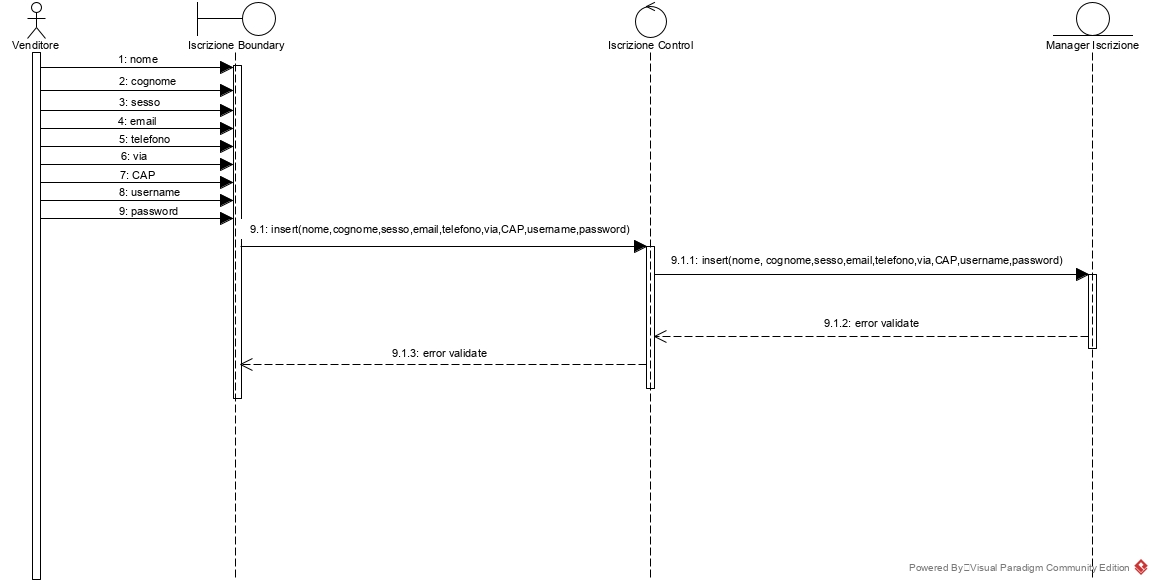
Immagine che contiene testo

Descrizione generata automaticamente

5.2.2) Registrazione venditore

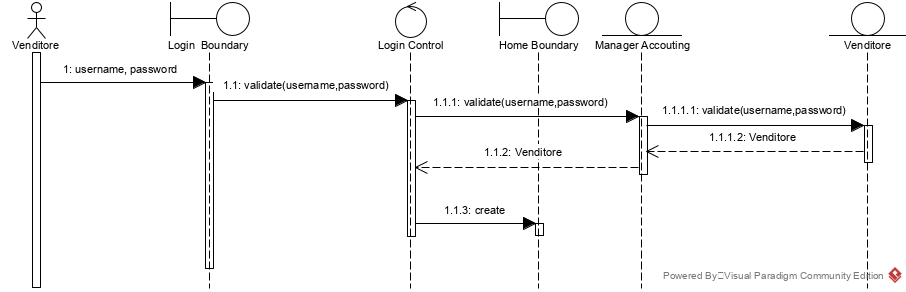
|  |  |  |
| --- | --- | --- |
| **Use case name** | Registrazione Venditore | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Il Venditore non possiede un account, si trova nella home page del negozio | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Accede al form per la registrazione.  2.1 Il venditore compila i seguenti campi: “Nome”, “Cognome”, “sesso”, “Email”, “Telefono”, “Via”, “Città”, “CAP”, “Username”, “Password”.  2.2 Il Venditore clicca sul bottone “Completa la Registrazione”  3.Dopo aver compilato i campi il venditore può decidere se creare il proprio negozio.  3.2.1 Se il venditore decide di non creare il negozio, viene reindirizzato sulla home page. | 2.Il sistema reindirizza il venditore in un form di registrazione.  3.1 Il sistema registra tutte le informazioni.  3.2 Rimanda il venditore alla creazione del negozio. |
| **Exit Condition** | Il caso d’uso termina quando i dati inseriti dal venditore saranno resi persistenti. | |
| **Exception/**  **Alternating Flows** | **5.2.2.1 checkRegistrazione:**  * 1. L’eccezione inizia nel punto 2.1 nel caso in cui uno o più campi sono sbagliati.   2. Il sistema reindirizza l’utente sullo stesso form evidenziando gli errori e non fa procedere con la registrazione fino a quando i dati inseriti non sono corretti.   2. Il percorso alternativo si collega al punto 3.1 | |
| **Formato dati** | [link](#_Iscrizione_Venditore) | |





5.2.3) Login Venditore

|  |  |  |
| --- | --- | --- |
| **Use case name** | Login Venditore | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Il Venditore possiede un account, si trova nella home page del negozio | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Entra nella home page del sito e trova la sezione per effettuare il login.  2.1. Il venditore per effettuare l’accesso compila i campi “Username”, “Password” e accede.  3.Dopo aver compilato i campi il venditore invia le informazioni al sistema.  3.1.2 Il venditore crea il proprio negozio inserendo le informazioni necessarie. (vedi prossimo caso d’uso) | 2.Il sistema reindirizza il venditore in un modulo per loggarsi.  3.1 Il sistema controlla se il negozio è già esistente .  3.1.1 Se il negozio è già esistente, allora, il sistema reindirizza il cliente nella creazione dello shop. (vedi prossimo caso d’uso)  3.2 Il sistema rimanda il venditore nella sua pagina riservata. |
| **Exit Condition** | Il caso d’uso termina quando le il venditore inserisce il proprio negozio oppure quando viene reindirizzato sulla sua pagina riservata. | |
| **Exception/**  **Alternating Flows** | **)errorLogin:**   * 1. L’eccezione inizia nel punto 3 In caso di accesso non valido.   2. Il sistema risponderà mostrando un messaggio con scritto “Cliente non trovato o password errata!”   2. Il percorso alternativo si collega al punto 2 | |



* ErrorLogin

### 

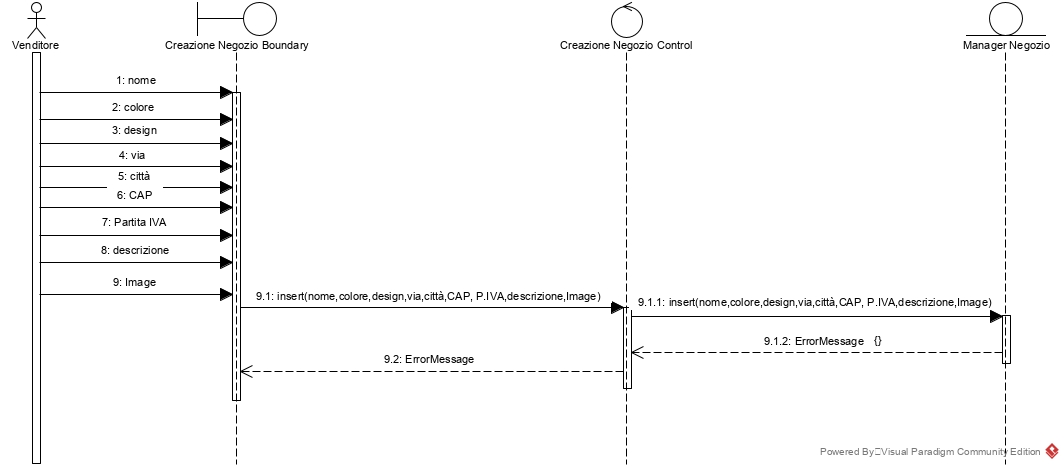
5.2.4) Creazione del negozio

|  |  |  |
| --- | --- | --- |
| **Use case name** | Creazione del negozio | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Il Venditore possiede un account, ed ha già effettuato accesso, non ha ancora creato il negozio. | |
| **Flow of Events** | Actors Steps | System Steps |
| 2.1 Il venditore compila i campi “Nome Shop”, “Colore”, “Design menù”, “Via”, “Città”, “CAP”, “Partita IVA”, “Breve Descrizione “. Inoltre il venditore può caricare l’immagine del proprio negozio cliccando su “scegli file” e poi “carica Immagine” o completare la creazione del negozio senza aggiungere una  3. Le informazioni inserite vengono inviate al sistema. | 1.Il sistema reindirizza il venditore in un form per inserire le informazioni necessarie per il negozio.  3.1 Il sistema salva i dati inseriti.  3.2 Rimanda il venditore nel Home del negozio. |
| **Exit Condition** | Il caso d’uso termina quando il venditore crea il proprio negozio e le informazioni saranno rese persistenti. | |
| **Exception/**  **Alternating Flows** | **5.2.4.1 checkCampi:**  * 1. L’eccezione inizia nel punto 2.1 nel caso in cui uno o più campi sono sbagliati.   2. Il sistema reindirizza l’utente sullo stesso form evidenziando gli errori e non fa procedere con la creazione del negozio fino a quando i dati inseriti non sono corretti. Il percorso alternativo si collega al punto 3. | |
| **Formato dati** | [link](#_Creazione_del_negozio) | |

Immagine che contiene screenshot, mappa

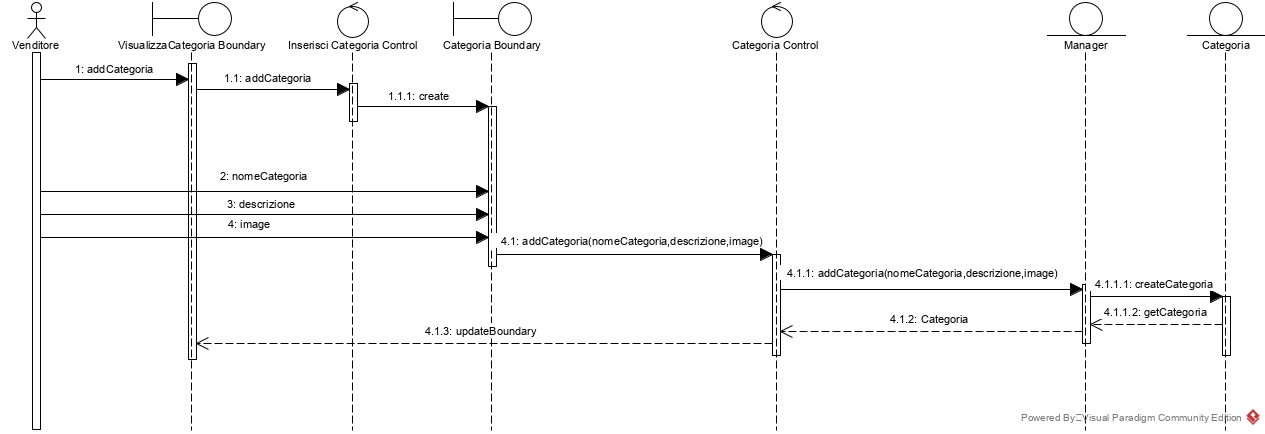
Descrizione generata automaticamente

-Errore Creazione Negozio



5.2.5) Inserisci categoria

|  |  |  |
| --- | --- | --- |
| **Use case name** | Inserisci categoria | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Il venditore possiede un account e un proprio negozio. Il venditore si trova nella sua home page del negozio. | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Il venditore clicca su “inserisci nuova categoria”  2.2. Il venditore compila i campi “Categoria”, “Descrizione” (opzionale).  2.2.1. Il venditore può decidere se caricare l’immagine della categoria” e poi “carica immagine” oppure completare direttamente l’inserimento della categoria  3. Il venditore conferma i dati inseriti. | 2.1. Il sistema reindirizza il venditore in un form per inserire i dati della categoria.  3.1. Il sistema salva i dati inseriti.  3.2 rimanda l’utente all’home delle categorie |
| **Exit Condition** | Il caso d’uso termina quando il venditore inserisce la categoria e le informazioni vengono rese persistenti e viene rimandato nella propria home del negozio. | |
| **Exception/**  **Alternating Flows** | **5.2.5.1 checkCampi:**  * 1. L’eccezione inizia nel punto 2.2 nel caso in cui uno o più campi sono sbagliati.   2. Il sistema reindirizza l’utente sullo stesso form evidenziando gli errori e non fa procedere con l’inserimento della categoria fino a quando i dati inseriti non sono corretti.   2. Il percorso alternativo si collega al punto 3. | |
| **Formato dati** | [link](#_Inserisci_Categoria) | |



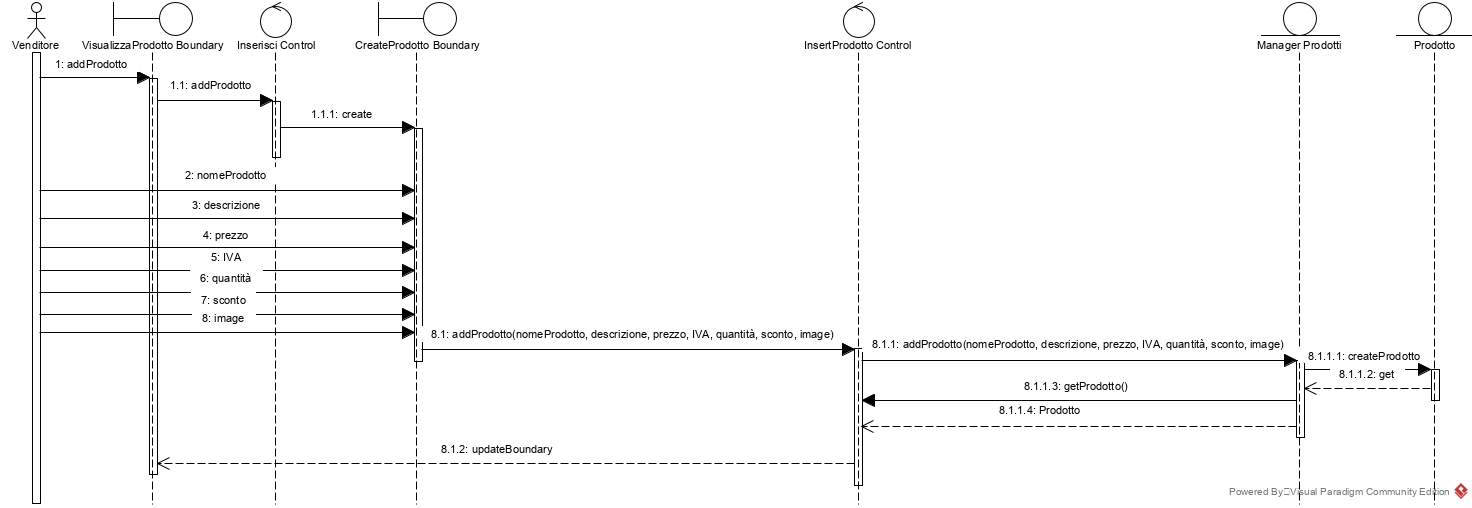
-Error CheckCampi

Immagine che contiene screenshot

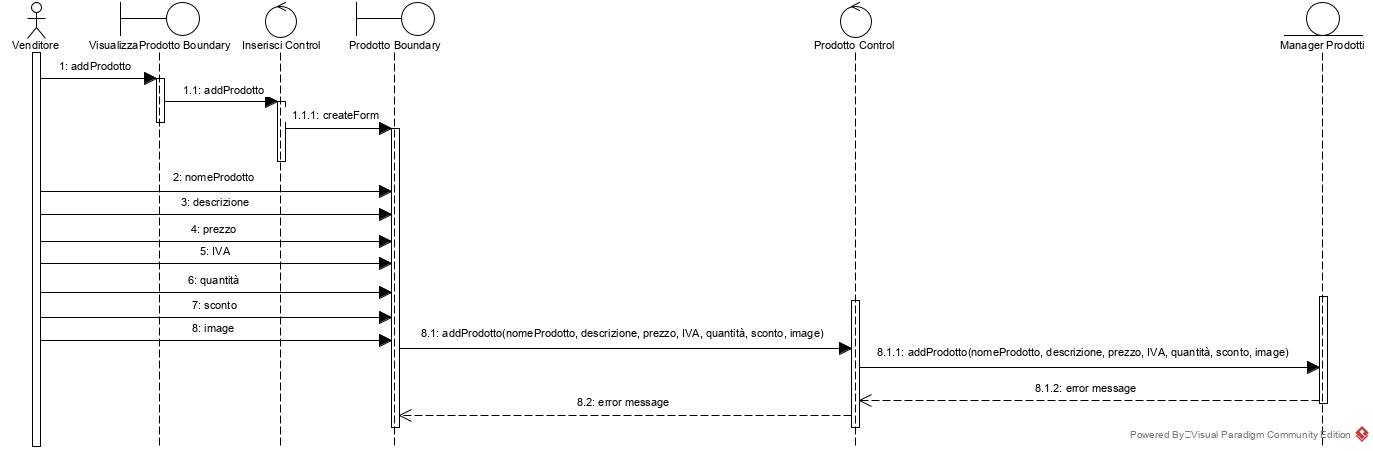
Descrizione generata automaticamente

5.2.6) Inserisci prodotto

|  |  |  |
| --- | --- | --- |
| **Use case name** | Inserisci Prodotto | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Il Venditore possiede un account e un proprio negozio. Dopo essersi loggato si trova nella propria home. | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Il venditore decide di “inserire un nuovo prodotto”  2.2. Il venditore compila i campi “Nome Prodotto”, “Descrizione”, “Prezzo”, “IVA”, “Quantità”, “Sconto”  2.2.1. Il venditore può decidere se caricare l’immagine del prodotto “carica immagine” oppure completare direttamente l’inserimento della categoria senza un immagine.  3. Il venditore conferma i dati inseriti. | 2.1. Il sistema reindirizza il venditore in un form per inserire i dati del prodotto.  3.1. Il sistema salva i dati inseriti.  3.2 Rimanda il venditore nella pagina dei prodotti. |
| **Exit Condition** | Il caso d’uso termina il venditore inserisce un prodotto nel negozio. | |
| **Exception/**  **Alternating Flows** | **5.2.6.1 checkCampi:**  * 1. L’eccezione inizia nel punto 2.2 nel caso in cui uno o più campi sono sbagliati.   2. Il sistema reindirizza l’utente sullo stesso form evidenziando gli errori e non fa procedere con l’inserimento del prodotto fino a quando i dati inseriti non sono corretti.   2. Il percorso alternativo si collega al punto 3. | |
| **Formato dati** | [link](#_Inserisci_Prodotto) | |



-Errore Campi inserimento prodotto

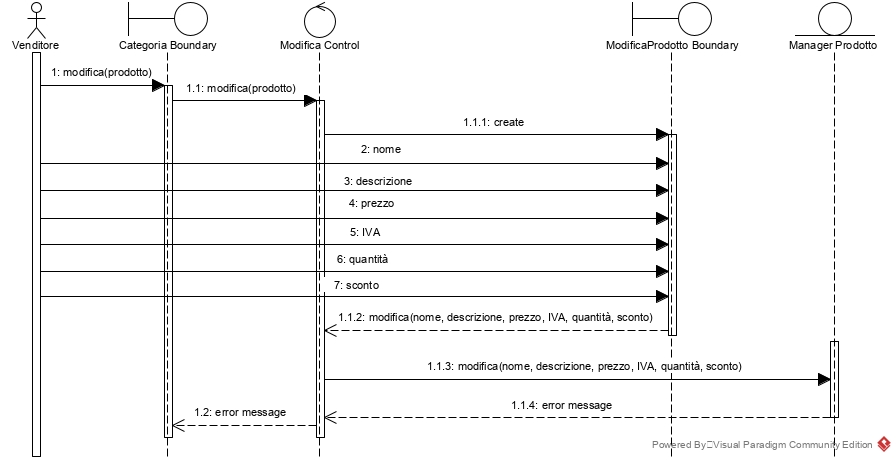


5.2.7) Modifiche categorie

|  |  |  |
| --- | --- | --- |
| **Use case name** | Modifica Categoria | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Il Venditore possiede un account e un proprio negozio. È loggato e si trova nella propria home | |
| **Flow of Events** | Actors Steps | System Steps |
| 1. Cerca la categoria da modificare nella sua area riservata.  2.1. Il venditore modifica il campo “Descrizione”, poi conferma la modifica.  2.1.1. Il venditore può decidere se modificare l’immagine del prodotto cliccando su “scegli file” e poi “carica immagine” oppure modificare direttamente il prodotto della categoria cliccando su “Completa inserimento”.  3.Dopo aver apportato le modifiche ai campi, il venditore conferma la modifica | 2.Il sistema reindirizza il venditore in un form di modifica delle info della categoria.  3.1 Il sistema salva tutte le modifiche aggiornando i campi  3.2 Rimanda il venditore alla home. |
| **Exit Condition** | Il caso d’uso termina quando le modifiche apportate alla categoria saranno rese persistenti. | |
| **Exception/**  **Alternating Flows** | **5.2.7.1 checkCampi:**  * 1. L’eccezione inizia nel punto 2.1 nel caso in cui uno o più campi sono sbagliati.   2. Il sistema reindirizza l’utente sullo stesso form evidenziando gli errori e non fa procedere con la modifica del prodotto fino a quando i dati inseriti non sono corretti.   2. Il percorso alternativo si collega al punto 3. | |
| **Formato dati** | [link](#_Inserisci_Categoria) | |

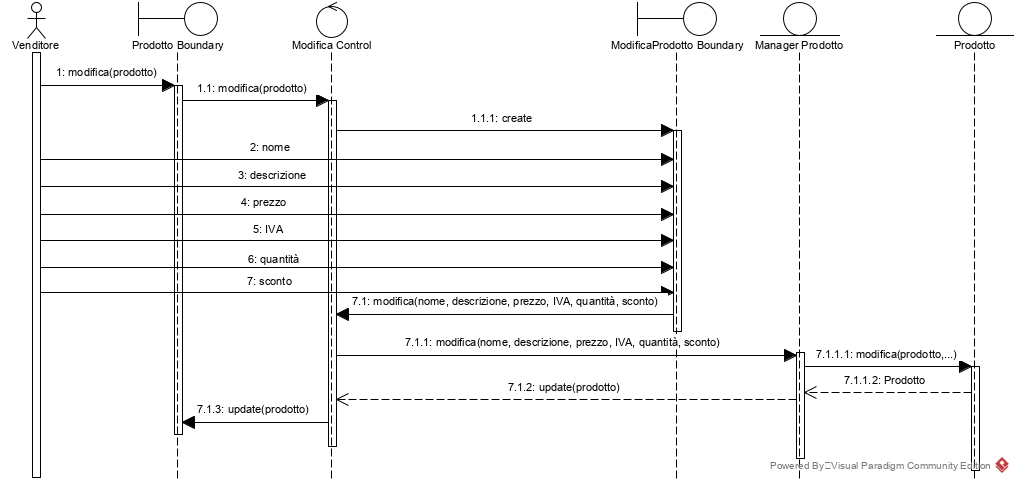
### 

-Errore Campi



5.2.8) Modifiche prodotto

|  |  |  |
| --- | --- | --- |
| **Use case name** | Modifiche prodotto | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Venditore deve essere loggato, si trova nell’ home page del proprio negozio | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Il venditore seleziona il prodotto, per modificarlo.  3.Il venditore può modificare i seguenti campi: “nome”, “descrizione”, “prezzo”, “IVA”, “quantità disponibile”, “sconto” (campi opzionali altrimenti restano quelli già inseriti). Conferma i dati  3.3 Il venditore può caricare una nuova immagine del prodotto (opzionale) | 2.Il sistema reindirizza il venditore in un form di modifica delle info del prodotto.  3.2 Il sistema reindirizza su un altro form per aggiornare immagine del prodotto.  4.1 Il sistema salva tutte le modifiche aggiornando i campi.  4.2 Rimanda il venditore alla pagina del prodotto. |
| **Exit Condition** | Il caso d’uso termina quando le modifiche apportate al prodotto saranno rese persistenti. | |
| **Exception/ Alternating Flows** | **5.2.8.1 notUploadImage:**  si attiva 3.3, il Venditore può decidere di non caricare una nuova immagine del prodotto, confermando direttamente senza caricare una nuova immagine. Riprende dal punto 4.1 | |
| **Formato dati** | [link](#_Inserisci_Prodotto) | |



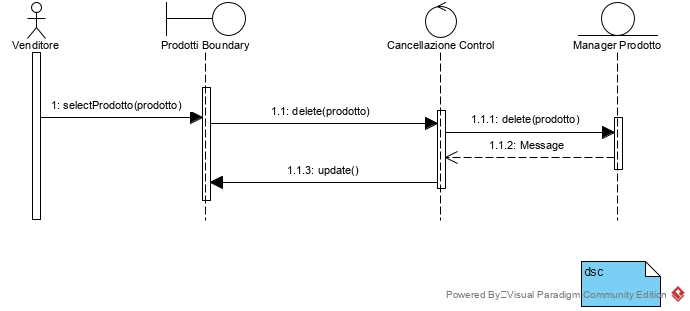
-Errore Modifica Prodotto campi errati

Immagine che contiene screenshot

Descrizione generata automaticamente

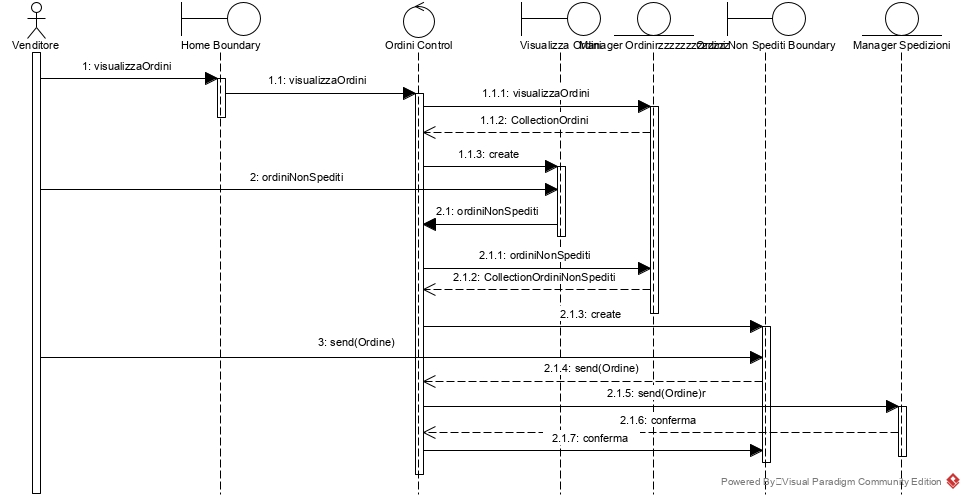
5.2.9) Cancellazione prodotto

|  |  |  |
| --- | --- | --- |
| **Use case name** | Cancellazione prodotto | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Venditore deve essere loggato, si trova nell’ home page del proprio negozio | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Cerca il prodotto da cancellare.  2. Il venditore seleziona rimuovi sull’opportuno prodotto da cancellare | 3. Il sistema cancella il prodotto.  3.1 rende permanenti le modifiche  3.2 Aggiorna la pagina dei prodotti, senza il prodotto appena rimosso. |
| **Exit Condition** | Il caso d’uso termina quando il prodotto sarà cancellato dal sistema. | |



5.2.10) Gestione degli ordini

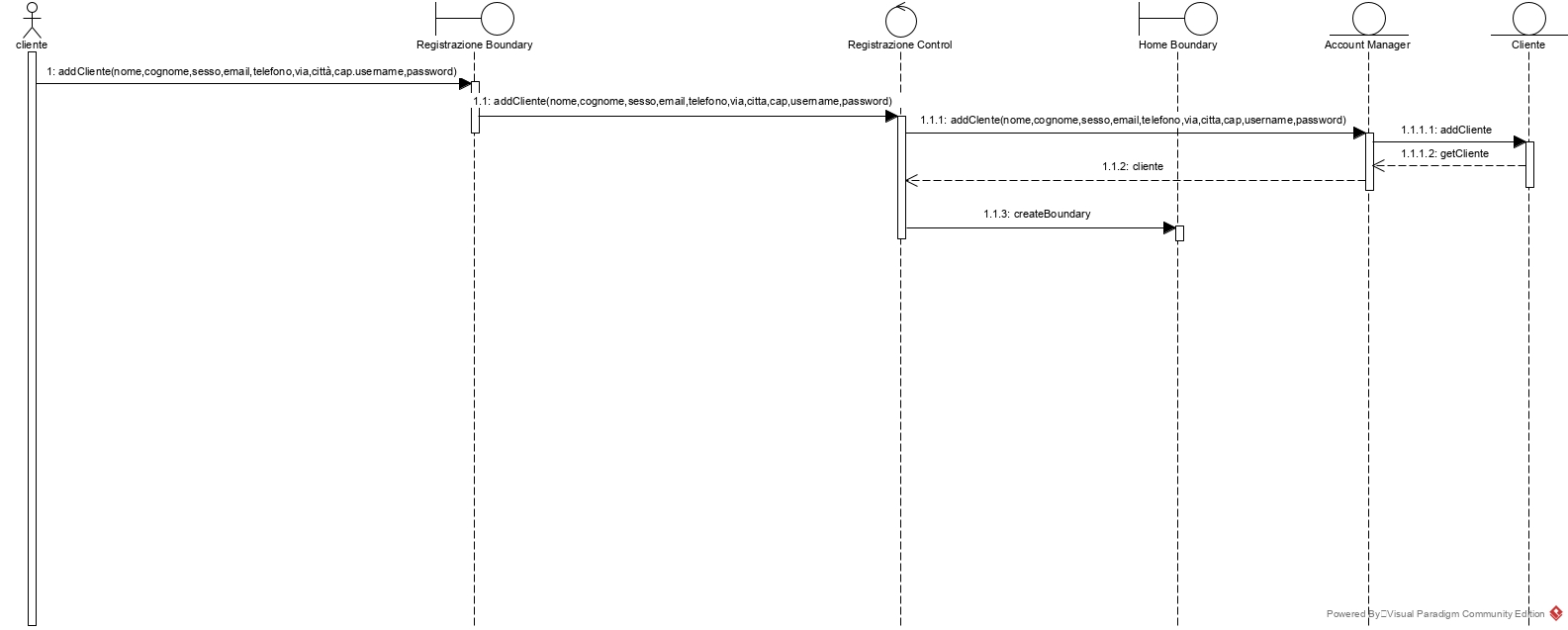
|  |  |  |
| --- | --- | --- |
| **Use case name** | Gestione degli ordini | |
| **Participating Actors** | Venditore | |
| **Entry Condition** | Venditore deve essere loggato, si trova nell’ home page del proprio negozio | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Il venditore entra nella sezione ordini non confermati.  2.1 Il venditore cerca le spedizioni ancora da spedire  3.Conferma gli ordini da far spedire, scegliendo un opportuno gestore delle spedizioni da quelle presenti. | 2.Il sistema visualizza tutti gli ordini.  4.Il sistema invia tutti i dati al gestore delle spedizioni che si occupa del trasporto e consegna. |
| **Exit Condition** | Il caso d’uso termina quando l’ordine è stato confermato | |



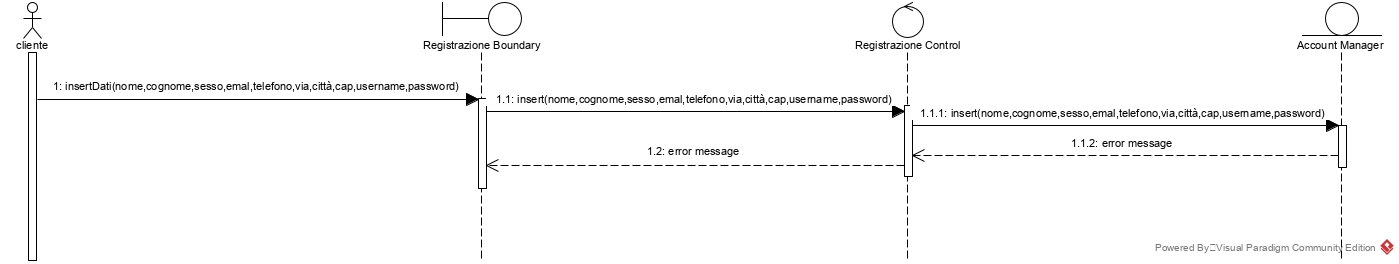
5.2.11) Registrazione Cliente

|  |  |  |
| --- | --- | --- |
| **Use case name** | Iscrizione | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Il Cliente visita il sito e decide di registrarsi alla piattaforma | |
| **Flow of Events** | **Actors Steps** | **System Steps** |
| **1.** Entra nella home page del sito e clicca sulla sezione dedicata alla registrazione.  **2.1** compila il modulo a “step”, inserendo i seguenti campi: “Nome”, “Cognome”, “sesso”, “Email”, “Telefono”, “Via”, “Città”, “CAP”, “Username”, “Password”.  **2.2** Il cliente clicca sul bottone “Completa la Registrazione” | **2**. Il sistema reindirizza il cliente in un form di registrazione.  **3.** Il sistema registra tutte le informazioni del cliente.  **3.1** Rimanda il cliente alla “home page” |
| **Exit Condition** | Il caso d’uso termina quando il cliente si trova davanti la sezione “accedi” | |
| **Exceptions/alternating flows** | **5.2.11.1 checkCampi:** **1.** L’eccezione inizia nel punto **2.1** nel caso in cui uno o più campi sono sbagliati.  **1.1** Il sistema reindirizza il cliente sullo stesso form evidenziando gli errori e non fa procedere con la registrazione fino a quando i dati inseriti non sono corretti.  **2.** Il percorso alternativo si collega al punto **3** | |

### 

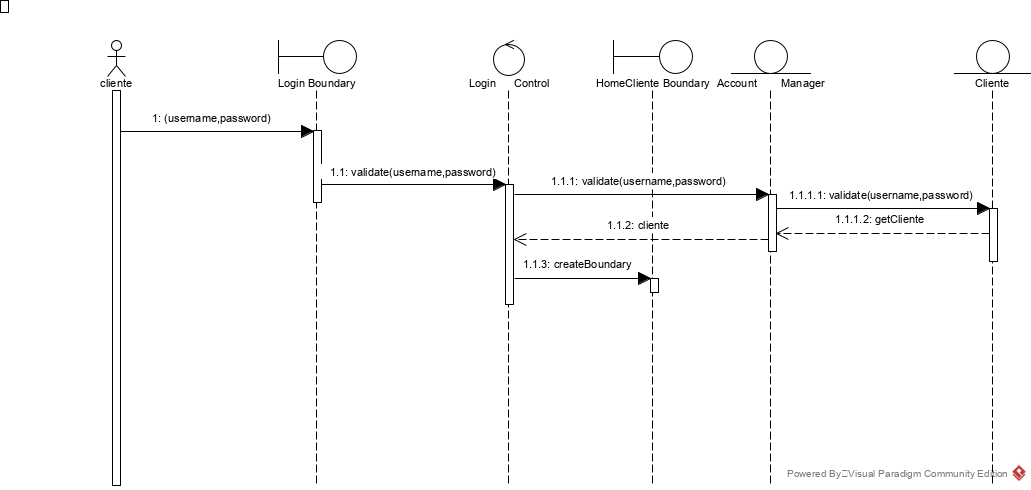


-Errore Registrazione



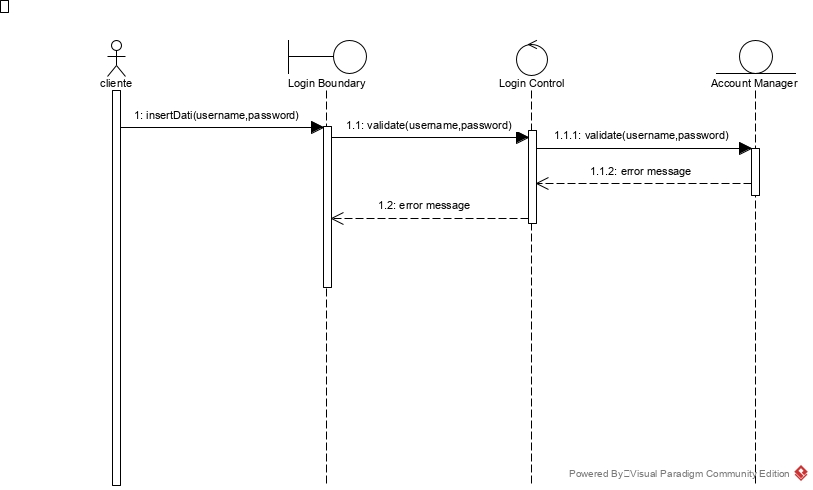
5.2.12) Login cliente

|  |  |  |
| --- | --- | --- |
| **Use case name** | Login cliente | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Il Cliente deve possedere un account | |
| **Flow of Events** | **Actors Steps** | **System Steps** |
| **1.** Entra nella home page del sito e trova la sezione per effettuare il login.  **2.1.** Il cliente per effettuare l’accesso compila i campi “Username”, “Password” e dopo conferma per accedere.  **2.2** Dopodiché le informazioni del cliente vengono inviate al sistema. | **2.** Il sistema reindirizza il cliente in un modulo per l’inserimento delle credenziali d’accesso.  **3.** Il sistema salva i dati, controllando i formati, dopodichè reindirizza il cliente nella pagina “autorizzata”. |
| **Exit Condition** | Il caso d’uso termina quando il cliente visualizza la sua pagina autorizzata | |
| **Exceptions/alternating flows** | **5.2.12.1 errorLogin:**  **1.1** L’eccezione inizia nel punto **2.2** in caso di accesso non valido.  **1.2** il sistema reindirizza il cliente alla home page mostrando un messaggio “cliente non trovato” o “password errata”  **2.** il percorso alternativo si collega al punto **3.** | |



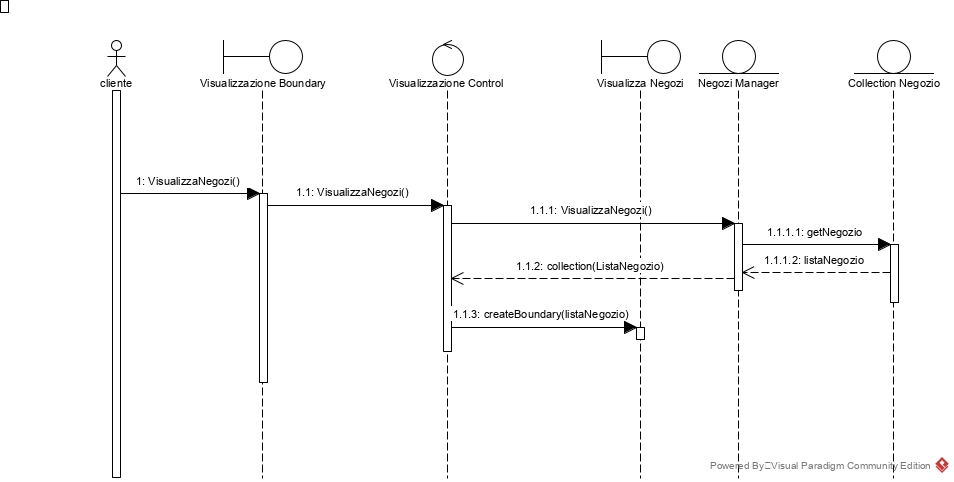
### 

-Error Login Cliente



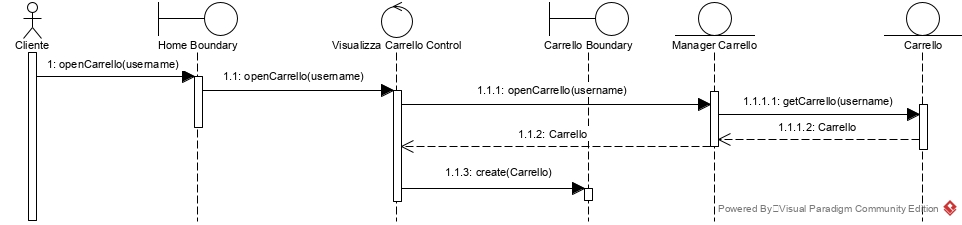
5.2.13) Visualizzazione negozio

|  |  |  |
| --- | --- | --- |
| **Use case name** | Visualizzazione negozio | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Il Cliente deve aver effettuato l’accesso | |
| **Flow of Events** | **Actors Steps** | **System Steps** |
| **1.** il cliente effettua l’accesso. | 2. Il sistema recupera tutti i negozi iscritti alla piattaforma.  3. Reindirizza utente sulla page che visualizza tutti i negozi. |
| **Exit Condition** | Il caso d’uso termina quando il cliente visualizza i negozi | |



5.2.14) Visualizza carello

|  |  |  |
| --- | --- | --- |
| **Use case name** | Visualizza carrello | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Il Cliente deve essere loggato, e deve aver inserito almeno un prodotto nel carrello | |
| **Flow of Events** | **Actors Steps** | **System Steps** |
| **1.** il cliente va nella sezione dedicata al carrello | **2.** il sistema reindirizza il cliente nell’area dedicata al carrello.  **2.1** Dopodiché il sistema recupera i prodotti inseriti nel carrello.  3.1 Reindirizza il cliente nella page del carello |
| **Exit Condition** | Il caso d’uso termina quando il cliente visualizza la pagina dedicata al carrello | |



### 

5.2.15) Aggiunta prodotto al carello

|  |  |  |
| --- | --- | --- |
| **Use case name** | Aggiunta prodotto al carrello | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Il cliente deve essere loggato, si trova nell’ home page. Dopo aver deciso in quale negozio comprare. | |
| **Flow of Events** | Actors Steps | System Steps |
| 1.Cliente sceglie il prodotto desidera.  2. Il cliente seleziona la quantità di prodotto desiderata che vuole, e può inserire una nota per il venditore inerente al prodotto. Lo aggiunge al carrello | 3. Il sistema conferma avvenuta aggiunta al carrello. |
| **Exit Condition** | Il caso d’uso termina quando il prodotto è aggiunto al carello son successo. | |
| **Exception/ Alternating Flows** | **5.2.15.1 pathAlternativo:**  1. Il cliente seleziona il negozio che contiene il prodotto da aggiungere al carrello.    1. Il sistema visualizza tutte le categorie del determinato negozio. 2. Il cliente seleziona la categoria desiderata    1. Il percorso alternativo si collega al principale al punto 2  **5.2.15.2 quantitàNonDisponibile:**  Questa eccezione si verifica quando il prodotto viene aggiunto al carrello, ma se la quantità di prodotto specificata non risulta presente, il sistema avvisa il cliente del problema. | |

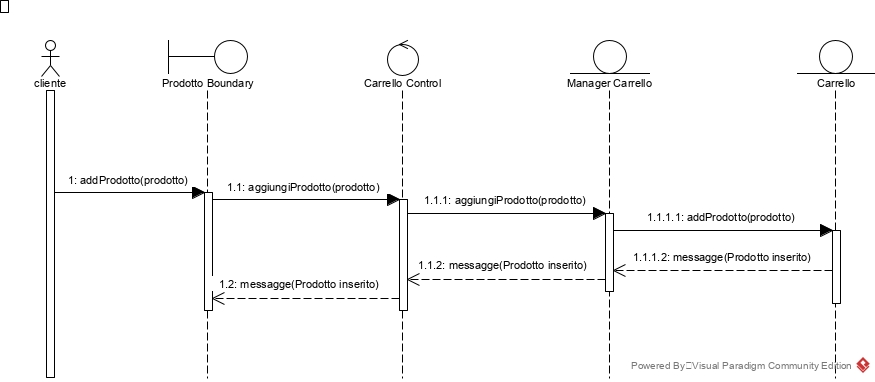
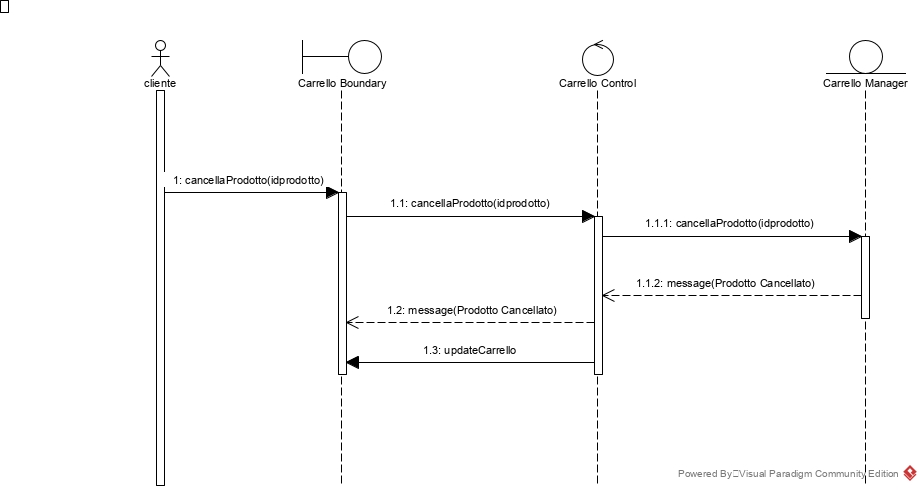


Immagine che contiene screenshot

Descrizione generata automaticamente

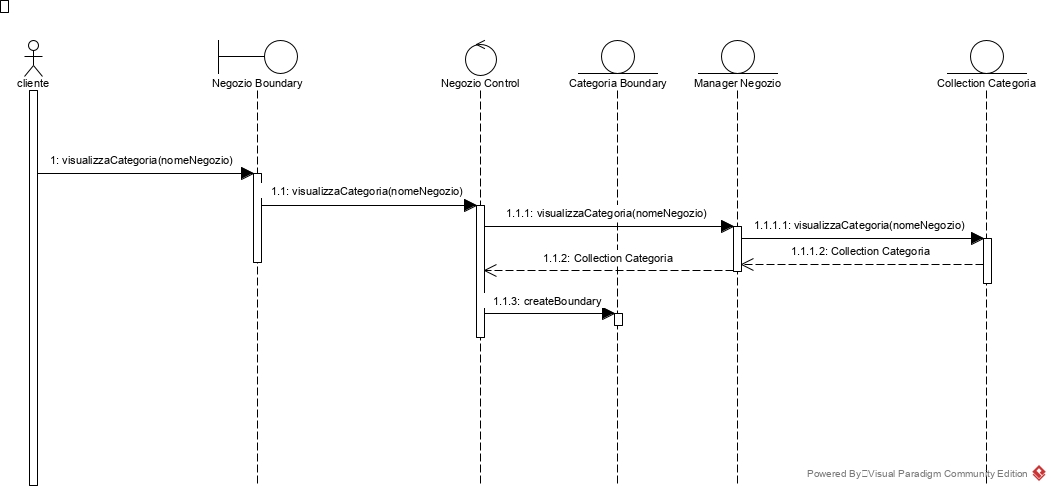
5.2.16) Rimozione prodotto (dal Carrello)

|  |  |  |
| --- | --- | --- |
| **Use case name** | Rimozione prodotto | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Il cliente deve essere loggato, si trova nel proprio carrello | |
| **Flow of Events** | Actors Steps | System Steps |
| 2.Il cliente seleziona il prodotto che deve rimuovere.  4. Utente conferma la rimozione | 1.Sistema visualizza tutti i prodotti aggiunti nel carrello, durante la sessione di acquisti.  3.Il sistema manda un messaggio di conferma di rimozione.  5.Il sistema rimuove il prodotto e reindirizza il cliente nel carrello. |
| **Exit Condition** | Il caso d’uso termina quando il prodotto selezionato è stato rimosso dal carrello | |



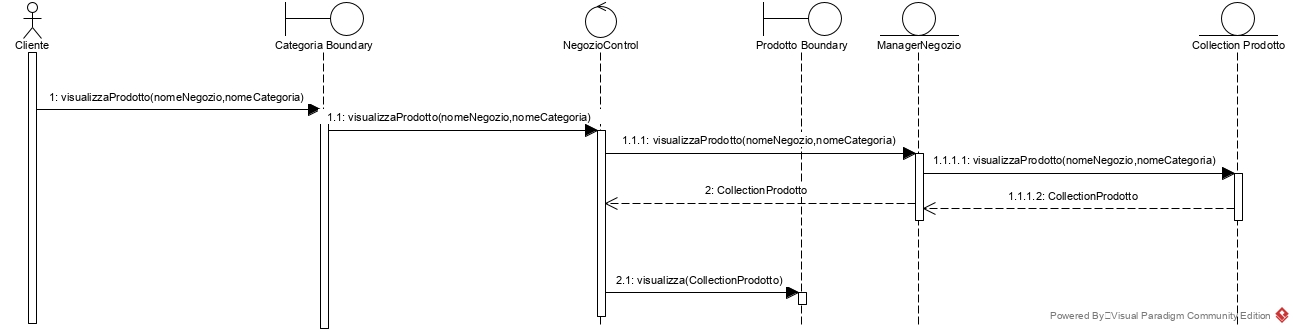
5.2.18) Visualizzazione categorie

|  |  |  |
| --- | --- | --- |
| **Use case name** | Visualizzazione categorie | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Cliente deve essere loggato, si trova nella propria Home | |
| **Flow of Events** | Actors Steps | System Steps |
| 1. il cliente clicca su un determinato negozio | 2. il sistema tramite il “nome del negozio” recupera le categorie associate.  2.1 e reindirizza il cliente nella pagina dedicata alle categorie. |
| **Exit Condition** | Il caso d’uso termina quando il cliente visualizza le categorie di un negozio. | |



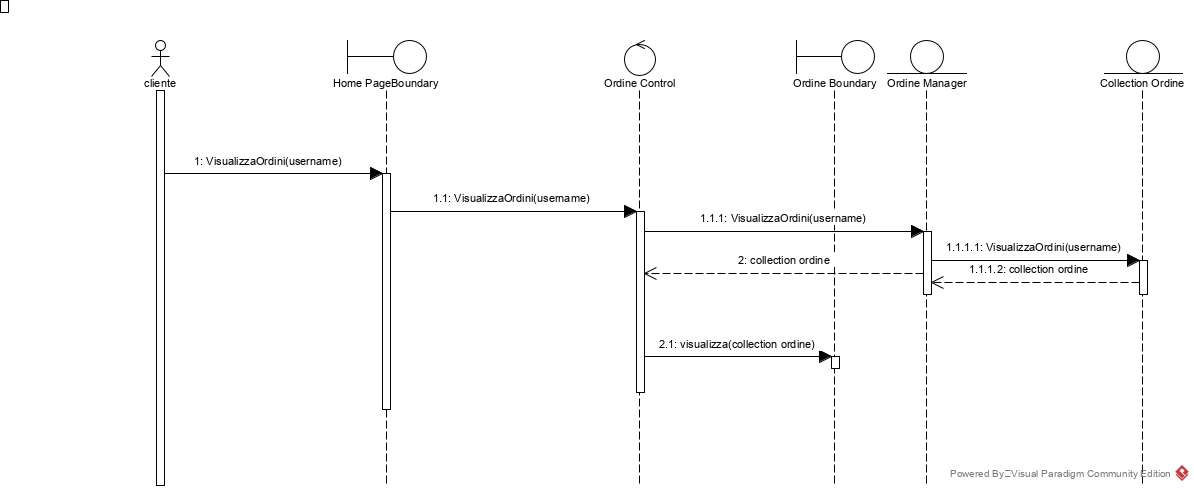
5.2.19) Visualizzazione prodotti

|  |  |  |
| --- | --- | --- |
| **Use case name** | Visualizzazione prodotti | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Cliente deve essere loggato | |
| **Flow of Events** | Actors Steps | System Steps |
| 1. il cliente clicca su una determinata catrgorie. | 2. il sistema tramite il “nome della categoria” recupera i prodotti associati.  2.1 e reindirizza il cliente nella pagina dedicata ai prodotti. |
| **Exit Condition** | Il caso d’uso termina quando il cliente visualizza i prodotti di una determinata categorie | |



5.2.20) Visualizzazione ordini

|  |  |  |
| --- | --- | --- |
| **Use case name** | Visualizzazione ordini | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Cliente deve essere loggato | |
| **Flow of Events** | Actors Steps | System Steps |
| 1. il cliente clicca sulla sezione “i miei ordini”. | 2. il sistema tramite il “username cliente” recupera gli ordini effettuati.  2.1 e reindirizza il cliente nella pagina dedicata agli ordini. |
| **Exit Condition** | Il caso d’uso termina quando il cliente visualizza tutti gli ordini effettuati. | |



5.2.21) Conferma un ordine

|  |  |  |
| --- | --- | --- |
| **Use case name** | Conferma un ordine | |
| **Participating Actors** | Cliente | |
| **Entry Condition** | Il cliente deve essere loggato, si trova nell’ area carrello | |
| **Flow of Events** | Actors Steps | System Steps |
| 2.Il cliente dopo aver visualizzato gli articoli presenti, con le quantità richieste, procede alla conferma dell’ordine, tramite bottone “conferma ordine”.  4. Il cliente visualizza il messaggio “ordine confermato”. | 1.Sistema visualizza tutti i prodotti aggiunti nel carrello, durante la sessione di acquisti  3.1 Il sistema riceve l’informazione, e memorizza l’ordine.  3.2 Il sistema avvisa il cliente dell’avvenuto ordine, visualizza schermata degli ordini effettuati.  5. Il sistema reindirizza il cliente nella pagina “ordine effettuati” |
| **Exit Condition** | Il caso d’uso termina quando ordine effettuato viene confermato, e il sistema visualizza la pagina “ordine effettuati” visualizzando l’ordine appena effettuato. | |
| **Exception/ Alternating Flows** | **5.2.17.1 checkOrdine:**  Eccezione si può verificare al punto 3) In caso non sia disponibile un prodotto, o la quantità desiderata il sistema avvisa il compratore al momento della conferma dell’ordine, bloccando l’ordine e indicando i problemi riscontrati. **5.2.17.2 nessunProdottonelCarrello:**   1. Inizia al punto 3. Nel caso in cui non ci sia almeno un prodotto nel carrello. 2. Il sistema risponderà con un messaggio al cliente “Non ci sono prodotti nel carrello”. | |

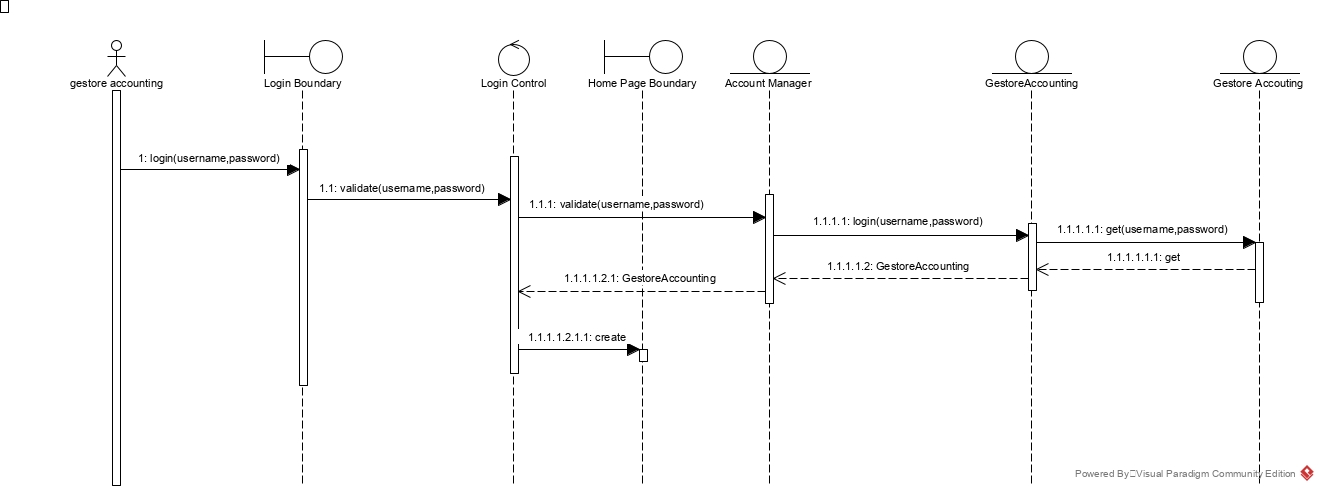
### 

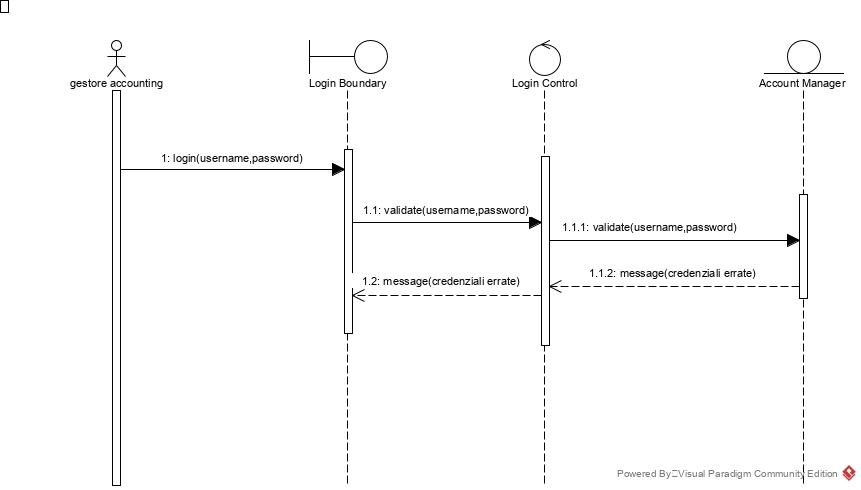
Immagine che contiene mappa

Descrizione generata automaticamente

5.2.22) Login Amministratore

|  |  |  |
| --- | --- | --- |
| **Use case name** | Login Gestore Accounting | |
| **Participating**  **Actors** | Gestore Accounting | |
| **Entry Condition** | Il Gestore Accounting possiede l’account deve effettuare il login | |
| **Flow of Events** | Actors Steps | System Steps |
|  | 1.Entra nella home del sito e trova la sezione per effettuare il login.  Clicca sul link dedicato.  3.Il venditore per effettuare l’ accesso completa i campi ”Username” e ”Password” e dopo clicca sul bottone “Accedi”.  3.1.2 Il Gestore Accounting è entrato nella propria area. | 2.Il sistema reindirizza il Gestore Accounting in un modulo per l’inserimento delle credenziali.  3.1 Il sistema controlla le credenziali  3.2 Se le credenziali sono corrette il sistema reindirizza il Gestore Accounting alla pagina dedicata alla gestione. |
| **Exit Condition** | Il caso d' uso termina quando il Gestore Accounting viene reindirizzato sulla sua pagina riservata | |
| **Exception/Alternative flow** | **5.2.18.1 errorLogin:**  * 1. L’ eccezione inizia nel punto 3.1 nel caso in cui le credenziali inserite dal Gestore Accounting sia sbagliate.   Il sistema indirizza il Gestore Accounting  Nuovamente al form per il reinserimento delle credenziali, l’accesso non verrà effettuato finché i dati inseriti non saranno corretti. | |
|  |  | |

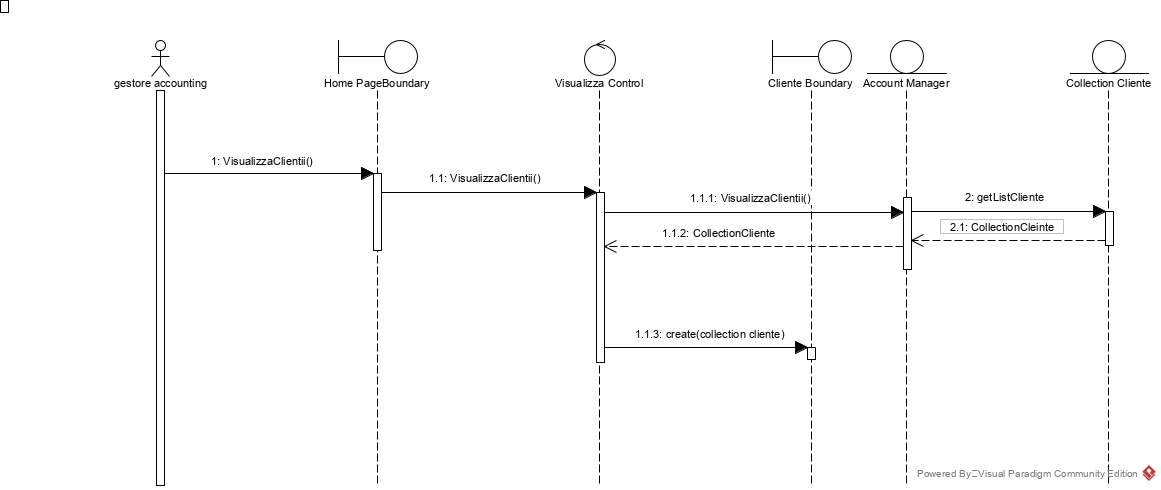




### 

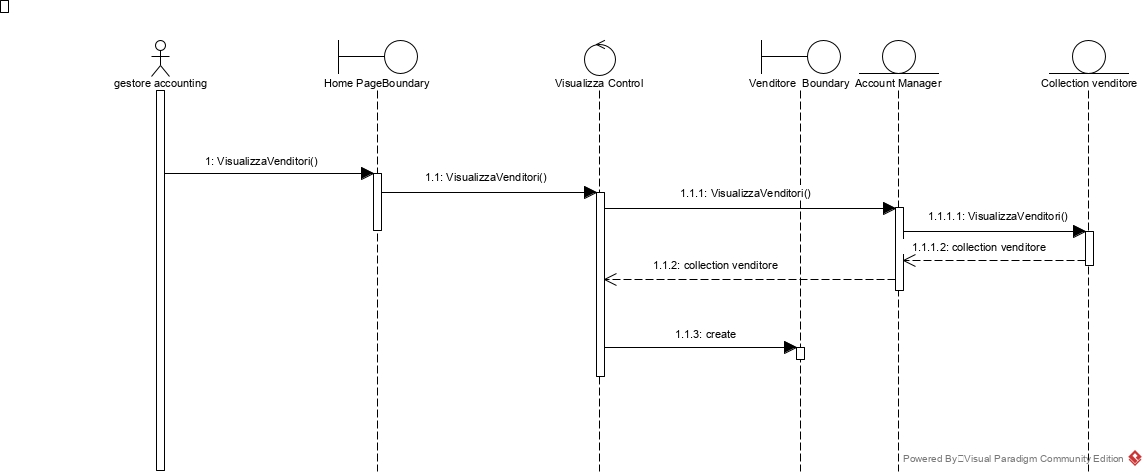
5.2.23) Visualizza clienti

|  |  |  |
| --- | --- | --- |
| **Use case name** | Visualizza clienti | |
| **Participating**  **Actors** | Gestore Accounting | |
| **Entry Condition** | Il Gestore Accounting possiede l'account, si trova nella home page | |
| **Flow of Events** | Actors Steps | System Steps |
|  | 1.Il Gestore Accounting dopo aver effettuato l'accesso seleziona il link alla pagina dedicata ”Visualizza clienti”. | 2.Il sistema preleva i dati di tutti i clienti e reindirizza il Gestore Accounting alla schermata dedicata. |
| **Exit Condition** | Il caso d' uso termina quando il Gestore Accounting visualizza la lista dei clienti registrati. | |



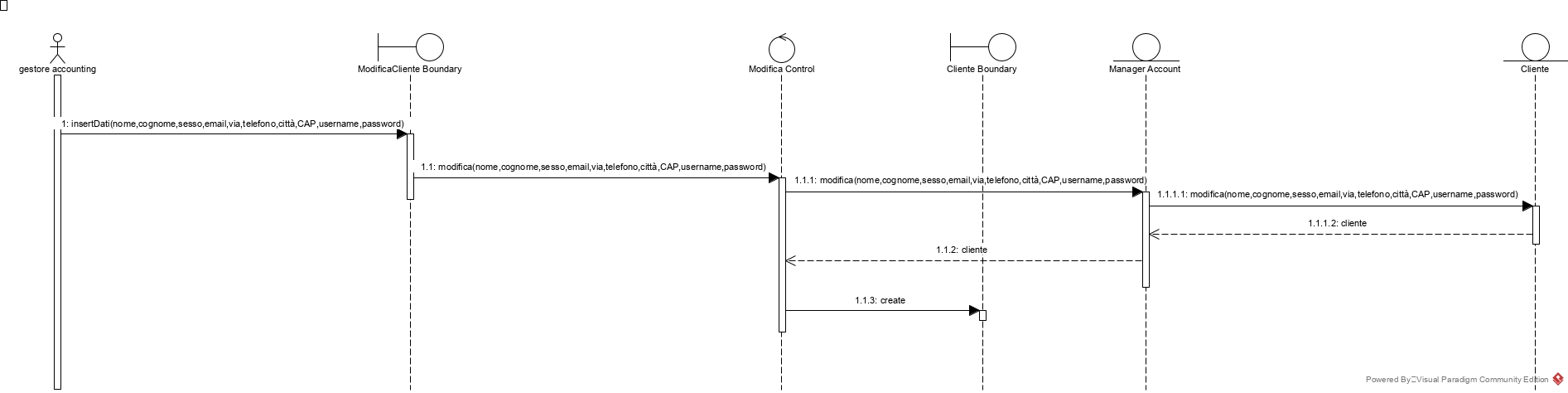
5.2.24) Visualizza venditori

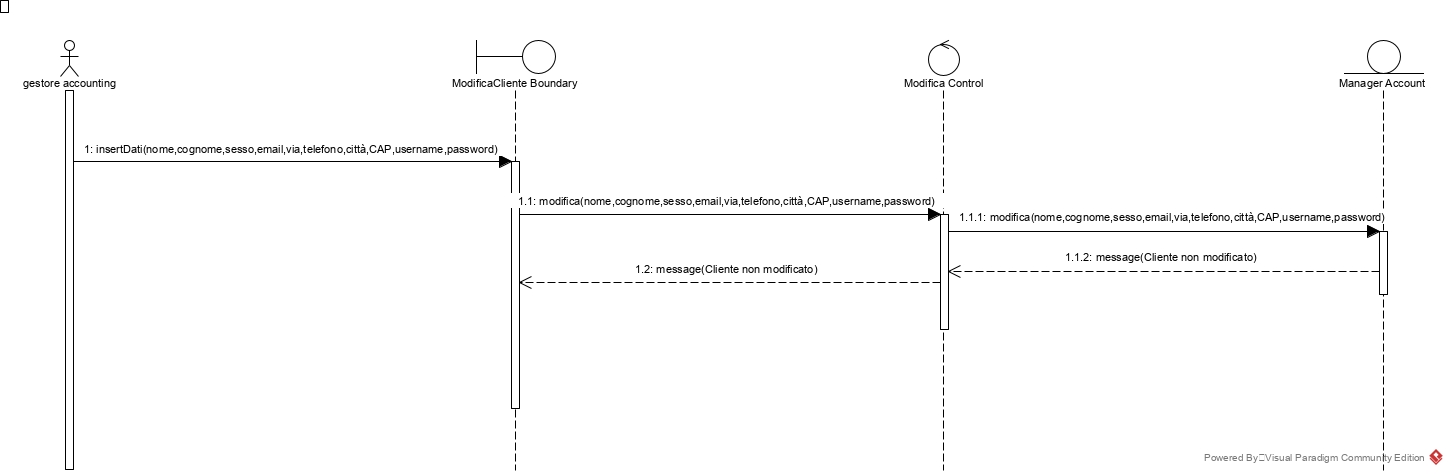
|  |  |  |
| --- | --- | --- |
| **Use case name** | Visualizza venditori | |
| **Participating**  **Actors** | Gestore Accounting | |
| **Entry Condition** | Il Gestore Accounting possiede l'account, si trova nella home page | |
| **Flow of Events** | Actors Steps | System Steps |
|  | 1.Il Gestore Accounting dopo aver effettuato l'accesso, seleziona il link alla pagina dedicata, ”Visualizza Venditori”.  3. Il Gestore Accounting tramite il bottone “Visualizza” visualizza i clienti registrati. | 2.Il sistema reindirizza il Gestore Accounting alla schermata dedicata. |
| **Exit Condition** | Il caso d' uso termina quando il Gestore Accounting visualizza i clienti registrati. | |



5.2.25) Modifica cliente

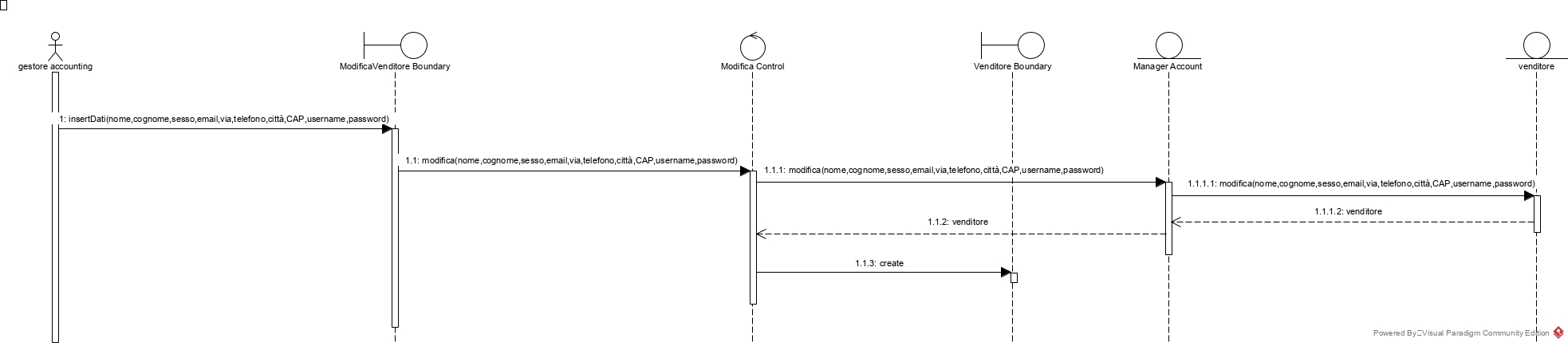
|  |  |  |
| --- | --- | --- |
| **Use case name** | Modifica cliente | |
| **Participating**  **Actors** | Gestore Accounting | |
| **Entry Condition** | Il Gestore Accounting possiede l' account,si trova nella home page | |
| **Flow of Events** | Actors Steps | System Steps |
|  | 1. Il Gestore Accounting dopo aver effettuato l’accesso tramite l' area dedicata ,”Modifica Cliente”, intende modificare dei dati.  3.Il Gestore Accounting può modificare alcuni dati come “Nome”, “Cognome”, “sesso”, “Email”, “Telefono”, “Via”, “Città”, “CAP”, “Username”, “Password” . Le informazioni inserite vengono inviate al sistema.  4.Il Gestore Accounting visualizza le modifiche. | 2.Il sistema reindirizza il Gestore Accounting in un form per modificare le informazioni.  3.1Il sistema registra tutte le modifiche effettuate. |
| **Exit Condition** | Il caso d' uso termina quando le modifiche inserite dal Gestore Accounting saranno rese persistenti. | |
| **Exception/Alternative flow** | **5.2.21.1 checkCampi:**  * 1. L’ eccezione inizia nel punto 3 nel caso in cui caso l’aggiunta di uno o più campi non rispetti alcune caratteristiche.   2. Il sistema evidenza gli errori è non fa procedere il Gestore Accounting, fino a quando non si inseriscono i dati corretti all’interno dei moduli. | |





5.2.26) Modifica venditore

|  |  |  |
| --- | --- | --- |
| **Use case name** | Modifica venditore | |
| **Participating**  **Actors** | Gestore Accounting | |
| **Entry Condition** | Il Gestore Accounting possiede l'account, si trova nella home page | |
| **Flow of Events** | Actors Steps | System Steps |
|  | 1. Il Gestore Accounting dopo aver effettuato l' accesso, tramite l’ area, “Modifica Cliente”, può modificare i dati.  3.Il Gestore Accounting può modificare alcuni campi personali del venditore “Nome”, “Cognome”, “sesso”, “E-mail”, “Telefono”, “Via”, “Città”, “CAP”, “Username”, “Password” ed inoltre può modificare i dati del negozio del venditore come “Nome Shop”, “Colore”, “Design menù”, “Via”, “Città”, “CAP”, “Partita IVA”, “Breve Descrizione “.  Le informazioni inserite vengono inviate al sistema.  4.Il Gestore Accounting visualizza le modifiche. | 2.Il sistema reindirizza il Gestore Accounting in un form per modificare le informazioni.  3.1Il sistema registra tutte le modifiche effettuate. |
| **Exit Condition** | Il caso d' uso termina quando le modifiche inserite dal Gestore Accounting saranno rese persistenti. | |
| **Exception/Alternative flow** | **5.2.22.1 checkCampi:** 1.1 L’ eccezione inizia nel punto 3 nel caso in cui caso l’aggiunta di uno o più campi non rispetti alcune caratteristiche.  1.2 Il sistema evidenza gli errori è non fa procedere il Gestore Accounting, fino a quando non si inseriscono i dati corretti all’interno dei moduli. | |
|  |  | |
| **Formato dati** | [link](#_Iscrizione_Venditore) | |





5.2.27) Cancellazione cliente

|  |  |  |
| --- | --- | --- |
| **Use case name** | Cancellazione cliente | |
| **Participating**  **Actors** | Gestore Accounting | |
| **Entry Condition** | Il Gestore Accounting possiede l’account, si trova nella home page | |
| **Flow of Events** | Actors Steps | System Steps |
|  | 1. Il Gestore Accounting dopo aver effettuato l’accesso, tramite il link dedicato ”Clienti ”può eliminare il cliente , tramite l’ interfaccia.  3.Il Gestore Accounting elimina il cliente. | 2 Il sistema indirizza al modulo dedicato.  3.1 Il sistema registra la modifica. |
| **Exit Condition** | Il caso d' uso termina quando la cancellazione del cliente viene resa persistente. | |
| **Exception/Alternative flow** | **5.2.23.1 annullaEliminazione:** 1.1 L’ eccezione inizia nel punto 3 nel caso in cui caso il Gestore Accounting sbagli la selezione del cliente per l’eliminazione.  1.2 Il sistema prima di procedere con la rimozione chiede al Gestore Accounting, la conferma eliminazione del cliente selezionato, tramite un apposito pop-up. | |



Immagine che contiene oggetto

Descrizione generata automaticamente

5.2.28) Cancellazione venditore

|  |  |  |
| --- | --- | --- |
| **Use case name** | Cancellazione venditore | |
| **Participating**  **Actors** | Gestore Accounting | |
| **Entry Condition** | Gestore Accounting possiede l’account, si trova nella home page | |
| **Flow of Events** | Actors Steps | System Steps |
|  | 1. Il Gestore Accounting dopo aver effettuato l’accesso, tramite il link dedicato ”Venditori” può eliminare il venditore , tramite l’ interfaccia.  3.Il Gestore Accounting elimina il venditore. | 2 Il sistema indirizza il Gestore Accounting al modulo dedicato.  3.1 Il sistema registra la modifica. |
| **Exit Condition** | Il caso d' uso termina quando la cancellazione del venditore viene resa persistente. | |
| **Exception/Alternative flow** | **5.2.24.1 annullaEliminazione:** 1.1 L’ eccezione inizia nel punto 3 nel caso in cui caso il Gestore Accounting sbagli la selezione del venditore per l’eliminazione.  1.2 Il sistema prima di procedere con la rimozione chiede il Gestore Accounting, la conferma eliminazione del cliente selezionato, tramite un apposito pop-up. | |

Immagine che contiene oggetto

Descrizione generata automaticamente

Immagine che contiene oggetto

Descrizione generata automaticamente

## **5.3) Formato dati:**

#### **Iscrizione Venditore/Modifica Venditore**

|  |  |
| --- | --- |
| **Campi** | **Formato** |
| Nome | Solo lettere. |
| Cognome | Solo lettere. |
| Sesso | RadioButton. |
| E-mail | Una sola @. |
| Telefono | 10 cifre senza spazi. |
| Via | Lettere e cifre senza limiti. |
| Città | Lettere e cifre senza limiti. |
| CAP | 5 cifre. |
| Username | Lettere e numeri, minimo 3, massimo 16. |
| Password | Almeno una lettera maiuscola e minuscola, un numero, un carattere speciale. Minimo 6, massimo 25. |

#### **Creazione del negozio/Modifica negozio**

|  |  |
| --- | --- |
| **Campi** | **Formato** |
| Nome Shop | Lettere e numeri. Minimo 3, massimo 16. |
| Colore | RadioButton. |
| Design menù | RadioButton. |
| Via | Lettere e cifre senza limiti. |
| Città | Lettere e cifre senza limiti. |
| Cap | 5 cifre. |
| Partita IVA | 11 cifre. |
| Breve Descrizione | Minimo 10 caratteri, massimo 500. |

#### **Inserisci Categoria/Modifica categoria**

|  |  |
| --- | --- |
| **Campi** | **Formato** |
| Categoria | Lettere e numeri. |
| Descrizione | Minimo 10, massimo 500 caratteri. |

#### **Inserisci Prodotto/Modifica Prodotto**

|  |  |
| --- | --- |
| **Campi** | **Formato** |
| Nome Prodotto | Campo illimitato, sia lettere che numeri. |

|  |  |
| --- | --- |
| **CAMPI** | **FORMATO** |
| Nome | Solo lettere |
| Cognome | Solo lettere |
| Sesso | Radio Button |
| E-mail | Una sola @ |
| Telefono | 10 cifre solo numeri senza spazi |
| Via | Cifre e lettere illimitate |
| Città | Cifre e lettere illimitate |
| Cap | Solo 5 numeri senza spazi |
| Username | Lettere e numeri, Min 3 Max 16 |
| Password | Almeno una lettera maiuscola, minuscola, numero e carattere speciale, Min 6 Max 25 |

# **5.4.1 Object Model**

In questa sezione descriviamo il modello a oggetti. Innanzitutto focalizziamo le classi correlate e forniamo i diagrammi delle classi UML per illustrare le loro relazioni.

Immagine che contiene testo, mappa

Descrizione generata automaticamente

## **5.4.1.1 Categoria**

La classe categoria è composta da un insieme di prodotti. La classe è definita dagli attributi nomeNegozio, nomeCategoria, descrizione, path.

L’attributo nomeNegozio rappresenta il nome del negozio a cui appartiene la categoria.

L’attributo nomeCategoria rappresenta il nome della categoria.

L’attributo descrizione fornisce brevi informazioni sulla categoria inserita.

L’attributo path rappresenta l’immagine fornita alla categoria.

Il metodo addCategoria serve ad aggiungere una nuova categoria.

Il metodo modifica serve per modificare i campi di una categoria.

## **5.4.1.2 Prodotto**

La classe prodotto è composta dagli attributi idProdotto, nomeNegozio, nomeCategoria, iva, path, prezzo, nome, qta, sconto, descrizione.

Tutti questi attributi formano una categoria.

L’attributo idProdotto indica il codice che rappresenta un determinato prodotto.

L’attributo nomeNegozio rappresenta il nome del negozio a cui appartiene il prodotto.

L’attributo nomeCategoria rappresenta il nome della categoria a cui appartiene il negozio.

L’attributo iva rappresenta un’imposta sul valore inserito.

L’attributo path rappresenta l’immagine fornita al prodotto.

L’attributo prezzo indica il prezzo attribuito al prodotto.

L’attributo nome indica il nome del prodotto.

L’attributo qta indica la quantità disponibile di uno specifico prodotto.

L’attributo sconto indica la percentuale di riduzione del prezzo di vendita di un prodotto.

L’attributo descrizione fornisce brevi informazioni sul prodotto inserito.

Il metodo addProdotto serve per inserire un nuovo prodotto.

Il metodo modifica serve per modificare i campi di un determinato prodotto.

Il metodo delete serve per eliminare un determinato prodotto.

## **5.4.1.3 Carrello**

La classe carrello è un’aggregazione di prodotti. Questa classe è formata dagli attributi listaProdotti, quantità, usernameCliente.

L’attributo listaProdotti rappresenta un insieme di prodotti inseriti nel carrello.

L’attributo quantità indica la quantità di prodotti inseriti nel carrello.

Nel carrello possono esserci uno o più prodotti.

L’attributo usernameCliente

## **5.4.1.4 Negozio**

La classe negozio è composta da una o più categorie. La classe è formata dagli attributi nome, usernameVenditore, design, colore, Piva, descrizione, dataIscrizione, via, città, cap, logo.

L’attributo nome indica il nome del negozio.

L’attributo usernameVenditore indica il nome del venditore.

L’attributo design memorizza il layout del menù (left / top).

L’attributo colore indica il colore dei pulsanti (giallo – verde – blu )

L’attributo Piva rappresenta una sequenza di 11 cifre che identifica univocamente un soggetto.

L’attributo descrizione fornisce brevi informazioni sul negozio inserito.

L’attributo dataIscrizione indica la data di iscrizione del negozio.

L’attributo via rappresenta la strada dove si trova il negozio.

L’attributo città indica il luogo in cui è situato il negozio.

L’attributo cap (codice di avviamento postale) è usato per facilitare l’individuazione del luogo del negozio.

L’attributo logo indica il logo rappresentante il negozio.

La classe Negozio possiede anche un metodo creaNegozio per creare e aggiungere un nuovo negozio.

## **5.4.1.5 Venditore**

La classe venditore rappresenta colui che possiede un negozio. Un venditore possiede un solo negozio. La classe venditore è formata dagli attributi nome, cognome, sesso, via, città, cap.

L’attributo nome indica il nome del venditore.

L’attributo cognome indica il cognome del venditore.

L’attributo via indica la strada dove abita il venditore.

L’attributo città indica il luogo in cui è situato il venditore.

L’attributo cap (codice di avviamento postale) è usato per facilitare l’individuazione del luogo del venditore.

Dopo aver compilato i campi, attraverso il metodo registrazione, viene creato l’oggetto venditore.

## **5.4.1.6 Utente**

L’utente viene specializzato in Venditore, Gestore Accounting o Cliente.

La classe è formata dagli attributi username e password.

L’attributo username rappresenta il nome dell’utente.

L’attributo password rappresenta la chiave d’accesso dell’utente.

La classe utente possiede anche i metodi login e logout.

Il metodo login si usa per effettuare l’accesso.

Il metodo logout si usa per scollegarsi dal sistema.

## **5.4.1.7 Gestore Accounting**

La classe Gestore Accounting è formata dagli attributi nome, cognome e sesso.

L’attributo nome indica il nome al Gestore Accounting.

L’attributo cognome indica il cognome del Gestore Accounting.

L’attributo sesso indica il sesso.

## **5.4.1.8 Cliente**

La classe cliente è formata dagli attributi nome, cognome, sesso, via, città, cap.

L’attributo nome indica il nome del cliente.

L’attributo cognome indica il cognome del cliente.

L’attributo sesso indica il sesso del cliente.

L’attributo via indica la strada dove abita il cliente.

L’attributo città indica il luogo in cui è situato il cliente.

L’attributo cap (codice di avviamento postale) è usato per facilitare l’individuazione del luogo del cliente.

Dopo aver compilato i campi, attraverso il metodo registrazione, viene creato l’oggetto cliente.

Il cliente inoltre, possedere un solo carrello.

## **5.4.1.9 Fattura**

La classe fattura è formata dagli attributi numeroFattura, username\_cliente, datOrdine, viaDestinazione, cittàDestinazione, capDestinazione.

Possiamo avere una o più fatture. Inoltre, una o più fatture possono essere emesse ad un negozio.

L’attributo numeroFattura indica il numero della fattura (differisce in base agli ordini).

L’attributo username\_cliente memorizza il nome utente del cliente.

L’attributo dataOrdine indica la data in cui è stato effettuato l’ordine.

L’attributo viaDestinazione rappresenta la strada dove sarà spedito l’ordine.

L’attributo cittàDestinazione rappresenta il luogo dove sarà spedito l’ordine.

L’attributo capDestinsazione indica il codice di avviamento postale per facilitare l’individuazione del luogo dove sarà spedito l’ordine.

## **5.4.1.10 Riferimento**

La classe riferimento è una classe associativa in quanto, una volta generata la fattura, i dati devono essere memorizzati perché alcuni attributi(qtàOrdinata, prezzoUnitario, iva, sconto) devono essere resi persistenti al momento dell’acquisto.

L’attributo qtàOrdinata indica il numero di prodotti ordinati.

L’attributo prezzoUnitario indica il prezzo del singolo prodotto.

L’attributo iva rappresenta un’imposta sul valore inserito.

L’attributo sconto indica la percentuale di riduzione del prezzo di vendita di un prodotto.

## **5.4.1.11 Gestore spedizioni**

La classe si riferisce ad una collezione di corrieri inseriti manualmente nel DBMS dal amministratore del sistema. Questi corrieri sono accessibili al venditore soltanto.

# **5.5 Dynamic model**

Immagine che contiene screenshot

Descrizione generata automaticamente

**Rappresentazione degli stati di un carrello.**

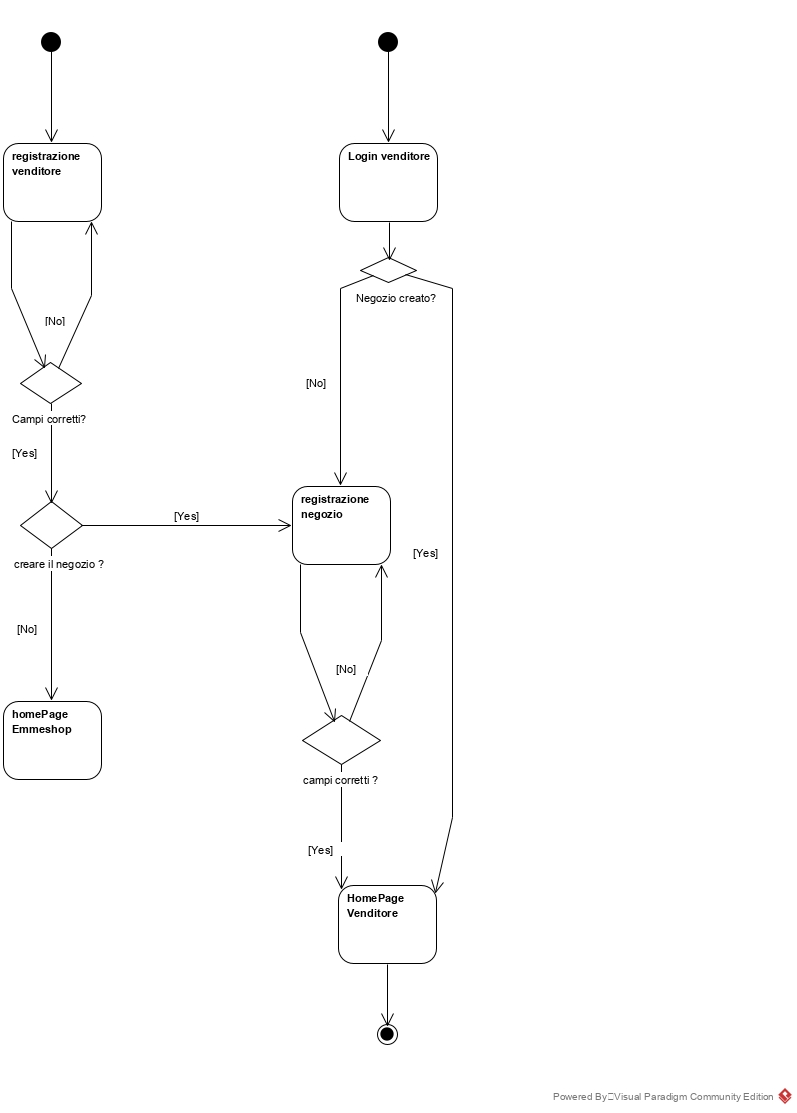
Un cliente dopo essersi loggato può visualizzare il proprio carrello, che si trova nello stato vuoto, durante la sua permanenza sulla piattaforma, può aggiungere prodotti al carrello, e si passa dallo stato vuoto a pronto per acquistare. Il cliente può continuare ad aggiungere prodotti, ma anche rimuoverli dal carrello tramite l’interfaccia. Quando il cliente visiona il carrello può modificare quantità dei prodotti, prima di effettuare l’ordine. Dopo aver effettuato l’ordine il carrello si svuota e ritorna allo stato originale.

Immagine che contiene mappa, screenshot

Descrizione generata automaticamente

**Rappresentazione degli stati dalla conferma alla ricezione dell’ordine.**

Arriva ordine da un cliente con il controllo che ci sia almeno un prodotto nell’ordine effettuato. Dopo sono richiesti i dati di pagamento, con controlli appropriati. In caso di errore ordine verrà annullato. Un esempio di errore è dato della carta falsa, credito non disponibile, o prodotti ordinati non disponibili (con relativo avviso da parte del Negozio). In caso di riuscita invece l’ordine verrà preso incarico, stato di lavorazione, il venditore dovrà scegliere un servizio di spedizione (corriere), a cui affida l’ordine. Il corriere (può essere anche lo stesso venditore se dispone di questo servizio) prepara la spedizione ed invia l’ordine, stato ordine spedito. Si termina quando ordine verrà consegnato.



**Activity diagram registrazione negozio**

Dopo aver effettuato attività di registrazione del venditore, abbiamo il controllo dei campi inseriti. Se la condizione è verificata al venditore verrà chiesto di creare il proprio negozio, o crearlo al prossimo accesso dopo aver effettuato il login. Verranno controllati i campi inseriti per il negozio, se questi soddisfano i controlli il venditore verrà ridiretto nella home page dove potrà iniziare ad inserire categorie e prodotti.

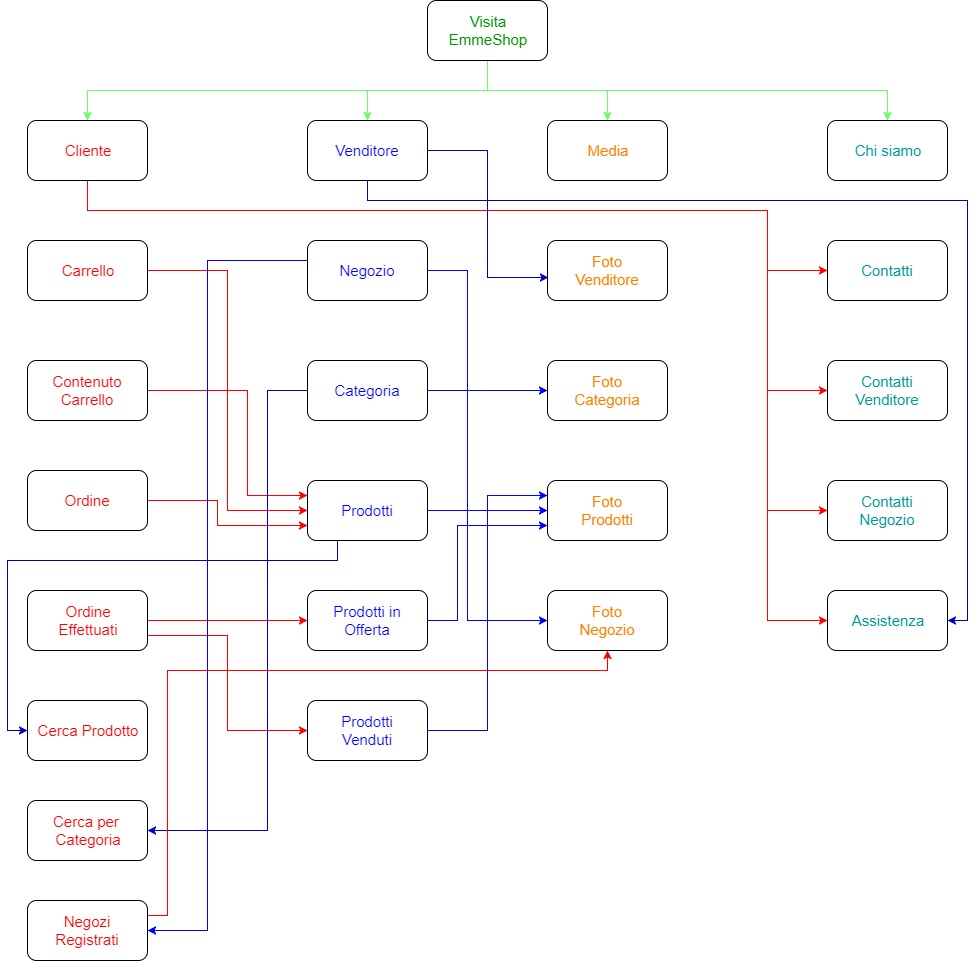
# **5.6 User interface, navigatioinal paths and screen mock-ups**

## **5.6.1 Navigational paths**

Immagine che contiene screenshot

Descrizione generata automaticamente

## **5.6.2 Content map**



## **5.6.3 User Interface and Screen mock-ups**

Visione desktop

## Immagine che contiene screenshot Descrizione generata automaticamente

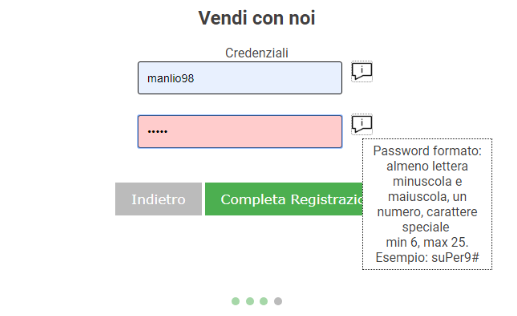
Visione mobile

**Immagine che contiene screenshot

Descrizione generata automaticamente**

Immagine che contiene screenshot

Descrizione generata automaticamente



Visione del Cliente registrato





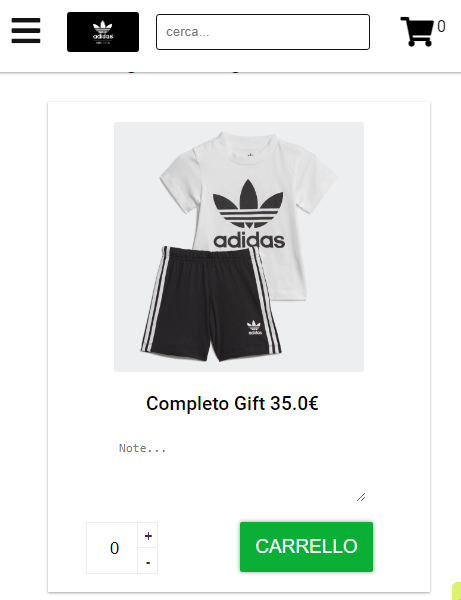
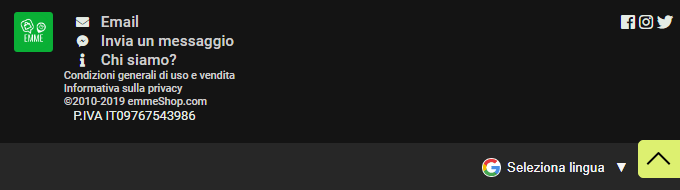


Immagine che contiene screenshot

Descrizione generata automaticamente



****

# **6)Glossario**

|  |  |
| --- | --- |
| ***Termine*** | **Definizione** |
| *Piattaforma* | La piattaforma indica una base software e/o hardware su cui sono sviluppate e/o eseguite applicazioni. |
| Gestore Accounting | Gestore dell’intera piattaforma. Gestisce gli utenti iscritti, sia Clienti che Venditori. Ha delle funzionalità specifiche per ciascuno di essi. |
| *Cliente* | Figura che cerca i prodotti da acquistare. |
| *Venditore* | Figura che possiede un negozio iscritto alla piattaforma. Gestisce il proprio catalogo di categorie e prodotti presenti nel negozio. |
| *Gestore delle spedizioni* | L’azienda che fornisce il servizio di spedizione e recapito. |
| *Iscrizione* | Inserimento di alcuni dati relativi ad una determinata figura (Cliente/Venditore) |
| *Login* | Procedura di accesso a un sistema informatico, che prevede l’inserimento di un codice identificativo e di una parola d’ordine da parte dell’utente. |
| *Form* | Interfaccia di un programma che consente a un utente di un sito web di inserire e inviare uno o più dati. |
| *Sistema* | Entità integrata costituita da diverse strutture o elementi interagenti; può indicare: il tipo fondamentale di software di base impiegato in un elaboratore (per es. il *s. MS-DOS*); il complesso del software e dello hardware di un computer, posto a confronto con un altro; un gruppo di computer interconnessi e in grado di compiere un'elaborazione unitaria. |
| *Categoria* | Suddivisione che si ottiene ordinando o classificando gli elementi secondo vari criteri. |
| *Area Riservata* | Parte di un sito web composta da una o più pagine, alla quale possono accedere solo utenti con determinate caratteristiche: solitamente si tratta degli utenti che hanno effettuato la registrazione al sito, inserendo i dati richiesti  Dal Gestore Accounting. |
| *CheckQuantità* | Controllo quantità del prodotto disponibile. |
| *CheckOrdine* | Controllo effettuato su un intero ordine (verifica della quantità, disponibilità). |