**Università Politecnica delle Marche**

**Dipartimento di Ingegneria dell’Informazione**

**Corso di Laurea Triennale in Ingegneria Informatica e dell’Automazione**

**Software per la gestione di un negozio dell’usato**

**Ursu Leon**

**Perazzoli Leonardo**

Indice

[1 INTRODUZIONE 2](#_Toc125910401)

[1.1 DESCRIZIONE IN LINGUAGGIO NATUREALE 2](#_Toc125910402)

[1.2 GLOSSARIO DEI TERMINI 3](#_Toc125910403)

[2 ANALISI DEI REQUISITI 3](#_Toc125910404)

[2.1 REQUISITI DEL SISTEMA 4](#_Toc125910405)

[2.1.1 REQUISITI FUNZIONALI 5](#_Toc125910406)

[2.1.2 REQUISITI NON FUNZIONALI 9](#_Toc125910407)

[2.2 DIAGRAMMI DEI CASI D’USO 9](#_Toc125910408)

[2.2.1 ATTORI 10](#_Toc125910409)

[2.2.2 GESTIONE AMMINISTRATORE 10](#_Toc125910410)

[2.2.3 GESTIONE CLIENTE-PROPRIETARIO 20](#_Toc125910411)

[2.2.4 GESTIONE SISTEMA 22](#_Toc125910412)

[2.2.5 GESTIONE USER 25](#_Toc125910413)

[2.4 MATRICE MAPPING DEI REQUISITI 27](#_Toc125910414)

[3 DIAGRAMMI DI ANALISI 27](#_Toc125910415)

[3.1 PACKAGE DI ANALISI 28](#_Toc125910416)

[3.2 DIAGRAMMI DELLE CLASSI DI ANALISI 29](#_Toc125910417)

[3.3 DIAGRAMMI DI SEQUENZA 30](#_Toc125910418)

[3.3.1 BACKUP 30](#_Toc125910419)

[3.3.2 CONTROLLASTATOPRODOTTO 31](#_Toc125910420)

[3.3.3 CRUD PRODOTTO 32](#_Toc125910421)

[3.3.4 LOGIN 37](#_Toc125910422)

[3.3.5 NOTIFICA 38](#_Toc125910423)

[3.3.6 RICERCACLIENTE 39](#_Toc125910424)

[3.3.7 RICERCAPRODOTTO 40](#_Toc125910425)

[3.3.8 VENDIPRODOTTO 41](#_Toc125910426)

[3.3.9 VISUALIZZASTATISTICHE 43](#_Toc125910427)

[4 DIAGRAMMI DI PROGETTAZIONE 44](#_Toc125910428)

[4.1 DIAGRAMMI DELLE CLASSI DI PROGETTAZIONE 44](#_Toc125910429)

[4.2 DIAGRAMMI DEI COMPONENTI 48](#_Toc125910430)

[4.3 DIAGRAMMI DELLE MACCHINE A STATI 49](#_Toc125910431)

[4.3.1 MACCHINE A STATI: BACKUP 49](#_Toc125910432)

[4.3.2 MACCHINE A STATI: CATEGORIA 51](#_Toc125910433)

[4.3.3 MACCHINE A STATI: LOGIN 52](#_Toc125910434)

[4.3.4 MACCHINE A STATI: PRODOTTO 53](#_Toc125910435)

[4.3.5 MACCHINE A STATI: RICEVUTA 54](#_Toc125910436)

[5 IMPLEMENTAZIONE 55](#_Toc125910437)

[5.1 DIAGRAMMA DI DEPLOYMENT 55](#_Toc125910438)

[6 MOCKUP 55](#_Toc125910439)

[6.1 VISTA LOGIN 56](#_Toc125910440)

[6.2 VISTA USER 56](#_Toc125910441)

[6.3 VISTA AMMNISTRATORE 57](#_Toc125910442)

[6.4 VISTA CLIENTE PROPRIETARIO 64](#_Toc125910443)

[7 PYUNIT 66](#_Toc125910444)

# 

# 1 INTRODUZIONE

## DESCRIZIONE IN LINGUAGGIO NATUREALE

DESCRIZIONE NEGOZIO

Il progetto proposto consiste nella realizzazione di un sistema informativo per la gestione di un negozio dell'usato.

L’immobile è composto da una sala di esposizione e un magazzino, nella sala di esposizione gli oggetti vengono esposti sugli scaffali, ogni oggetto ha una determinata categoria, un id oggetto, un prezzo e una data di esposizione. L’unico lavoratore presente nel negozio è colui che lo gestisce interamente.

REGISTRAZIONE UTENTE

Il cliente-proprietario arrivato al negozio, se non lo ha già fatto in precedenza, dovrà registrarsi insieme all’amministratore fornendo i propri dati personali, la propria e-mail e la password concordata con l’amministratore, gli verrà fornito un codice cliente. Il sistema invierà un’e-mail di registrazione effettuata con successo, dove verranno indicati i rispettivi dati di registrazione per l’accesso locale al sistema.

ACCETAZIONE OGGETTO

Solo se già registrato, il cliente-proprietario consegnerà l'oggetto destinato alla vendita, al cassiere(amministratore), il quale provvederà ad inserirlo nel sistema, compilando i dati dell'oggetto: nome categoria, id cliente-proprietario, data di registrazione e il prezzo iniziale che verrà concordato da ambo le parti (il cassiere e il cliente-proprietario).

L'oggetto rimarrà in vendita per un massimo di 5 mesi:  
- Al secondo mese dalla data di registrazione il prezzo iniziale del prodotto subirà un decremento del 30%.  
- Al terzo mese dalla data di registrazione il prezzo iniziale del prodotto subirà un decremento del 40%.  
- Al quarto mese dalla data di registrazione il prezzo iniziale del prodotto subirà un decremento del 50%.  
- Al quinto mese dalla data di registrazione il prodotto verrà eliminato dalla lista di oggetti disponibili.

Nel caso in cui l’oggetto giunga alla data di scadenza invenduto, l’oggetto deve essere ritirato dal cliente-proprietario altrimenti si procederà allo smaltimento.

VENDITA OGGETTO

Al momento della vendita l’amministratore effettuerà la ricerca dell’oggetto nel sistema effettua la vendita dell’oggetto con il prezzo di vendita. La data e ora di vendita viene acquisita automaticamente. L’oggetto venduto viene rimosso dalla lista degli oggetti disponibili e viene archiviato. Inoltre, verrà inoltrata un’e-mail per notificare la vendita del prodotto al corrispettivo cliente-proprietario.

ALTRE FUNZIONALITA’

Nella struttura vi è un terminale accessibile ai visitatori del negozio

Il cliente-acquirente, non sarà vincolato dalla registrazione, potrà visualizzare l'intera lista di prodotti disponibili ed avrà a disposizione strumenti per effettuare filtraggio in base alle proprie preferenze (categoria, prezzo, data).

L’amministratore avrà a disposizione degli strumenti per visualizzare delle statistiche sui prodotti venduti e sul numero di clienti avendo modo così di analizzare l’andamento della propria attività commerciale. Inoltre, avrà anche la possibilità di effettuare un backup manualmente.

Alla fine della giornata lavorativa il sistema dovrà essere chiuso per effettuare la creazione delle statistiche e un backup dei dati, e che consisterà nel copiare tutti i dati che  
costituiscono il database, sul disco, al fine di riparare a eventuali perdite di dati o anomalie.

## GLOSSARIO DEI TERMINI

In seguito, si trova il glossario del progetto in questione. Tutti i termini presenti nella colonna denominata “PAROLA” sono stati estratti dalla “1.1 Descrizione in Linguaggio naturale”. Tali termini sono stati definiti per ridurre l’ambiguità generale nella comprensione con i corrispettivi, se presenti, sinonimi e omonimi.

|  |  |  |  |
| --- | --- | --- | --- |
| **PAROLA** | **DEFINIZIONE** | **SINONIMI** | **OMONIMI** |
| Cliente-Proprietario |  | ClienteP | nessuno |
| Amministratore |  | Admin | nessuno |
| User |  | Utente | nessuno |
| Codice cliente |  | Id Cliente | nessuno |
| Dati di registrazione |  | Credenziali | nessuno |
| Prezzo iniziale |  | Prezzo originale | nessuno |
| Prezzo di vendita |  | Prezzo corrente | nessuno |
| Categoria |  | nessuno | nessuno |
| Account |  | nessuno | nessuno |
| Prodotto |  | oggetto | nessuno |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# ANALISI DEI REQUISITI

Dall’analisi della documentazione prodotta durante le interviste con gli stakeholder, sono stati individuati dei requisiti che fungeranno come vincoli da rispettare durante la progettazione del sistema. Essi sono suddivisi in *requisiti funzionali* e *requisiti non funzionali.*

* *requisiti funzionali:* sono delle caratteristiche che il sistema deve garantire all’utilizzatore.
* *requisiti non funzionali:* sono dei vincoli imposti dal sistema.

Dall’analisi dei requisiti del sistema e alla descrizione nel linguaggio naturale è stato possibile individuare i casi d’uso del sistema.

## REQUISITI DEL SISTEMA

In questa sezione verranno trattati i requisiti del sistema, come da figura numero 1, i requisiti sono stati suddivisi in package diversi.

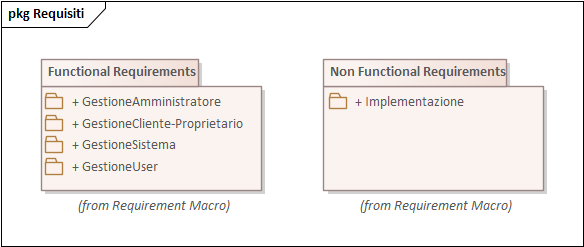


Figura - Package dei requisiti

### REQUISITI FUNZIONALI

Nella figura numero 2 troviamo la rappresentazione grafica di come abbiamo suddiviso i *requisiti funzionali.* La suddivisione in package diversi permette di individuare aree diverse del nostro sistema

Immagine che contiene testo

Descrizione generata automaticamente

Figura - Requisiti funzionali

**RF1 –** Genera Statistiche

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

*Il sistema dovrà effettuare un backup dei dati*

**RF2 -** Backup

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà gestire la scadenza dei prodotti non venduti

**RF3 –** Scadenza Prodotto

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà gestire la scadenza dei prodotti non venduti

**RF4 -** Notifica Registrazione Cliente

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà notificare al cliente la propria registrazione

**RF5 -** Notifica Vendita Prodotto

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà notificare al cliente la vendita del proprio oggetto

**RF6 -** Vendita Prodotto

Type:**Requirement**

Status: proposed

Package: *GestioneAmministratore*

Il sistema dovrà notificare al cliente la vendita del proprio oggetto

**RF5 -** Notifica Prodotto Scaduto

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà notificare al cliente la scadenza dell’oggetto

**RF6 -** Vendita Prodotto

Type:**Requirement**

Status: proposed

Package: *GestioneAmministratore*

Il sistema dovrà permettere all’Amministratore di vendere un prodotto

**RF7 -** CRUD Prodotto

Type:**Requirement**

Status: proposed

Package: *GestioneAmministratore*

Il sistema dovrà essere in grado di effettuare operazioni CRUD sul prodotto

**RF8 -** CRUD Ricevuta

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà essere in grado di effetturare operazioni CRUD sulla ricevuta

**RF9 -** CRUD Cliente-Proprietario

Type:**Requirement**

Status: proposed

Package: *GestioneAmministratore*

Il sistema dovrà effettuare operazioni CRUD sul Cliente-Proprietario

**RF10 -** Visualizza Statistiche

Type:**Requirement**

Status: proposed

Package: *GestioneAmministratore*

Il sistema dovrà poter visualizzare statististiche sul prodotto

**RF11 -** Filtraggio Prodotto Per Data

Type:**Requirement**

Status: proposed

Package: *GestioneUser*

Il sistema dovrà effettuare il filtraggio dei prodotti con la data di inserimento

**RF12 -** Filtraggio Prodotto Per Prezzo

Type:**Requirement**

Status: proposed

Package: *GestioneUser*

Il sistema dovrà effettuare il filtraggio dei prodotti con il loro prezzo corrente

**RF13 -** Filtraggio Prodotto Per Categoria

Type:**Requirement**

Status: proposed

Package: *GestioneUser*

Il sistema dovrà effettuare il filtraggio dei prodotti con la loro categoria

**RF14 -** Login

Type:**Requirement**

Status: proposed

Package: *GestioneUser*

Il sistema dovrà permettere di effettuare il login

**RF15 -** Notifica Prodotto Scaduto

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà notificare al cliente che il suo prodotto è scaduto

**RF16 -** Statistiche Prodotti Venduti

Type:**Requirement**

Status: proposed

Package: *GestioneSistema*

Il sistema dovrà effettuare statistiche su prodotti venduti

**RF17 –** Visualizza Info Account

Type:**Requirement**

Status: proposed

Package: *GestioneCliente-proprietario*

Il sistema dovrà permettere di visualizzare le info del proprio account

**RF18 –** Visualizza Prodotti Propri

Type:**Requirement**

Status: proposed

Package: *GestioneCliente-proprietario*

Il sistema dovrà permettere di visualizzare i propri prodotti presenti nel sistema

**RF19 –** Backup Manuale

Type:**Requirement**

Status: proposed

Package: *GestioneAmministratore*

Il sistema dovrà permettere all’Amministratore di effettuare un backup manuale

**RF20 -** Visualizza Prodotto

Type:**Requirement**

Status:proposed

Package: *GestioneSistema*

Il sistema dovrà permettere di visualizzare e info di un prodotto

### REQUISITI NON FUNZIONALI

Nella figura numero 3 troviamo la rappresentazione grafica di come abbiamo suddiviso i *requisiti non funzionali*. La suddivisione in package diversi permette di individuare aree diverse del nostro sistema.

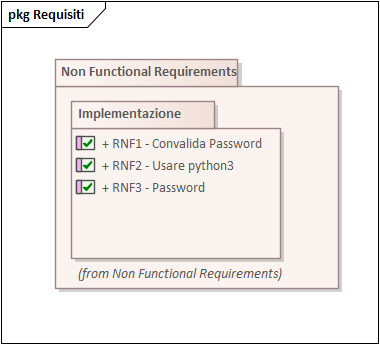


Figura - Requisiti non funzionali

**RNF1 -** Usare Python 3

Type:**Requirement**

Status:proposed

Package:Requisiti non funzionali

Il sistema dovrà essere implementato in Python 3

**RNF2 -** Password

Type:**Requirement**

Status:proposed

Package:Requisiti non funzionali

Il sistema dovrà poter gestire una password

**RNF3 -** Convalida Password

Type:**Requirement**

Status:proposed

Package:Requisiti non funzionali

Il sistema dovrà effettuare la convalida della password

## DIAGRAMMI DEI CASI D’USO

Con l’analisi dei requisiti è stato possibile ricavare i casi d’uso del sistema, scenari che si possono verificare durante l’esecuzione del sistema.

I seguenti diagrammi sono stati suddivisi in base alle varie aree di interazione da parte degli attori con il sistema in uso, in modo da rendere indipendenti, dove è possibile, le interazioni dei vari attori con il sistema.

### 2.2.1 ATTORI

Nella figura NUMERO vi è il diagramma degli attori individuati durante l’analisi, gli attori: Amministratore e Cliente-Proprietario sono delle generalizzazioni dell’attore User, ciò significa che potranno trovarsi nelle situazioni dell’attore User.

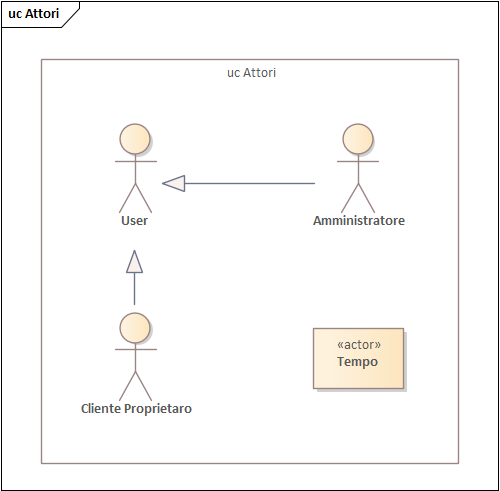


Figura - Diagramma degli attori

### 2.2.2 GESTIONE AMMINISTRATORE

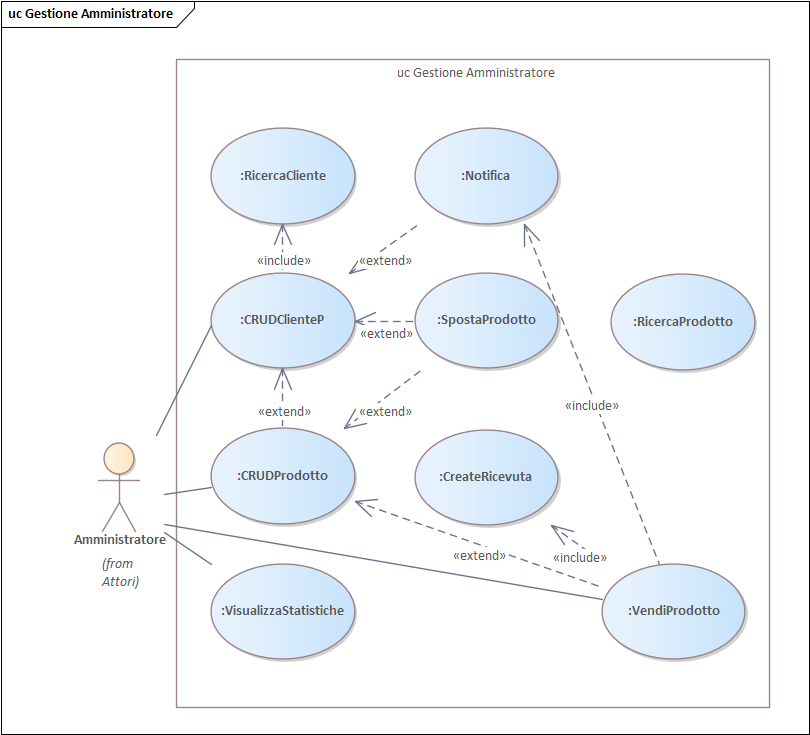


Figura – Diagramma dei casi d’uso: Gestione Amministratore

|  |
| --- |
| Caso d’uso: CreateRicevuta |
| ID: 1 |
| Descrizione: Questo caso d’uso permette di effettuare operazioni di creazione di una ricevuta |
| Attori primari: Amministratore |
| Attori secondari: nessuno |
| Precondizioni: l’Amministratore ha venduto dei prodotti |
| Sequenza eventi principali:  1. il caso d’uso inizia quando sono stati venduti dei prodotti  2. il sistema crea la ricevuta  3. il sistema aggiunge i prodotti nella ricevuta  4. il sistema stampa la ricevuta |
| Postcondizioni: il sistema si rende nuovamente disponibile per altre operazioni |
| Sequenza degli eventi alternativa: nessuna |

|  |
| --- |
| Caso d’uso: CRUDClienteProprietario |
| ID: 2 |
| Descrizione: Questo caso d’uso permette di effettuare operazioni CRUD sul ClienteProprietario |
| Attori primari: Amministratore |
| Attori secondari: ClienteProprietario |
| Precondizioni: l’amministratore del sistema desidera effettuare operazioni CRUD per un ClienteProprietario |
| Sequenza eventi principali:  1. il caso d’uso inizia quando l’attore primario desidera effettuare un  operazione CRUD di un ClienteProprietario  2. **if** l’Amministratore vuole inserire nel database un ClienteProprietario  2.1 **include** (:RicercaCliente)  2.2 **if** il cliente risulta esistente nel sistema  2.2.1 il sistema restituisce un errore  2.3 **else**  2.3.1 L’Amministratore inserisce i dati del ClientePr nel database  2.3.2 il sistema memorizza i dati del cliente nel database  2.3.3 il sistema manda una notifica  3. **else if** l’amministratore vuole visualizzare un ClienteProprietario  3.1 **include** (:RicercaCliente)  3.2 **if** il cliente risulta inesistente nel database  3.2.1 il sistema restituisce un errore  3.3 **else** il sistema mostra il cliente  4. **else if** l'amministratore vuole aggiornare i dati relativi ad un ClienteP  4.1. **include** (:RicercaCliente);  4.2. **if** il cliente risulta esistente nel database  4.2.1 l’amministratore specifica i nuovi dati  4.2.2. Il sistema aggiorna i dati del cliente nel database  4.3. **else**  3.3.1. il sistema restituisce un errore  5. **else if** l’amministratore vuole cancellare un ClienteP dal database  5.1 **include** (:RicercaCliente)  5.2 **if** il cliente risulta esistente nel database  5.2.1 **include** (:RicercaProdotto)  5.2.2 **while** (esistono prodotti del ClienteP)  5.2.2.1 il sistema elimina il prodotto  5.2.3 il sistema elimina il ClienteP dal database  5.3. **else**  5.3.1. il sistema restituisce un errore |
| Postcondizioni: il sistema si rende nuovamente disponibile per altre operazioni |
| Sequenza degli eventi alternativa: nessuna |

|  |
| --- |
| Caso d’uso: CRUDProdotto |
| ID: 3 |
| Descrizione: Questo caso d’uso permette di effettuare operazioni CRUD sul prodotto |
| Attori primari: Amministratore |
| Attori secondari: Nessuno |
| Precondizioni: l’Amministratore vuole effettuare un’operazione CRUD su un prodotto |
| Sequenza eventi principali:  1. **if** l’amministratore vuole creare un prodotto  1.1 il sistema chiede di inserire i dati del prodotto da creare  1.2 L’Amministratore inserisce i dati del prodotto nel sistema  1.3 il sistema memorizza i dati del prodotto nel database  1.4 **if** prodotto inserito correttamente  1.4.1 il sistema comunica il successo dell’operazione  1.5 **else**  1.5.1 il sistema restituisce un errore  2. **else if** l’Amministratore primario vuole visualizzare il prodotto  2.1 **include** (:RicercaProdotto)  2.2 **if** prodotto trovato nel database  2.2.1 il sistema restituisce l’oggetto trovato  2.3 else  2.3.1 il sistema restituisce un errore  3. **else if** l'Amministratore primario vuole aggiornare i dati relativi ad un prodotto  3.1. **include** (:RicercaProdotto);  3.2. **if** il prodotto risulta esistente nel database  3.2.1 l’attore primario specifica i nuovi dati  3.2.2. Il sistema aggiorna i dati del prodotto nel database  3.3. **else**  3.3.1. il sistema restituisce un errore  4. **else if** l’Amministratore vuole cancellare il prodotto dal database  4.1 **include** (:RicercaProdotto)  4.2 **if** il prodotto risulta esistente nel database  4.2.1 il sistema elimina il prodotto dal database  4.3. **else**  4.3.1. il sistema restituisce un errore |
| Postcondizioni: il sistema si rende nuovamente disponibile per altre operazioni |
| Sequenza degli eventi alternativa: nessuna |

|  |
| --- |
| Caso d’uso: Notifica |
| ID: 4 |
| Descrizione: Invia una notifica al cliente registrato |
| Attori primari: Amministratore |
| Attori secondari: Nessuno |
| Precondizioni Segmento 1: L’utente è stato creato  Precondizioni Segmento 2: Il prodotto è stato rimosso  Precondizioni Segmento 3: Il prodotto è stato venduto |
| Sequenza eventi Segmento 1:  1. il caso d’uso inizia quando l’amministratore crea un nuovo Cliente Proprietario  2. il sistema invia una e-mail al Cliente Proprietario  Sequenza eventi Segmento 2:  1. il caso d’uso inizia quando il Sistema rimuove il prodotto di un Cliente Proprietario  2. **if** il prodotto è stato venduto  2.1 il sistema invia una e-mail al cliente proprietario con l’esito della vendita  3.**else if** il tempo di vendita del prodotto è scaduto  3.1 il sistema invia una e-mail al cliente proprietario indicando la scadenza del prodotto registrato  Sequenza eventi Segmento 3:  1. Il caso d’uso inizia quando il prodotto viene venduto  2. il sistema invia un’e-mail al ClienteProprietario |
| Postcondizioni: il sistema si rende nuovamente disponibile per altre operazioni |
| Sequenza degli eventi alternativa: Nessuna |

|  |
| --- |
| Caso d’uso: RicercaCliente |
| ID: 5 |
| Descrizione: Questo caso d’uso permette di effettuare la ricerca di un cliente nel sistema |
| Attori primari: Amministratore |
| Attori secondari: Cliente |
| Precondizioni: l’attore primario desidera effettuare una ricerca del cliente nel database |
| Sequenza eventi principali:  1. il caso d’uso inizia quando l’attore primario desidera effettuare la ricerca di un Cliente nel database del sistema  2. l’attore primario specifica i dati di ricerca  3. il sistema effettua la ricerca con i dati forniti dall’attore primario  4. **if** il Cliente è stato trovato  4.1 il sistema restituisce i dati del Cliente  5. **else**  5.1 il sistema restituisce un errore |
| Postcondizioni:   1. il sistema ha restituito dei dati |
| Sequenza degli eventi alternativa: nessuna |

|  |
| --- |
| Caso d’uso: RicercaProdotto |
| ID: 6 |
| Descrizione: Questo caso d’uso permette di effettuare la ricerca di un prodotto nel sistema |
| Attori primari: Amministratore, ClienteP, User |
| Attori secondari: sistema |
| Precondizioni: l’attore primario desidera effettuare la ricerca di un prodotto |
| Sequenza eventi principali:  1. il caso d’uso inizia quando l’attore primario desidera effettuare la ricerca di un prodotto nel database del sistema  2. il sistema richiede all’attore primario di inserire i dati di ricerca  3. l’attore primario specifica i dati di ricerca  4. il sistema effettua la ricerca con i dati forniti dall’attore primario  5. **if** il prodotto è stato trovato  5.1 il sistema restituisce i dati del prodotto  6. **else**  6.1 il sistema restituisce un errore |
| Postcondizioni: nessuna |
| Sequenza degli eventi alternativa: nessuna |

|  |
| --- |
| Caso d’uso: SpostaProdotto |
| ID: 7 |
| Descrizione: Questo caso d’uso permette di effettuare lo spostamento di un prodotto nel sistema |
| Attori primari: Amministratore |
| Attori secondari: nessuno |
| Precondizioni: è necessario lo spostamento di un prodotto |
| Sequenza eventi principali:  1. Il caso d’uso inizia quando l’attore primario richiede di spostare un prodotto  2. Il sistema richiede all’attore primario un luogo di A e un luogo B  3. l’attore primario inserisce i dati richiesti  4. Il sistema richiede il prodotto o un suo riferimento da spostare dal luogo  A ad un luogo B  5. l’attore primario inserisce i dati richiesti  6. Il sistema sposta il prodotto dal luogo A ad un luogo B  7. **if** l’operazione è stata effettuata  7.1. il sistema comunica il successo dell’operazione  8. **else**  8.1 Il sistema restituisce un errore |
| Postcondizioni: Nessuna |
| Sequenza degli eventi alternativa: Nessuna |

|  |
| --- |
| Caso d’uso: VendiProdotto |
| ID: 8 |
| Descrizione: Questo caso d’uso permette di effettuare la vendita di un prodotto |
| Attori primari: Amministratore |
| Attori secondari: Nessuno |
| Precondizioni: l’amministratore desidera effettuare la vendita di un prodotto |
| Sequenza eventi principali:  1. il caso d’uso inizia quandol’Amministratore vuole vendere un prodotto  2. l’amministratore inserisce l’oggetto da vendere  3. il sistema inizia la ricerca del prodotto  3.1 **include** (:RicercaProdotto)  4. il sistema inizia a preparare la ricevuta di vendita  4.1 **include** (:CRicevuta)  5. il sistema elimina il prodotto  5.1 **include** (:DProdotto)  6. il sistema manda una notifica di vendita prodotto  6.1 **include** (:Notifica) |
| Postcondizioni: il prodotto è stato venduto |
| Sequenza degli eventi alternativa: Nessuna |

|  |
| --- |
| Caso d’uso: VisualizzaStatistiche |
| ID: 9 |
| Descrizione: Questo caso d’uso permette di effettuare la visualizzazione delle statistiche disponibili |
| Attori primari: Amministratore |
| Attori secondari: Nessuno |
| Precondizioni: l’amministratore desidera visualizzare le statistiche |
| Sequenza eventi principali:  1. Il caso d’uso inizia quando l’attore primario seleziona “visualizzare le statistiche”  2. Il sistema preleva dal database le statistiche  3. **if** statistiche disponibili  4. Il sistema mostra le statistiche |
| Postcondizioni: Nessuna |
| Sequenza degli eventi alternativa: Nessuna |

### 2.2.3 GESTIONE CLIENTE-PROPRIETARIO

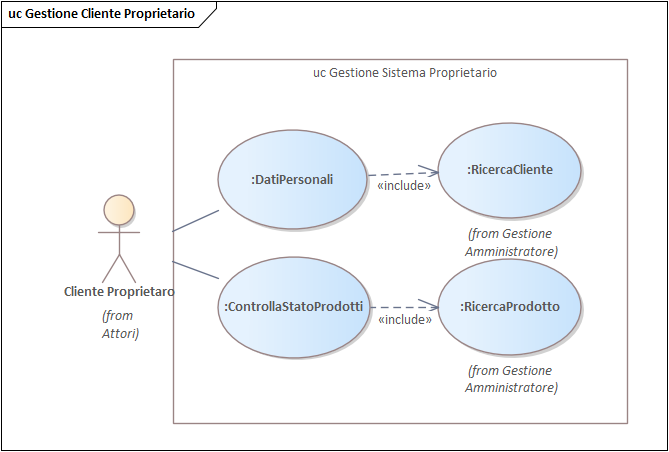


Figura - Diagramma dei casi d’uso: Gestione Cliente-Proprietario

|  |
| --- |
| Caso d’uso: ControllaStatoProdotti |
| ID: 10 |
| Descrizione: Questo caso d’uso permette di effettuare in controllo dello stato dei prodotti di un ClienteP |
| Attori primari: ClienteP |
| Attori secondari: Nessuno |
| Precondizioni: L’attore primario deve aver effettuato il login |
| Sequenza eventi principali:  1. Il caso d’uso inizia quando l’attore primario desidera visualizzare lo  stato dei prodotti  2. Il sistema ricerca i prodotti associati all’attore primario  3. **if** esistono oggetti associati  3.1. il sistema preleva dal database gli oggetti associati all’attore  Primario  3.2 Il sistema mostra all’attore primario lo stato dei prodotti associati  4. **else** il sistema restituisce un errore |
| Postcondizioni: Nessuna |

|  |
| --- |
| Caso d’uso: DatiPersonali |
| ID: 11 |
| Descrizione: Questo caso d’uso permette ad un Cliente proprietario di effettuare in controllo dei propri dati presenti nel sistema |
| Attori primari: ClienteP |
| Attori secondari: Nessuno |
| Precondizioni: L’attore primario deve aver effettuato il login |
| Sequenza eventi principali:  1. Il caso d’uso inizia quando l’attore primario desidera visualizzare i propri dati  2. Il sistema ricerca l’account associato all’attore primario  3. **if** trovato account associato  3.1. il sistema preleva dal database i dati associati all’attore  primario  3.2 Il sistema mostra all’attore primario i propri dati  4. **else** il sistema restituisce un errore |
| Postcondizioni: Nessuna |

### 2.2.4 GESTIONE SISTEMA

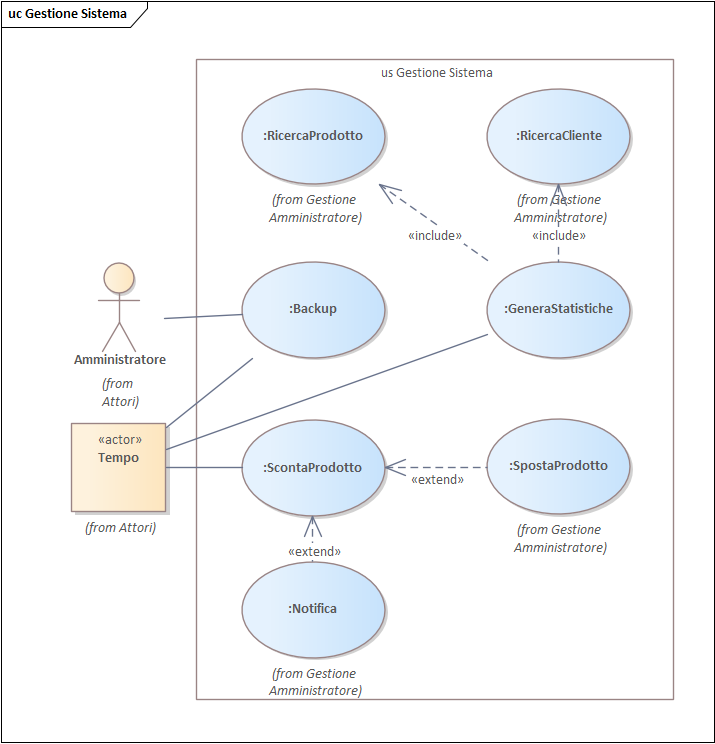


Figura - Diagramma dei casi d’uso: Gestione Sistema

|  |
| --- |
| Caso d’uso: Backup |
| ID: 12 |
| Descrizione: il sistema effettua il backup dei dati |
| Attori primari: Tempo |
| Attori secondari: Nessuno |
| Precondizioni: Nessuno il programma è stato chiuso |
| Sequenza eventi principali:  1. Il caso d’uso inizia quando il programma è stato chiuso  2. Il sistema preleva i dati dal database  3. Il sistema copia su disco i dati prelevati |
| Postcondizioni: I dati sono stati copiati. |
| Sequenza degli eventi alternativa: Nessuna |

|  |
| --- |
| Caso d’uso: GeneraStatistiche |
| ID: 13 |
| Descrizione: Il sistema genera statistiche |
| Attori primari: Tempo |
| Attori secondari: Nessuno |
| Precondizioni: Nessuna |
| Sequenza eventi principali:  1. Il caso d’uso inizia quando il sistema viene chiuso  2. Il sistema preleva dati riguardanti i prodotti  3. Il sistema preleva dati riguardanti le ricevute  4. Il sistema preleva dati riguardanti i Clienti Proprietari  5. Il sistema genera statistiche  6. Il sistema salva i dati su disco |
| Postcondizioni: Le statistiche generate sono salvate su disco |
| Sequenza degli eventi alternativa: Nessuna |

|  |
| --- |
| Caso d’uso: ScontaProdotto |
| ID: 14 |
| Descrizione: Il sistema sconta il prodotto |
| Attori primari: Tempo |
| Attori secondari: Nessuno |
| Precondizioni Segmento 1: Sono passati 60 giorni dalla registrazione del prodotto  Precondizioni Segmento 2: Sono passati 90 giorni dalla registrazione del prodotto  Precondizioni Segmento 3: Sono passati 120 giorni dalla registrazione del prodotto |
| Sequenza eventi Segmento 1:  1.Il caso d’uso inizia quando sono passati 60 giorni dalla registrazione del prodotto.  2.Il prodotto viene scontato del 30% rispetto al prezzo iniziale  Sequenza eventi Segmento 2:  1.Il caso d’uso inizia quando sono passati 90 giorni dalla registrazione del prodotto.  2.Il prodotto viene scontato del 40% rispetto al prezzo iniziale  Sequenza eventi Segmento 3:  1.Il caso d’uso inizia quando sono passati 120 giorni dalla registrazione del prodotto  2.Il prodotto viene scontato del 50% rispetto al prezzo iniziale |
| Postcondizioni: Il prodotto viene scontato |
| Sequenza degli eventi alternativa: Nessuna |

### 2.2.5 GESTIONE USER

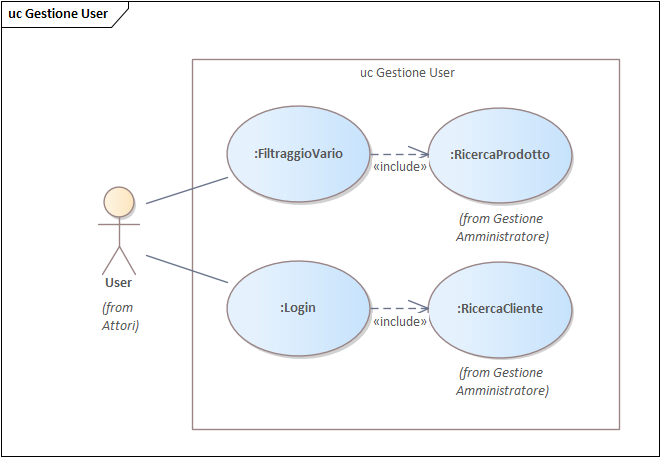


Figura - Diagramma dei casi d’uso: Gestione User

|  |
| --- |
| Caso d’uso: FiltraggioVario |
| ID: 15 |
| Descrizione: Filtraggio dei prodotti in varie modalità |
| Attori primari: User |
| Attori secondari: Nessuno |
| Precondizioni: l’utente vuole effettuare il filtraggio dei prodotti |
| Sequenza eventi principali:  1. il caso d’uso inizia quando l’utente vuole filtrare i prodotti disponibili  2. l’utente inserisce la modalità di filtraggio e preme invio  3. il sistema effettua il filtraggio con il filtro applicato  4. **if** ci sono oggetti nella categoria cercata  4.1. mostra all’utente la lista degli oggetti trovati  5. **else** mostra all’utente un messaggio indicando che non sono stati trovati  oggetti nella categoria |
| Postcondizioni: Nessuna |
| Sequenza degli eventi alternativa: nessuna |

|  |
| --- |
| Caso d’uso: Login |
| ID: 16 |
| Descrizione: L’User accede al sistema |
| Attori primari: User |
| Attori secondari: Nessuno |
| Precondizioni: Nessuna |
| Sequenza eventi principali:  1. Il caso d’uso inizia quando l’User vuole accedere al sistema  2. L’User inserisce il nome utente e la password  3. Il Sistema controlla i dati inseriti  4. Login |
| Postcondizioni: L’User è loggato |
| Sequenza degli eventi alternativa:  La sequenza degli eventi alternativa inizia dal punto 3  4.Il nome utente e la password non esistono o sono errati  5.Il sistema restituisce un messaggio di errore |

## 2.4 MATRICE MAPPING DEI REQUISITI

È utile tenere traccia della realizzazione tra i casi d’uso individuati e i requisiti in modo da comprendere se la progettazione sta andando in linea con l’idea di cosa il sistema debba essere in grado di svolgere.

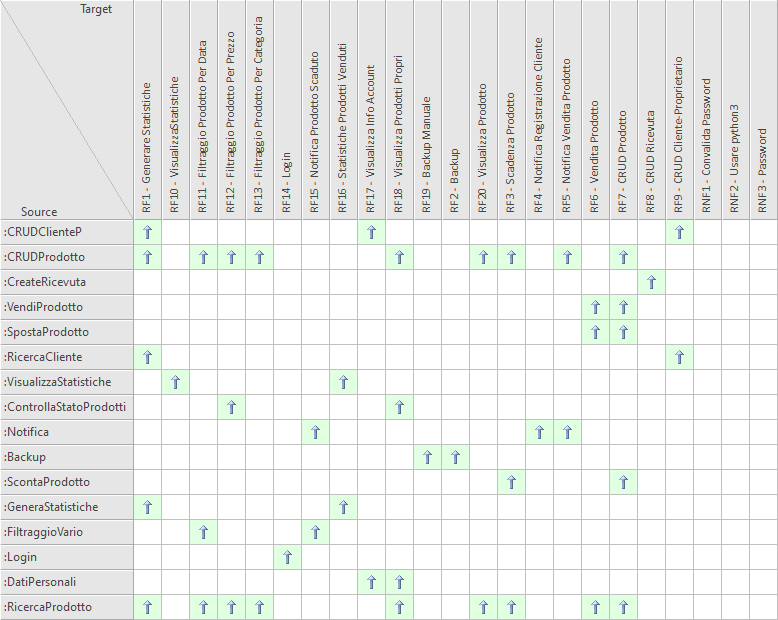


Figura - Matrice mapping dei requisiti

# DIAGRAMMI DI ANALISI

Il diagramma di analisi permette una prima rappresentazione delle classi di oggetti individuate durante l’analisi, la loro individuazione è stata frutto delle tecniche di associazione tra ogni aspetto del negozio fisico con una rappresentazione a classi di oggetti e l’interazione tra classi con altre classi. La suddivisione in macroaree *Attività & Servizio* è stata realizzata valutando l’interazione tra le classi di oggetti e vedendo che gli oggetti appartenenti alla stessa macroarea sono maggiormente dipendenti tra essi. Sono state attribuite due definizioni diverse alle macroaree:

*Attività:* rappresenta il negozio nel suo complesso di utilizzo da parte di utenti.

*Servizio:* rappresenta a seconda degli aspetti fisici del negozio in questione i servizi che offre*.*

L’accesso ai membri interni alle classi di oggetti è stato reso pubblico per assecondare le implicazioni derivanti dal vincolo del RNF1.

## 3.1 PACKAGE DI ANALISI

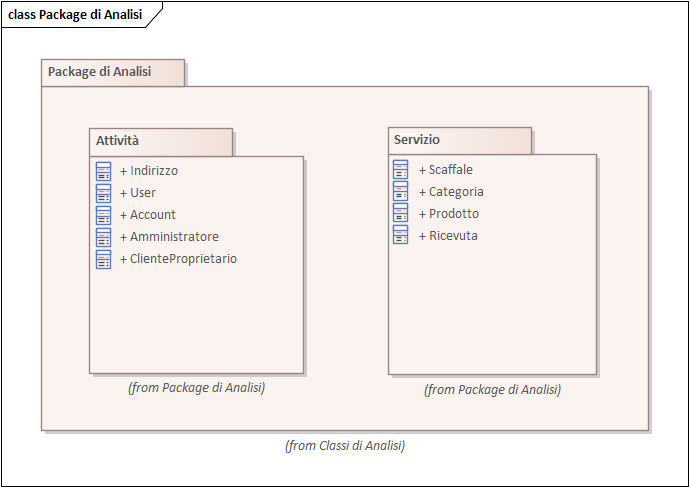


Figura - Package di analisi

## 3.2 DIAGRAMMI DELLE CLASSI DI ANALISI

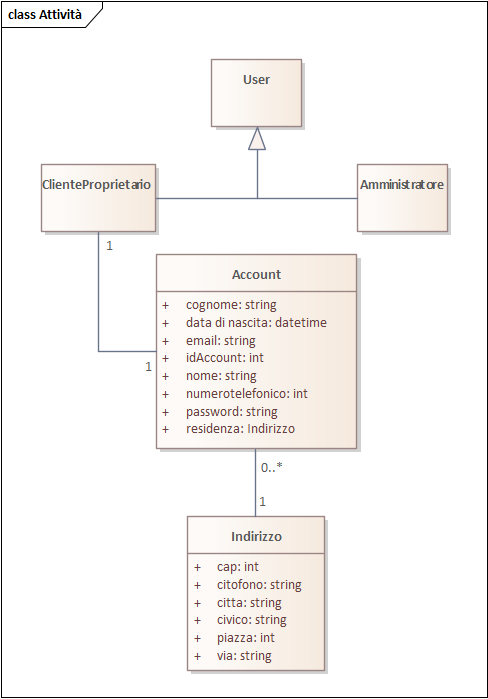


Figura - Diagramma di analisi delle classi di Attività

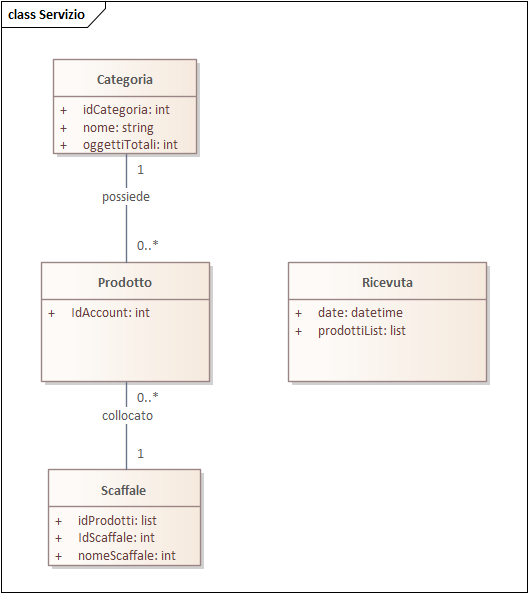


Figura - Diagramma di analisi delle classi di Servizio

## 3.3 DIAGRAMMI DI SEQUENZA

I diagrammi di sequenza modellano le interazioni tra oggetti e attori, rappresentano una specifica sequenza di interazione per svolgere una funzione, l’ordine in cui le interazioni avvengono dopo l’attivazione da parte di un attore che interagisce con il sistema è stabilito dai diagrammi, questo permetterà un’implementazione più agevole.

### 3.3.1 BACKUP

Tale diagramma modella l’interazione da parte dell’attore Tempo con la richiesta di effettuare un backup dei dati. L’interazione con il Database avviene nella breve linea di vita in cui vi è la copia dei dati in esso contenuti.

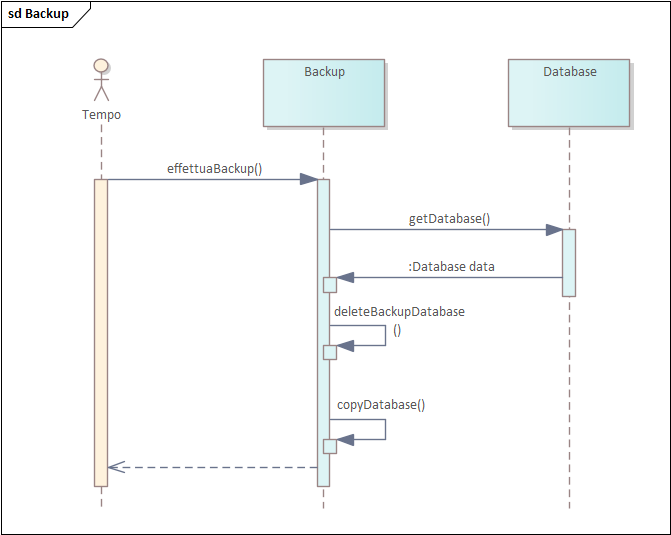


Figura - Diagramma di sequenza: Backup

### 3.3.2 CONTROLLASTATOPRODOTTO

Tale diagramma modella l’interazione dell’attore Cliente Proprietario per controllare lo stato degli oggetti registrati nel sistema.

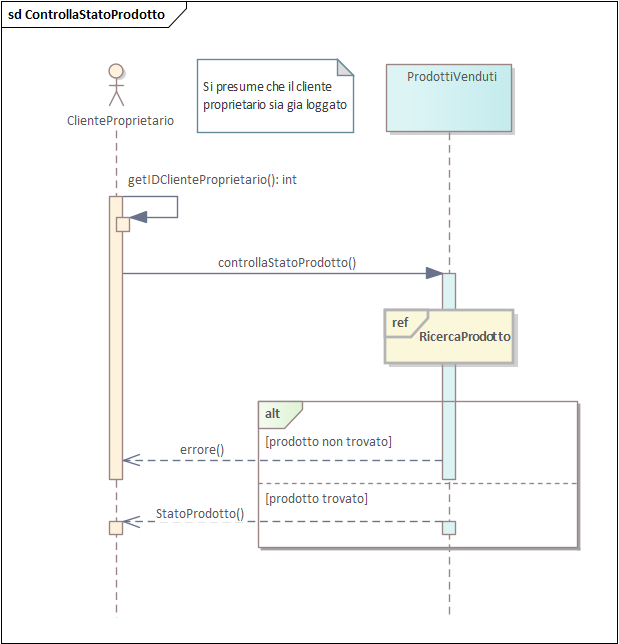


Figura - Diagramma di sequenza: ControllaStatoProdotto

### 3.3.3 CRUD PRODOTTO

Tali diagrammi modellano le interazioni CRUD dell’attore Amministratore con un prodotto presente nel sistema

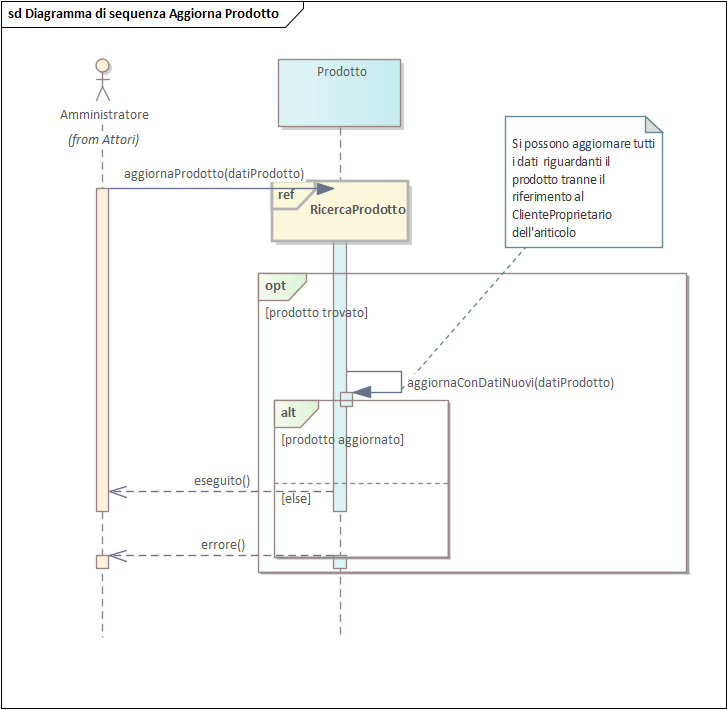


Figura - Diagramma di sequenza: Aggiorna Prodotto

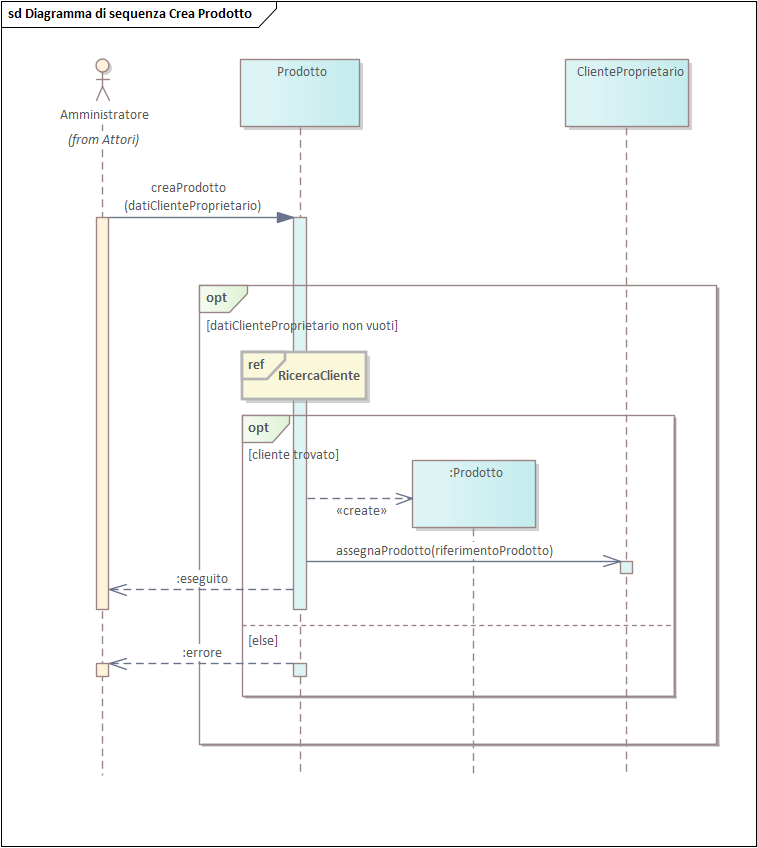


Figura - Diagramma di sequenza: Crea Prodotto

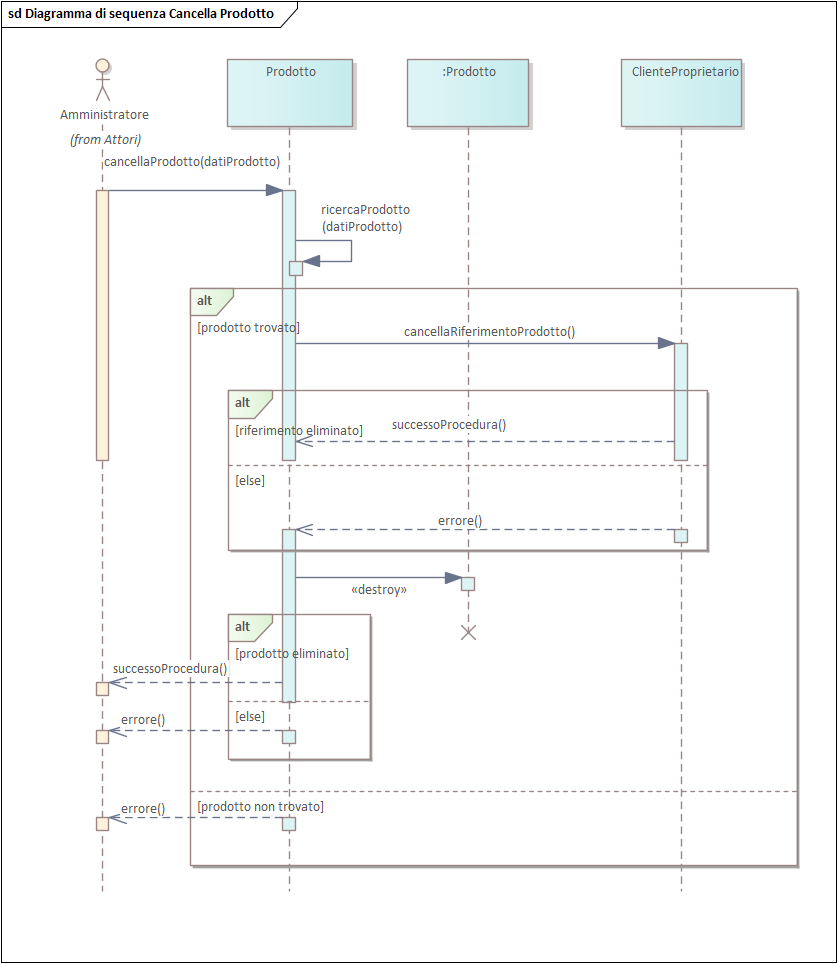


Figura - Diagramma di sequenza: Cancella Prodotto

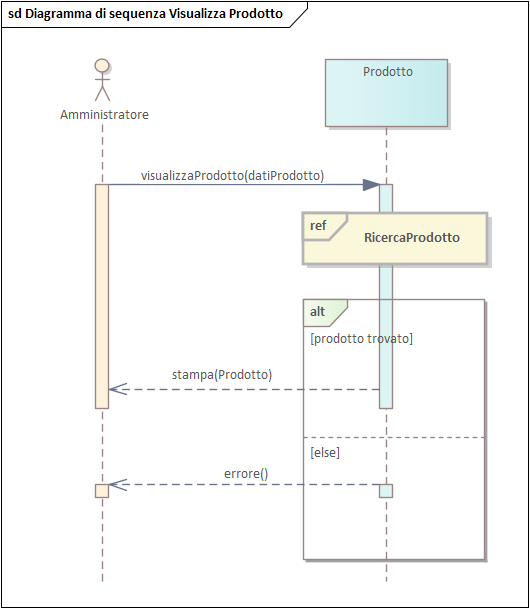


Figura - Diagramma di sequenza: Visualizza Prodotto

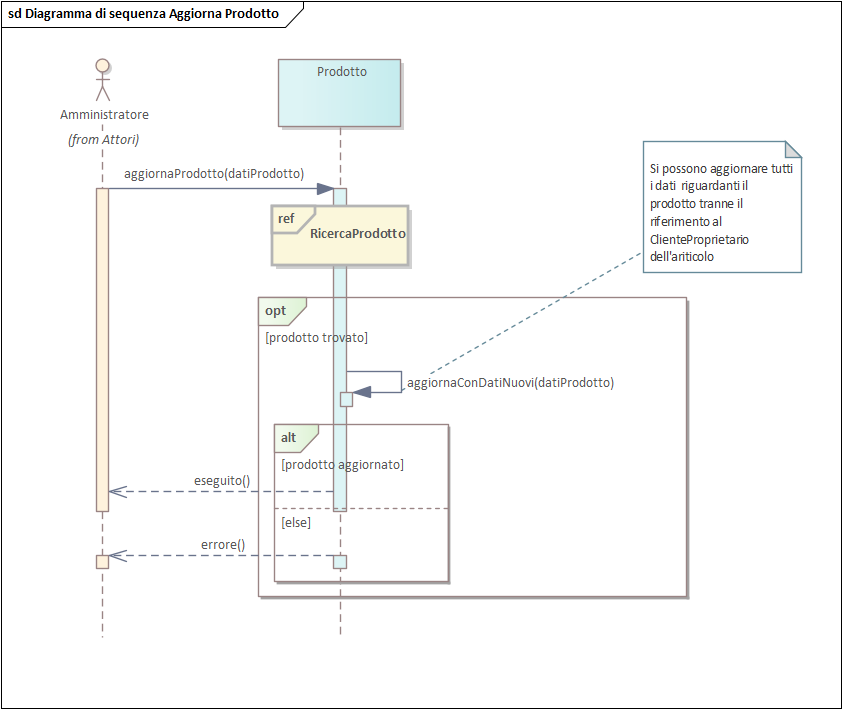


Figura - Diagramma di sequenza: Visualizza Prodotto

### 3.3.4 LOGIN

Tale diagramma modella l’interazione dell’attore User in possesso di un account per l’accesso ai servizi del sistema.

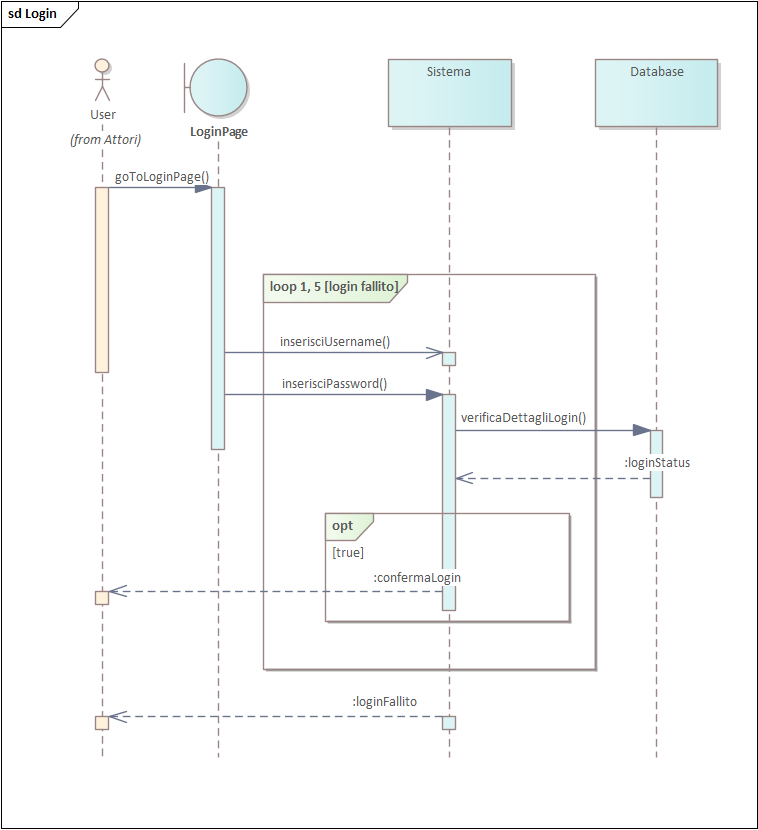


Figura - Diagramma di sequenza: Login

### 3.3.5 NOTIFICA

Tale diagramma modella l’interazione dell’attore Tempo con l’invio di una e-mail di notifica.

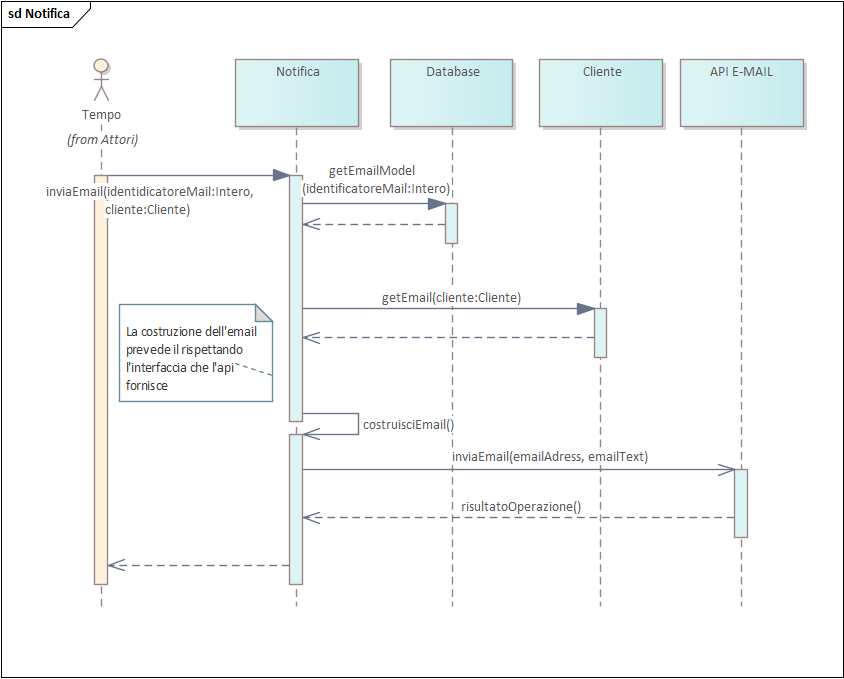


Figura - Diagramma di sequenza: Notifica

### 3.3.6 RICERCACLIENTE

Tale diagramma modella l’interazione dell’attore Amministratore con il sistema per ricercare un Cliente-Proprietario registrato.

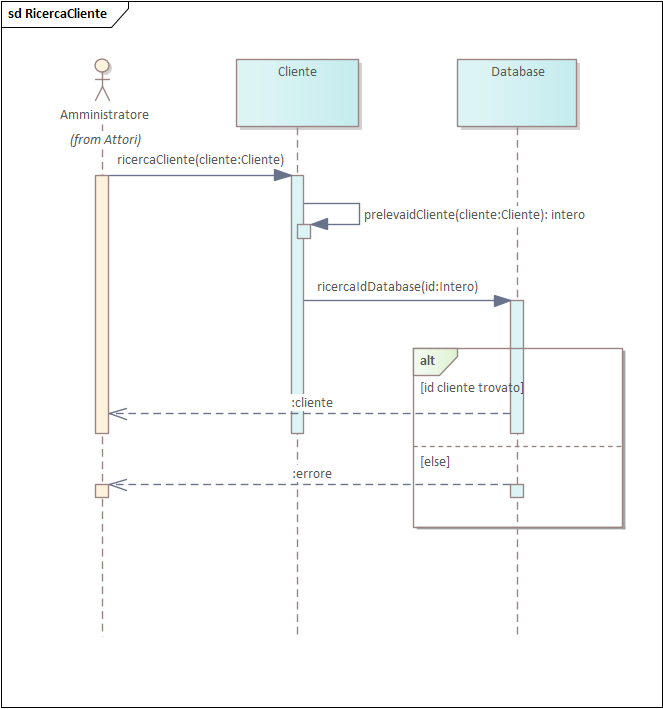


Figura - Diagramma di sequenza: Ricerca Cliente

### 3.3.7 RICERCAPRODOTTO

Tale diagramma modella l’interazione dell’attore Amministratore con il sistema per la ricerca di un prodotto.

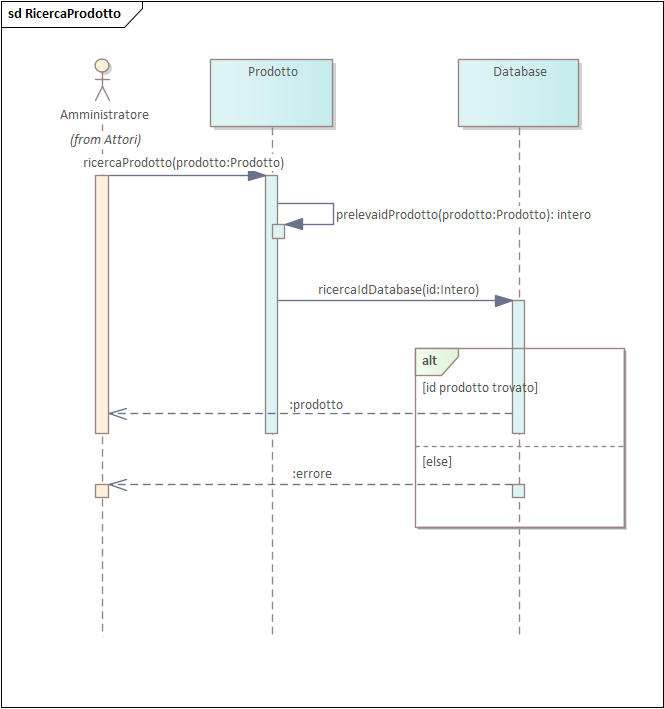


Figura - Diagramma di sequenza: RicercaProdotto

### 3.3.8 VENDIPRODOTTO

Tale diagramma modella l’interazione dell’attore Amministratore con il sistema per la vendita di un prodotto.

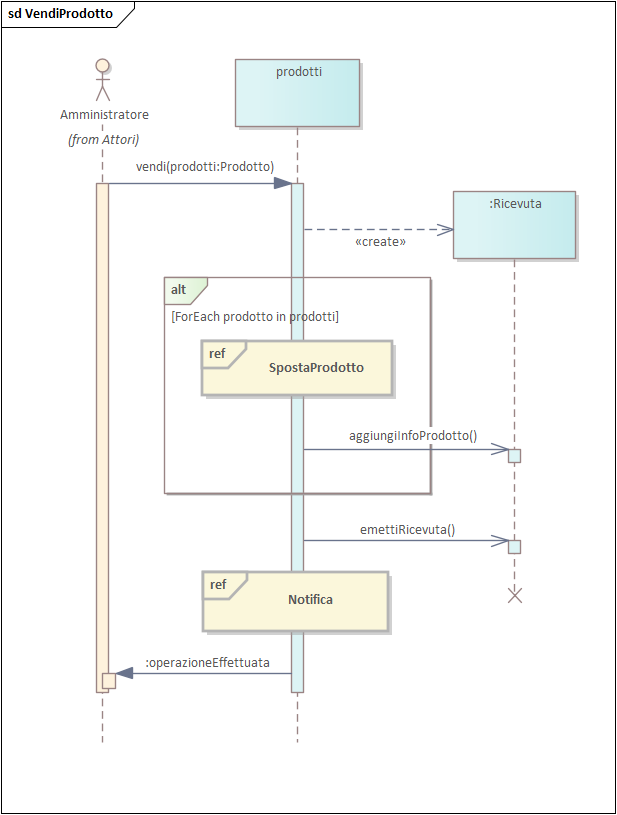


Figura - Diagramma di sequenza: VendiProdotto

### 3.3.9 VISUALIZZASTATISTICHE

Tale diagramma modella l’interazione dell’attore Amministratore con il sistema per la visualizzazione di statistiche disponibili.

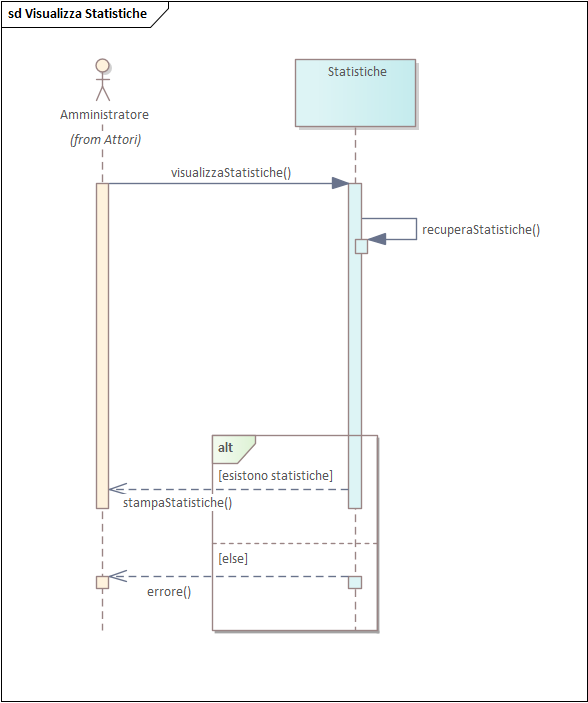


Figura - Diagramma di sequenza: Visualizza Statistiche

# DIAGRAMMI DI PROGETTAZIONE

Sono una raffinazione dei diagrammi delle classi di analisi infatti le seguono temporalmente. Procedendo con la progettazione

## 4.1 DIAGRAMMI DELLE CLASSI DI PROGETTAZIONE

Immagine che contiene testo

Descrizione generata automaticamente

Figura - Package di Progettazione



Figura - Diagramma di progettazione delle classi di Attività

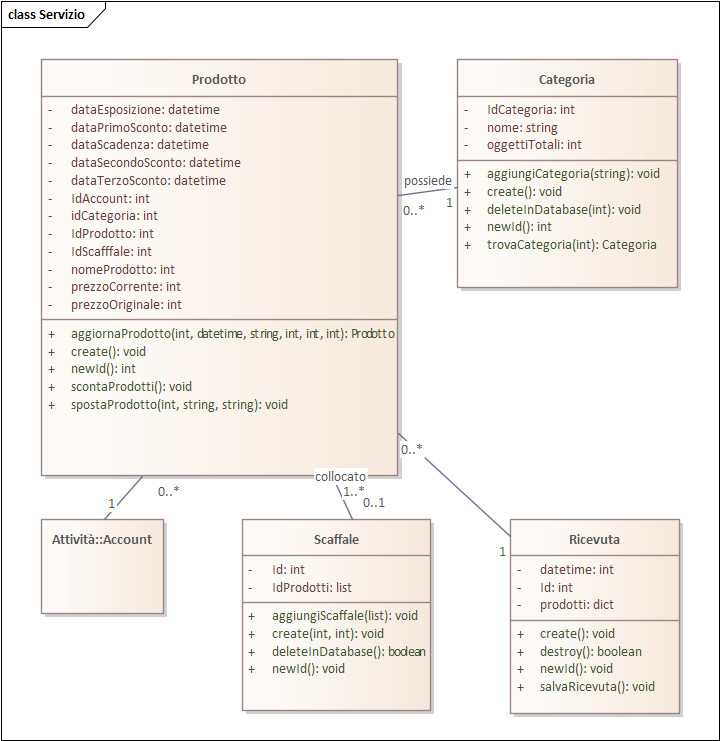


Figura - Diagramma di progettazione delle classi di Servizio

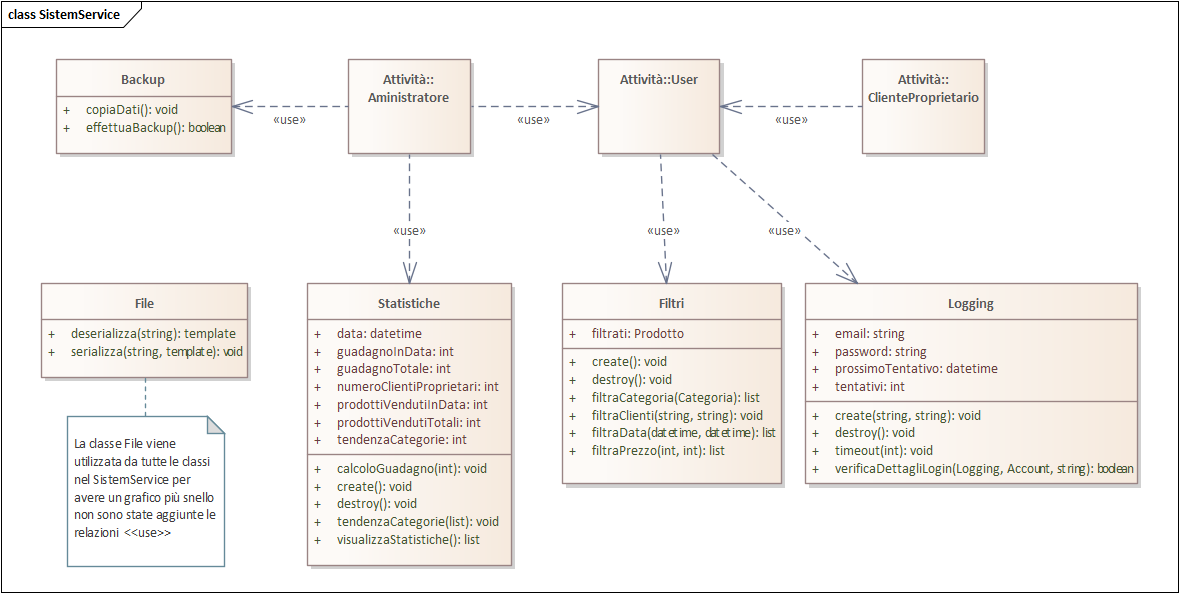


Figura - Diagramma di progettazione delle classi di SistemService

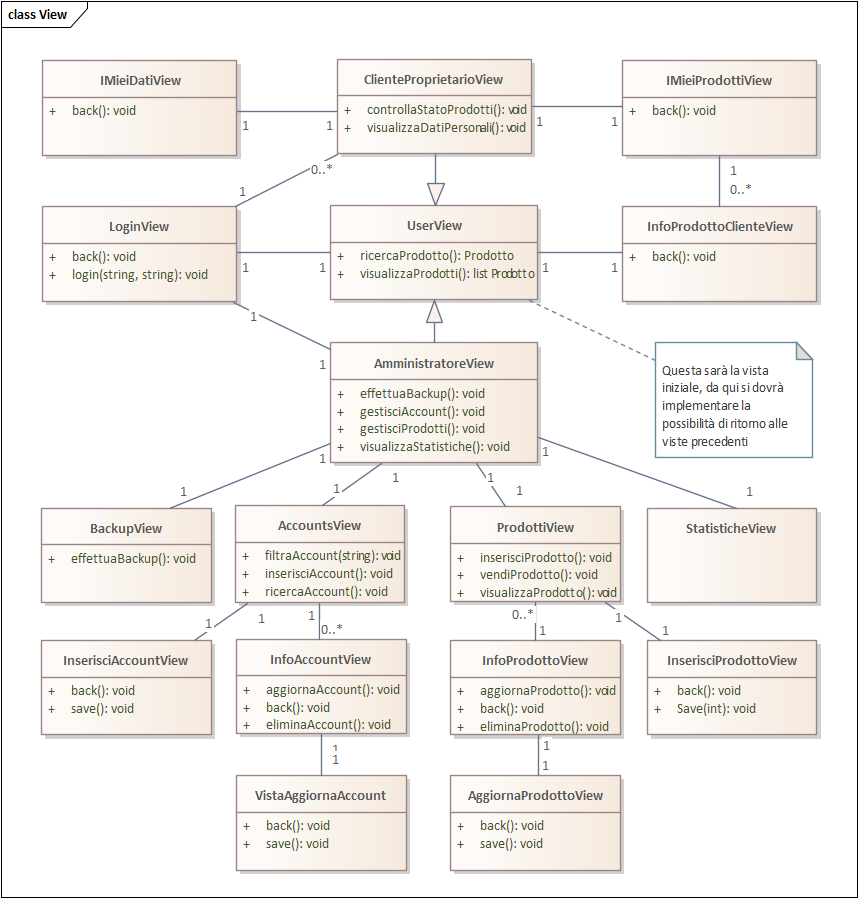


Figura - Diagramma di progettazione delle classi di view

## 4.2 DIAGRAMMI DEI COMPONENTI

Una progettazione per componenti è alla base della moderna ingegneria del software, tale approccio permette di individuare già nella fase di sviluppo dei confini tra varie parti del sistema, grazie a questo il software potrà essere rimodellato e adattato a varie situazioni di funzionamento. Di seguito si riportano alcuni diagrammi di come questo concetto è stato pensato per il nostro sistema con un determinato grado di dettagli interni.

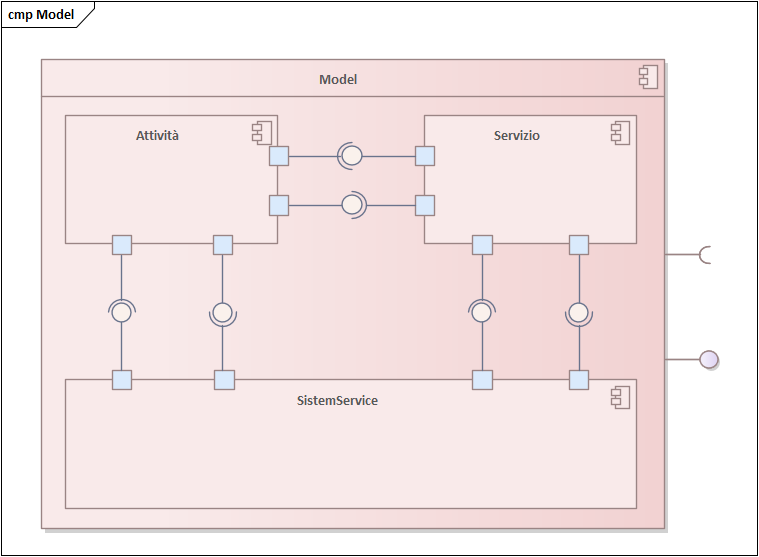


Figura - Diagramma dei componenti: Model

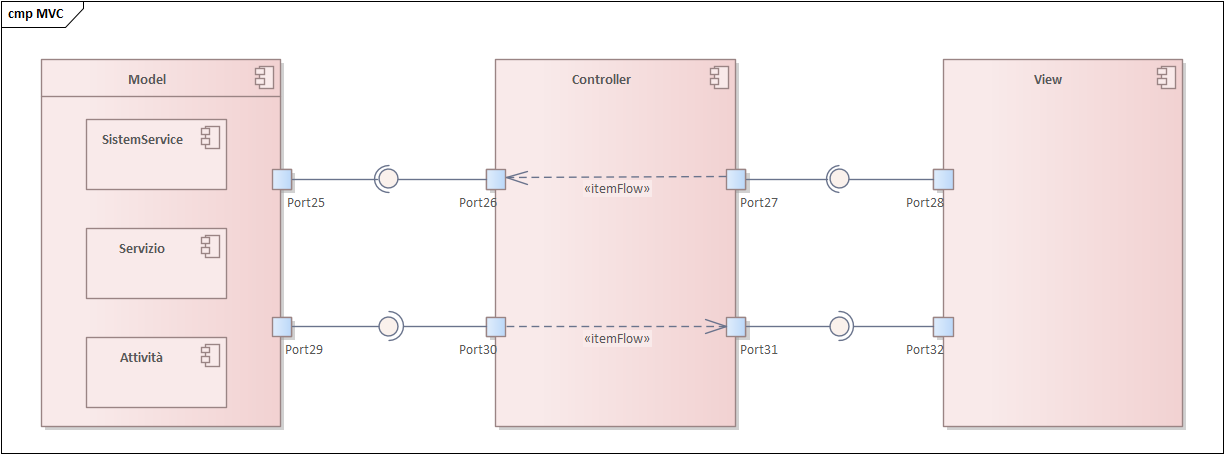


Figura - Diagramma dei componenti: MVC

## 4.3 DIAGRAMMI DELLE MACCHINE A STATI

### 4.3.1 MACCHINE A STATI: BACKUP

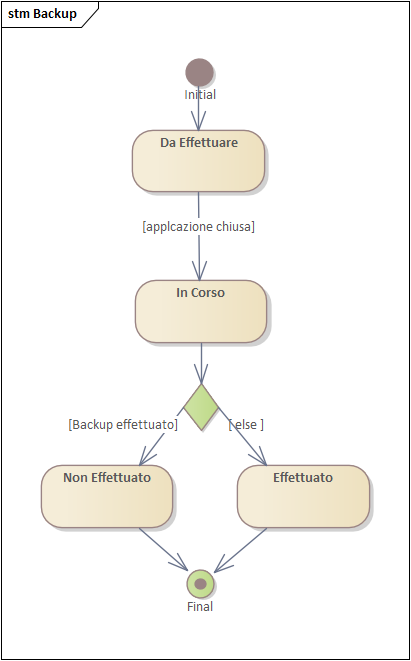


Figura - Diagramma della macchina a stati: Backup

### 4.3.2 MACCHINE A STATI: CATEGORIA

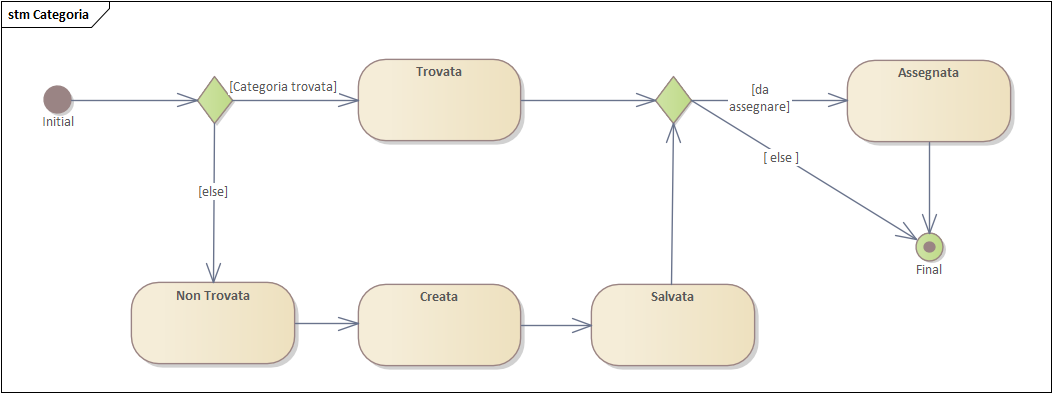


Figura - Diagramma della macchina a stati: Categoria

### 4.3.3 MACCHINE A STATI: LOGIN

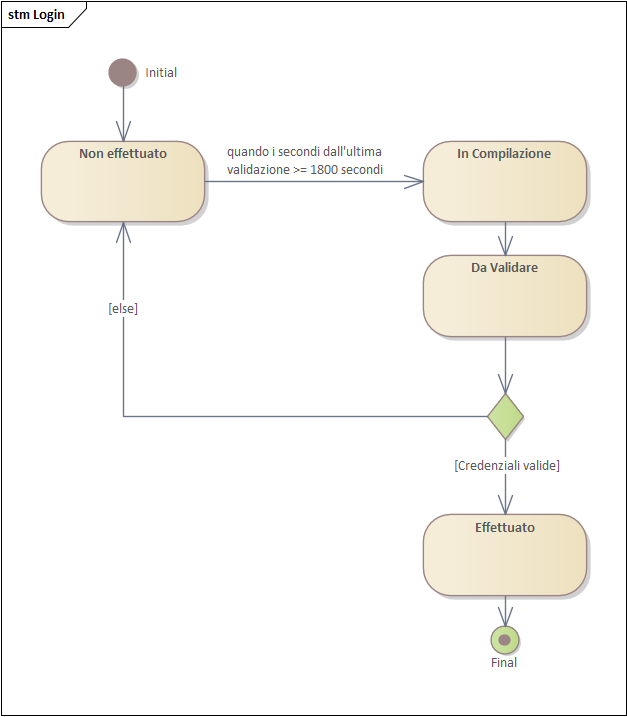


Figura - Diagramma della macchina a stati: Login

### 4.3.4 MACCHINE A STATI: PRODOTTO

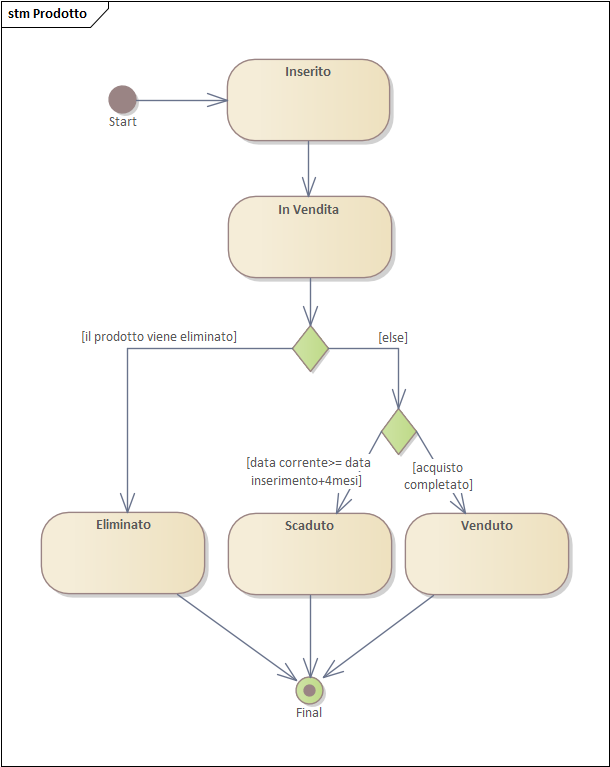


Figura - Diagramma della macchina a stati: Prodotto

### 4.3.5 MACCHINE A STATI: RICEVUTA

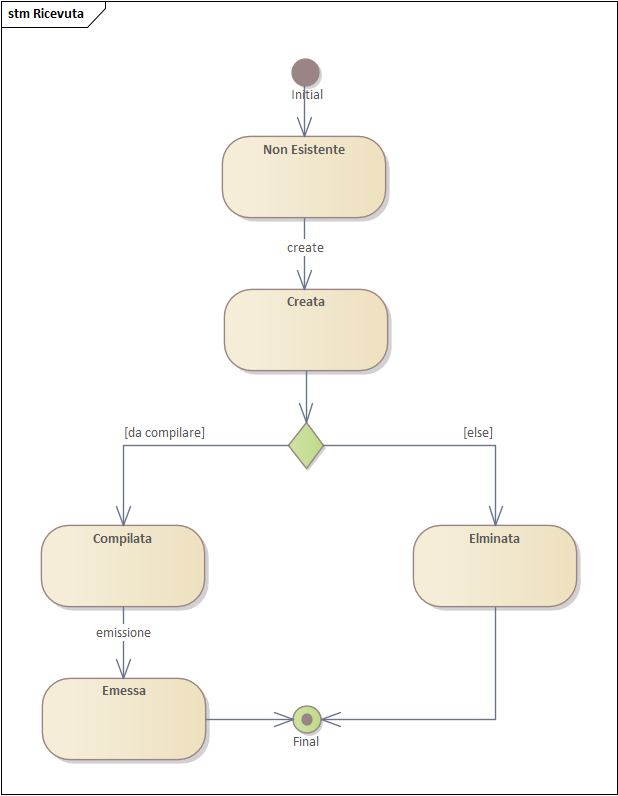


Figura - Diagramma della macchina a stati: Ricevuta

# IMPLEMENTAZIONE

L’implementazione del sistema avverrà in Python 3.10.4

Pattern utilizzati durante la realizzazione:

* MVC (Model View Controller)
* Builder

Il codice eseguibile verrà distribuito sui seguenti sistemi operativi:

* Windows
* Linux
* macOS

## 5.1 DIAGRAMMA DI DEPLOYMENT

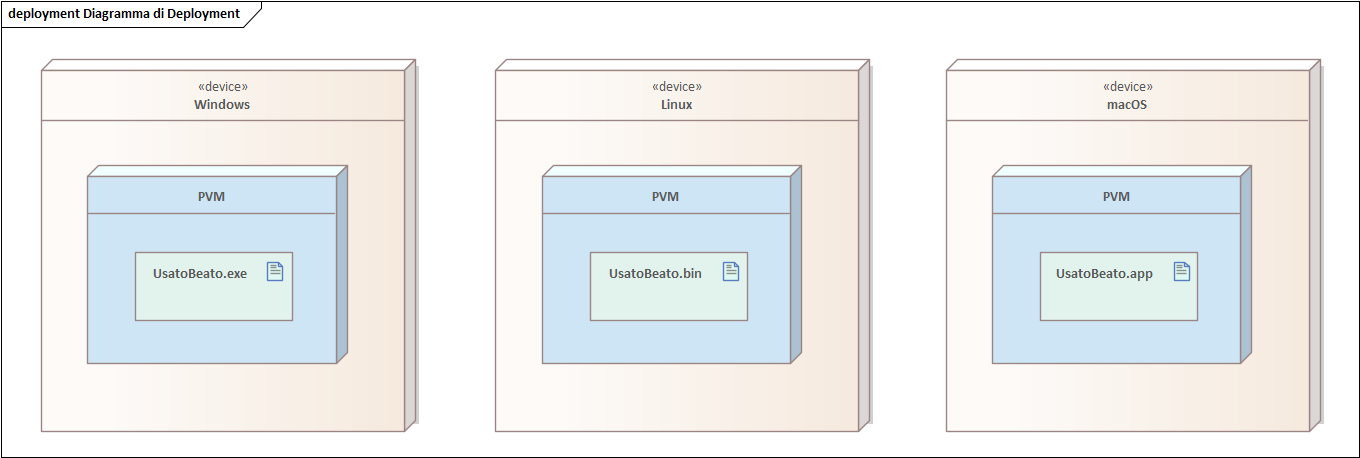


Figura - Diagramma di deployment

# MOCKUP

I seguenti Mockup sono stati realizzati con il modulo ufficiale per Python: PySide6

Tale modulo permette l’accesso completo a **Qt 6.0+** **framework** e inoltre integra ancheil **QtDesigner,** un software che ci ha permesso la progettazione facilitata dei Mockup.

Il fine è stato rendere dinamica la progettazione delle interfacce grafiche e quindi evitare una struttura monolitica, per questo motivo è stato utilizzato per la progettazione dei Mockup il suddetto software. Vi è la possibilità di compilare la progettazione grafica nel designer in un “NomeFile.ui” e di utilizzarlo come componente nella gestione delle interfacce grafiche. Il fatto di aver reso l’interfaccia grafica un componente permette di effettuare manutenzioni future in maniera estremamente efficiente.

La seguente versione dei Mockup è aggiornata alla prima versione di UsatoBeato rilaciata.

## VISTA LOGIN

Tale vista è accessibile dalla vista del User, è necessaria per poter effettuare l’accesso al proprio account

Immagine che contiene testo, screenshot, elettronico

Descrizione generata automaticamente

Figura – vista login

## VISTA USER

Tale vista è caricata all’avvio dell’eseguibile, vi sono tre barre di attivazione per la selezione di filtri da applicare ai prodotti disponibili nel negozio.



Figura - vista user

## 6.3 VISTA AMMNISTRATORE

Nella vista NUMERO che è accessibile dall’utente Amministratore, è possibile visualizzare le informazioni di un prodotto di interesse.

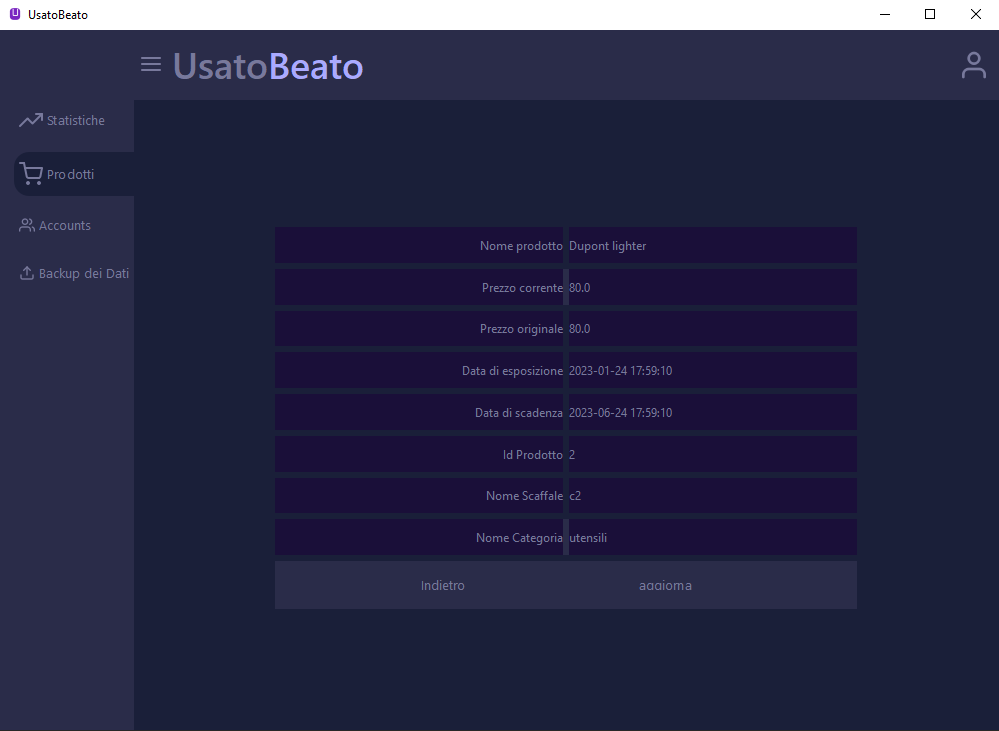


Figura - vista info prodotto

Nella vista NUMERO che è accessibile dall’utente Amministratore, è possibile visualizzare le informazioni di un account registrato nel sistema come Cliente-Proprietario.

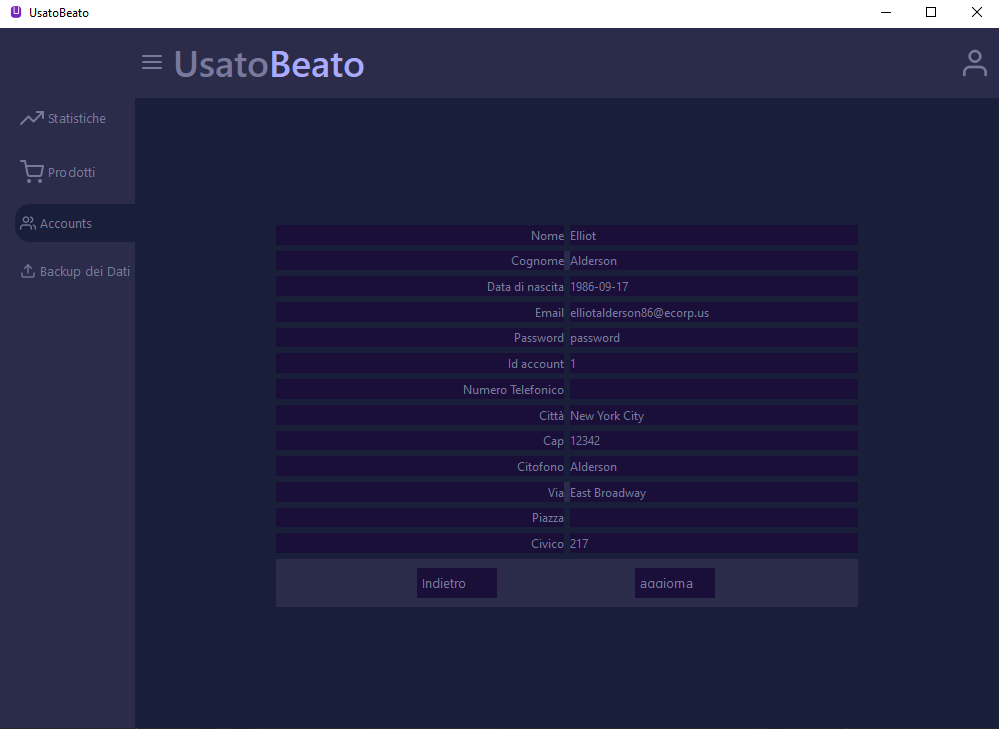


Figura - vista info account

Nella vista NUMERO che è accessibile dall’utente Amministratore, è possibile visualizzare le statistiche disponibili per valutare l’andamento del negozio

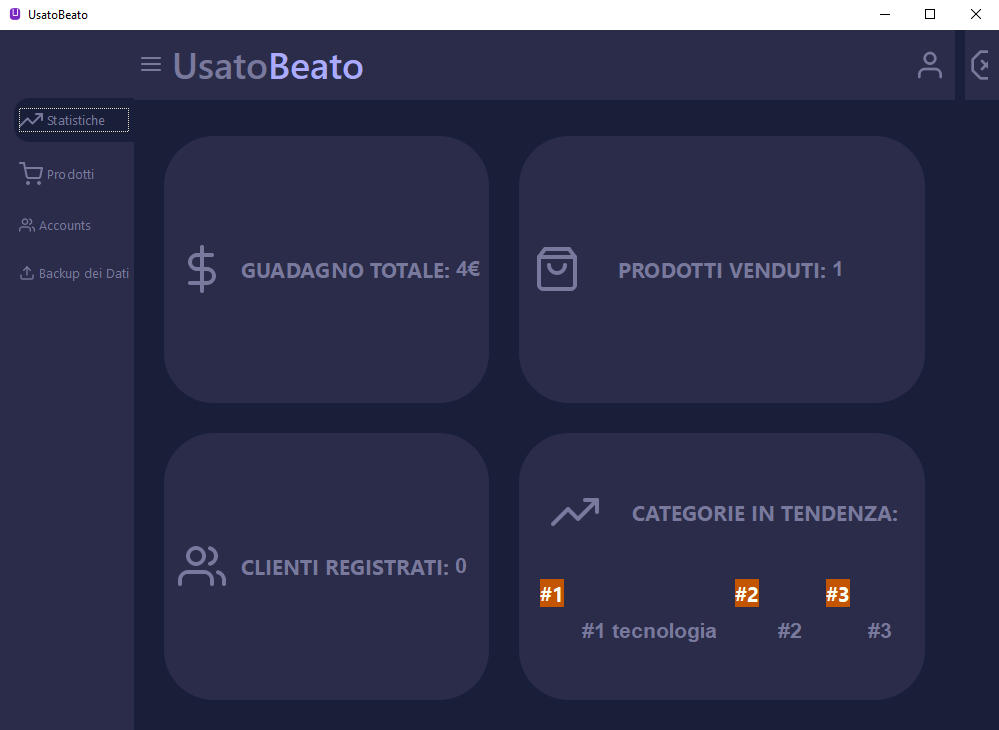


Figura - vista statistiche amministratore

Nella vista NUMERO che è accessibile dall’utente Amministratore, è possibile visualizzare la lista dei prodotti disponibili nel negozio.

Immagine che contiene testo, screenshot, monitor

Descrizione generata automaticamente

Figura - vista prodotti amministratore

Nella vista NUMERO che è accessibile dall’utente Amministratore, è possibile effettuare l’operazione di backup.

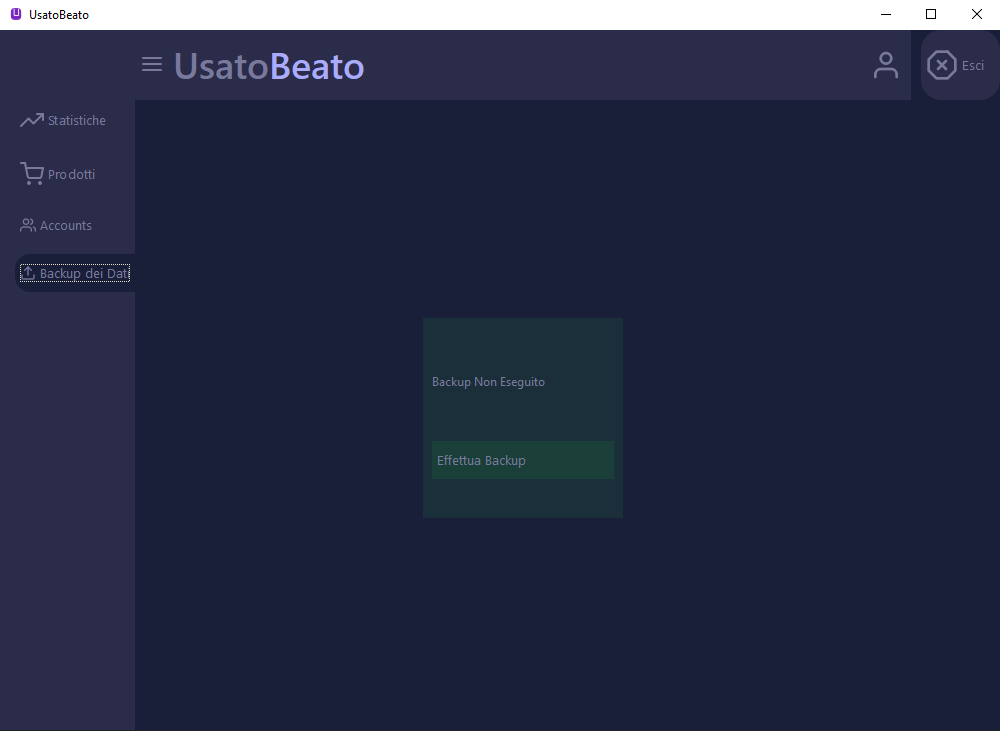


Figura - vista backup

Nella vista NUMERO che è accessibile dall’utente Amministratore, è possibile visualizzare la lista degli utenti registrati nel sistema

Immagine che contiene testo, screenshot, monitor

Descrizione generata automaticamente

Figura - vista account amministratore

## 6.4 VISTA CLIENTE PROPRIETARIO

Nella vista 47 accessibile dall’utente Cliente-Proprietario, è possibile visualizzare lo stato degli oggetti associati all’account del cliente.



Figura - vista prodotti del cliente proprietario

Nella vista 48 accessibile dall’utente Cliente-Proprietario, è possibile visualizzare i propri dati.

Immagine che contiene testo, monitor, screenshot, schermo

Descrizione generata automaticamente

Figura - vista dati del cliente proprietario

Nella vista 49 accessibile dall’utente Cliente-Proprietario, è possibile vedere la lista degli oggetti disponibili all’interno del negozio.

Immagine che contiene testo, screenshot, monitor

Descrizione generata automaticamente

Figura - vista home cliente proprietario

# 7 PYUNIT

Il testing del sistema è stato portato avanti durante la progettazione con un picco nella fase di fine progettazione del **Model** del sistema, i test eseguiti sono stati fondamentali per il corretto funzionamento del codice, eseguiti grazie alla libreria *TestCase* del modulo *unittest.*

Di seguito un estratto di alcuni test eseguiti:

Immagine che contiene testo

Descrizione generata automaticamente

Figura - test vediProdotti

Immagine che contiene testo

Descrizione generata automaticamente

Figura - test inserisciAccount

Immagine che contiene testo

Descrizione generata automaticamente

Figura - test aggiornaProdotto

Immagine che contiene testo

Descrizione generata automaticamente

Figura - test aggiornaAccount