**1 DESCRIZIONE IN LINGUAGGIO NATUREALE**

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

REGISTRAZIONE UTENTE

Il cliente-proprietario arrivato al negozio dovrà registrarsi insieme all’amministratore, gli verrà fornito un codice cliente, in caso non sia già stata effettuata la registrazione precedentemente, fornendo i propri dati personali, la propria e-mail e la password concordata con l’amministratore. A registrazione effettuata, 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.

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’

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.

**1.1 GLOSSARIO DEI TERMINI**

**2 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.* I primi sono delle caratteristiche che il sistema deve garantire all’utilizzatore mentre i secondi sono dei vincoli imposti dal sistema.

**2.1 REQUISITI DEL SISTEMA**

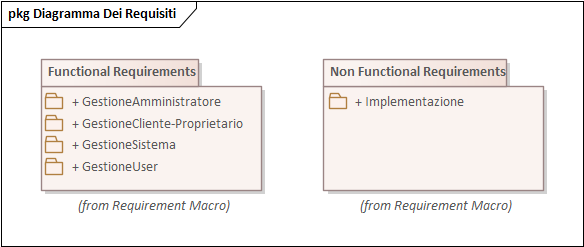


Figura NUMERO: Distribuzione dei package dei requisiti

**2.1.1 REQUISITI FUNZIONALI**

Immagine che contiene testo

Descrizione generata automaticamente

Figura NUMERO: Diagramma dei 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

**2.1.2 REQUISITI NON FUNZIONALI**

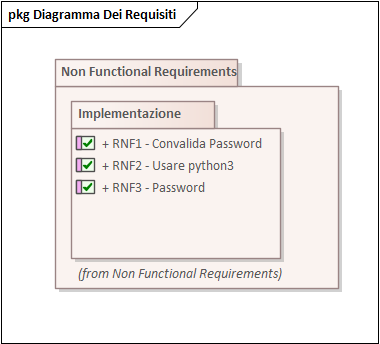


Figura NUMERO: Diagramma dei requisiti non funzionali

**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

**2.2 DIAGRAMMI DEI CASI D’USO**

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 con gli 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.

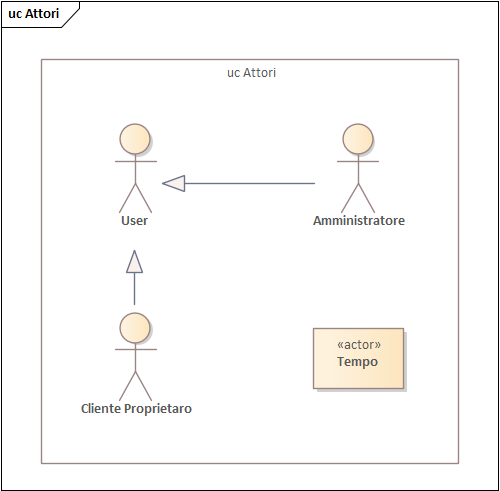


Figura NUMERO: Diagramma degli attori

**2.2.2 GESTIONE AMMINISTRATORE**

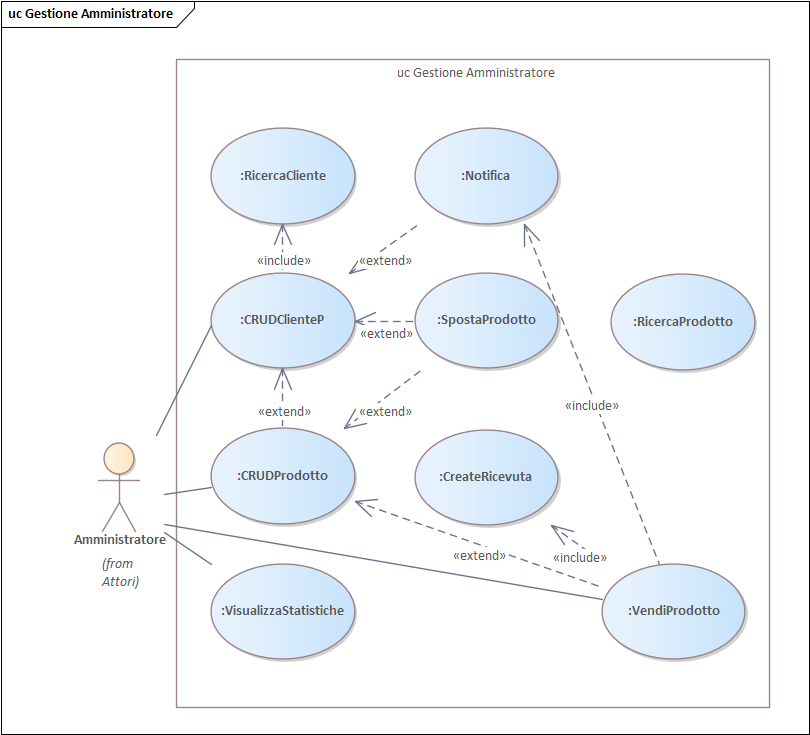


Figura NUMERO: Diagramma Gestione Amministratore

|  |
| --- |
| 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: nessuna |
| Sequenza degli eventi alternativa: nessuna |

|  |
| --- |
| Caso d’uso: CRUDClienteP |
| 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 ClienteP  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 ClienteP 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 ClienteP  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: nessuna |
| 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 email 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 email 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 email 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’email al ClienteProprietario |
| Postcondizioni: Nessuna |
| 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. **include** (:RicercaProdotto)  4. **include** (:CRicevuta)  5. **include** (:DProdotto)  6. **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: x  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**

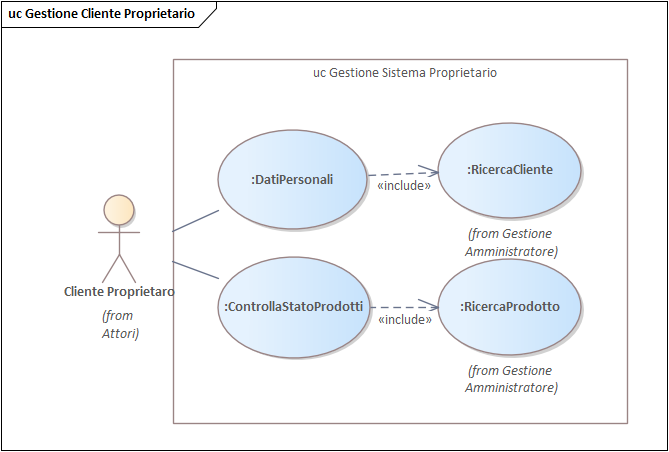


Figura NUMERO: Diagramma Gestione Cliente-Proprietario

|  |
| --- |
| 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 di effettuare in controllo dei propri dati |
| 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**

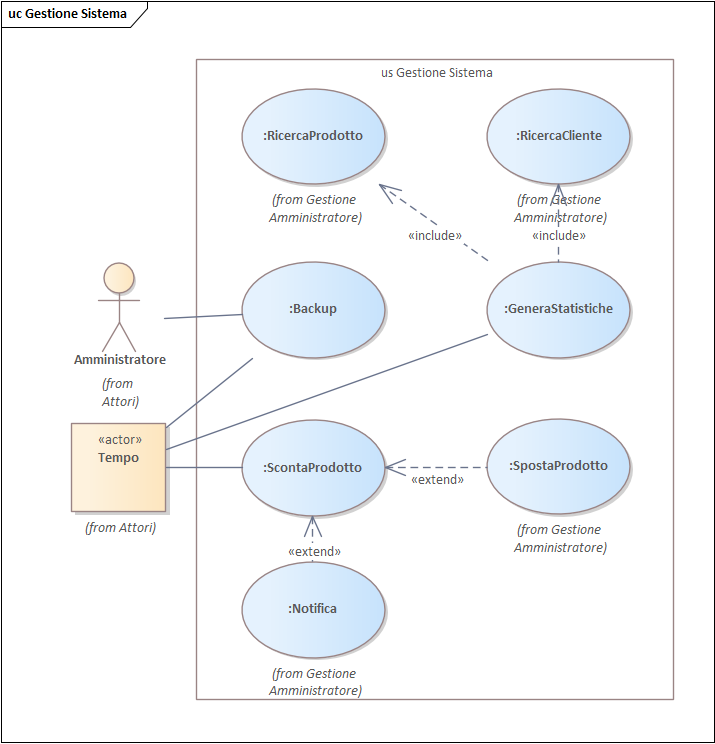


Figura NUMERO: Diagramma Gestione Sistema

|  |
| --- |
| 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**

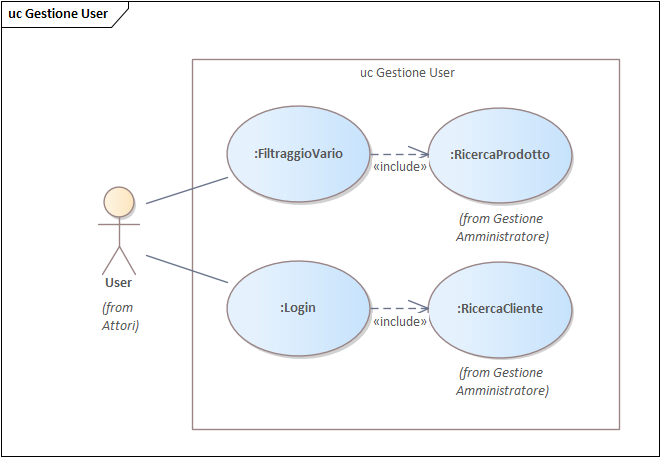


Figura NUMERO: Diagramma Gestione User

|  |
| --- |
| 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 |

**3 DIAGRAMMI DI ANALISI**

**3.1 p DI ANALISI**

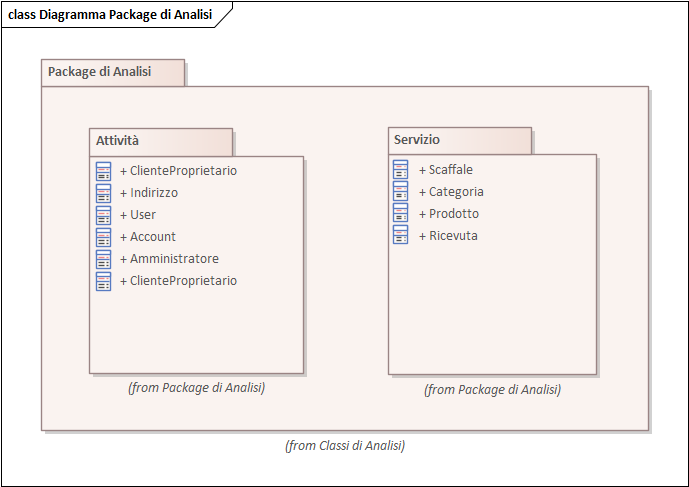


Figura NUMERO: Diagramma del package di analisi

**3.2 DIAGRAMMI DELLE CLASSI DI ANALISI**

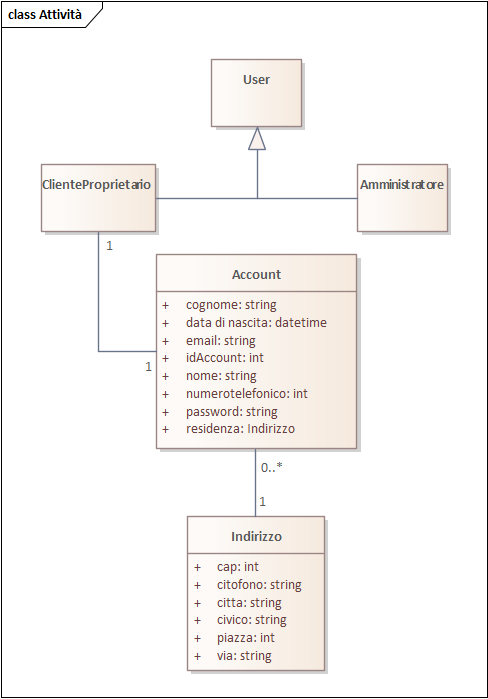


Figura NUMERO: Diagramma di analisi delle classi di Attività

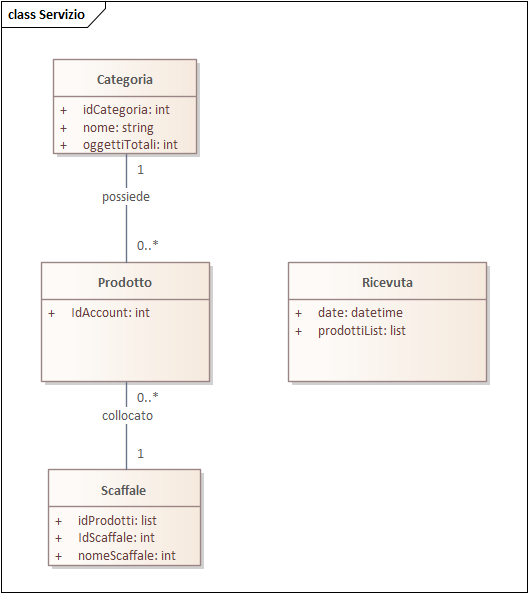


Figura NUMERO: Diagramma di analisi delle classi di Servizio

**3.3 DIAGRAMMI DI SEQUENZA**

**3.3.1 BACKUP**

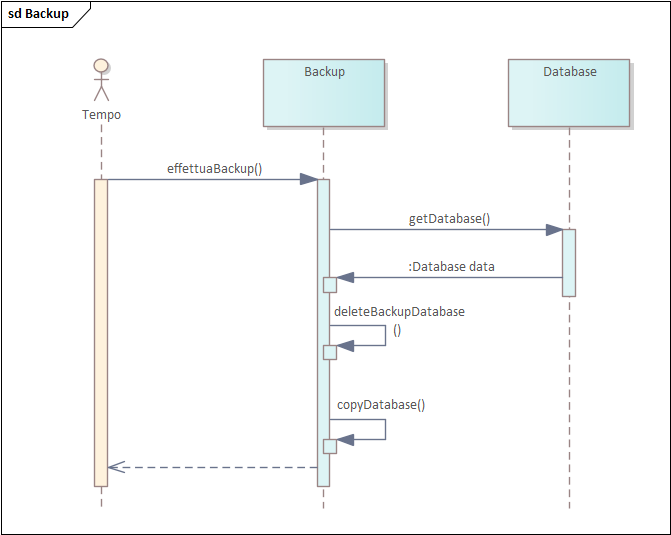


Figura NUMERO: Diagramma di sequenza: Backup

**3.3.2 CONTROLLASTATOPRODOTTO**

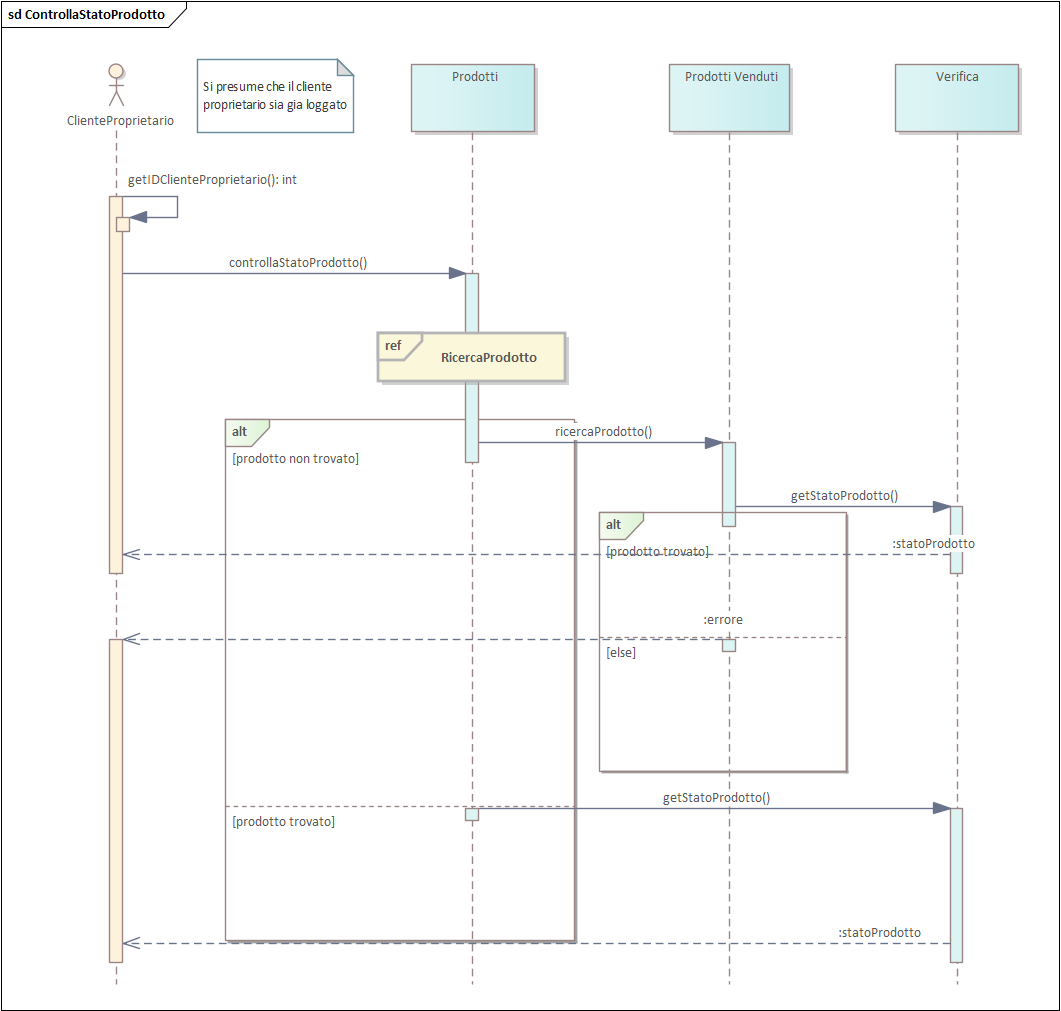


Figura NUMERO: Diagramma di sequenza: ControllaStatoProdotto

**3.3.3 CRUD PRODOTTO**

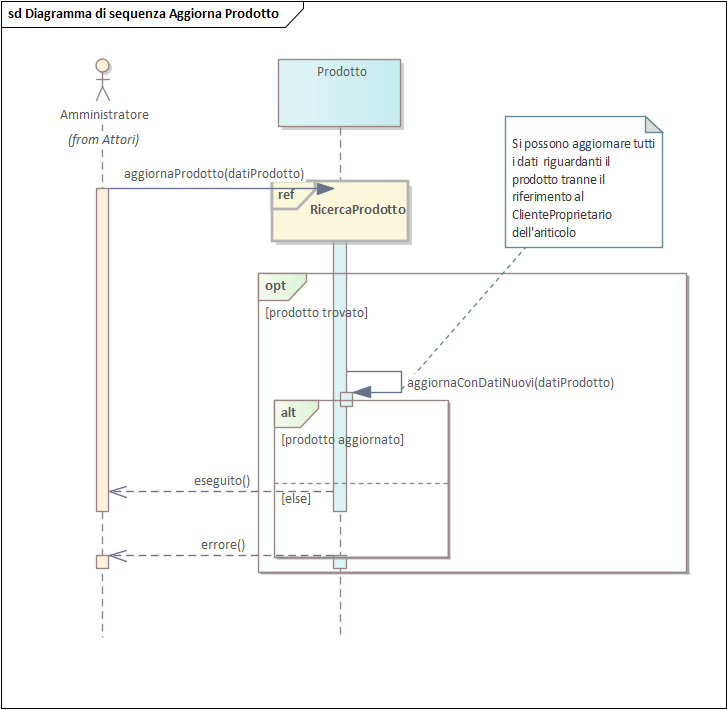


Figura NUMERO: Diagramma di sequenza: Aggiorna Prodotto

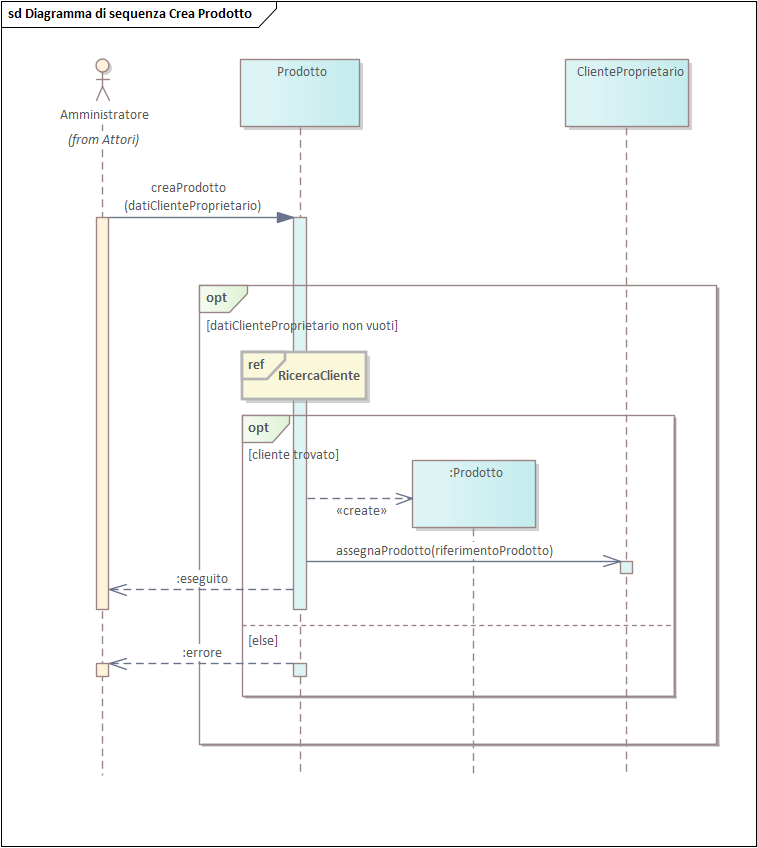


Figura NUMERO: Diagramma di sequenza: Crea Prodotto

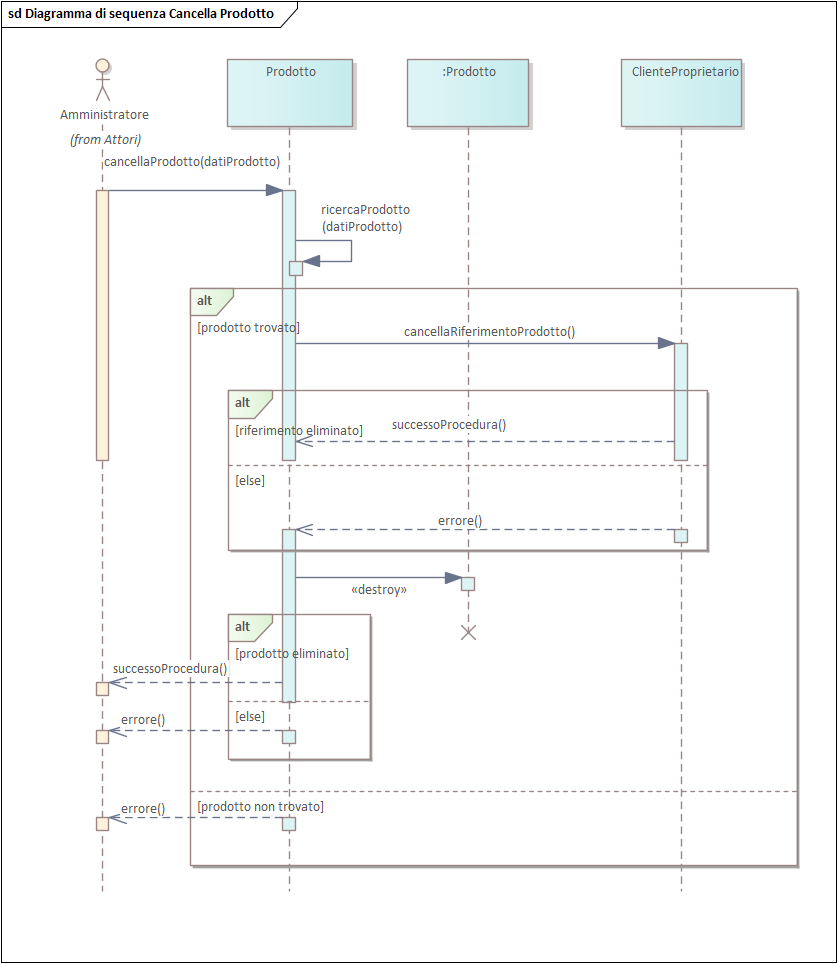


Figura NUMERO: Diagramma di sequenza: Cancella Prodotto

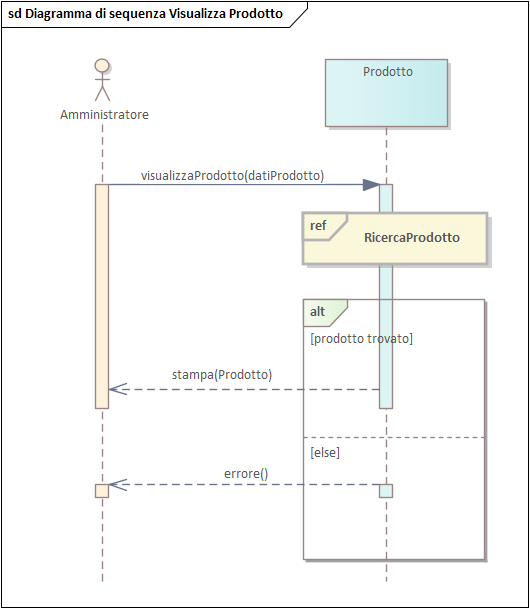


Figura NUMERO: Diagramma di sequenza: Visualizza Prodotto

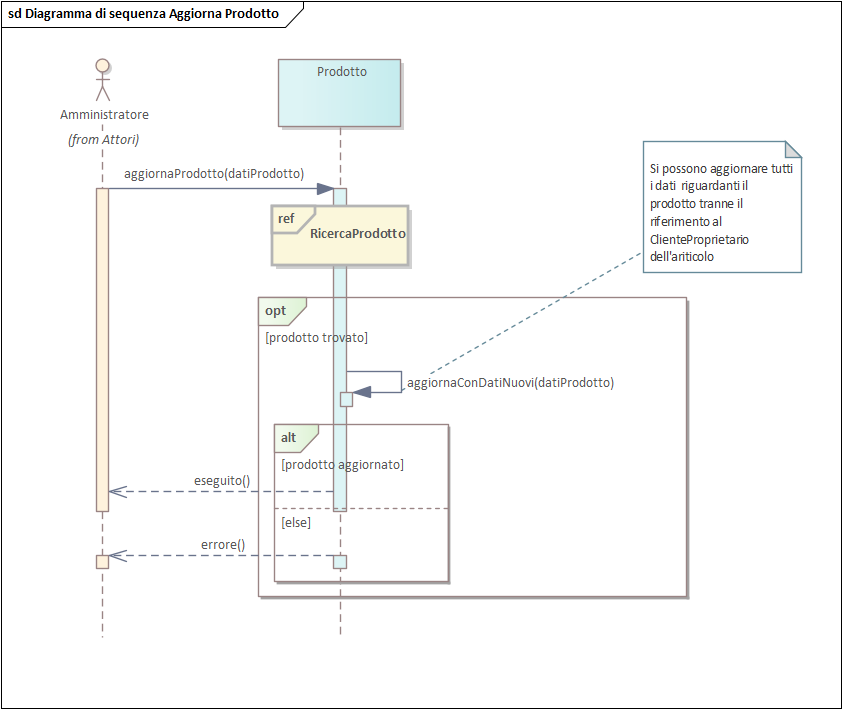


Figura NUMERO: Diagramma di sequenza: Visualizza Prodotto

**3.3.4 LOGIN**

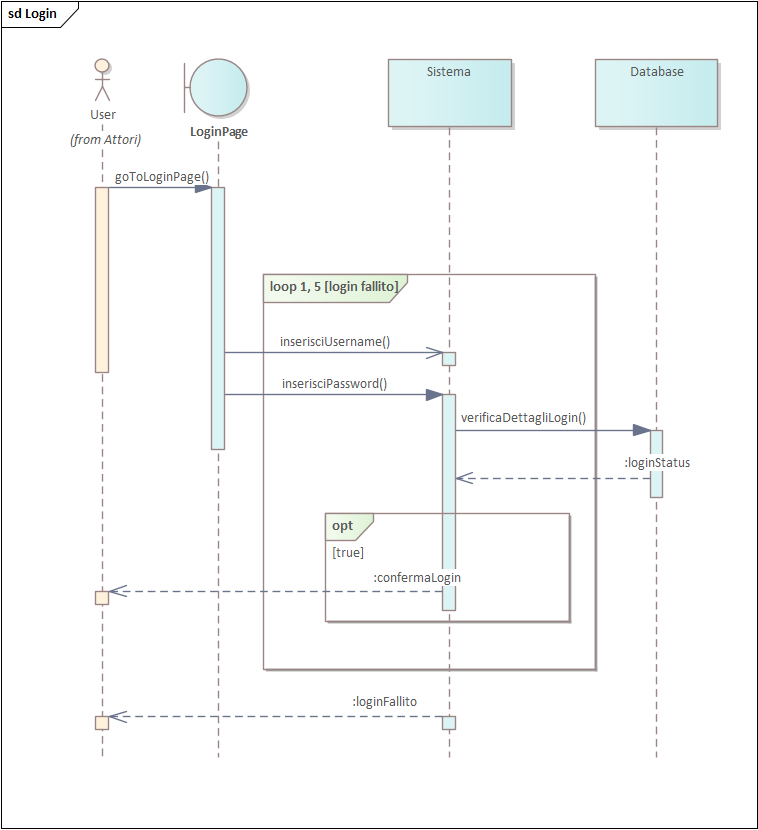


Figura NUMERO: Diagramma di sequenza: Login

**3.3.5 NOTIFICA**

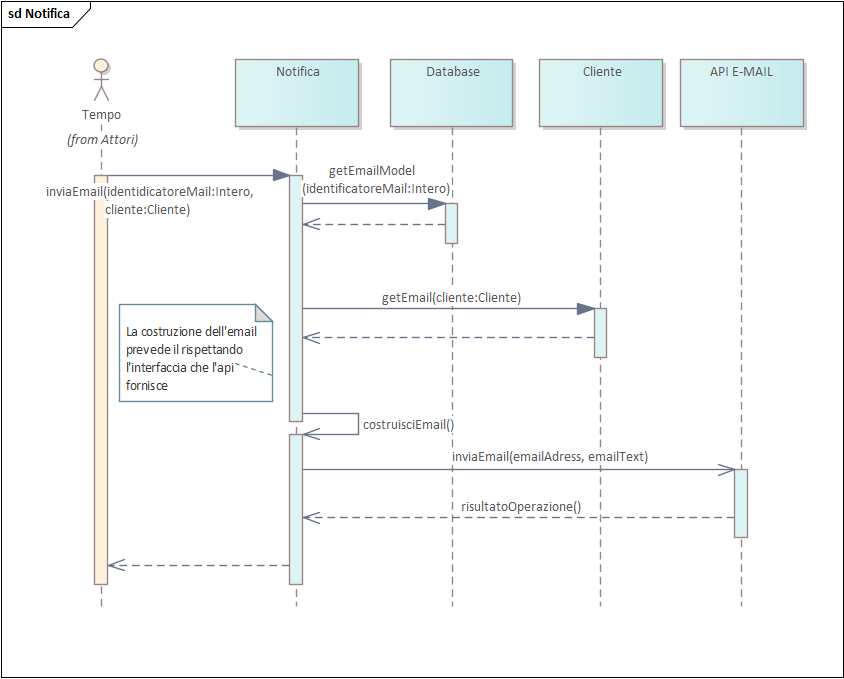


Figura NUMERO: Diagramma di sequenza: Notifica

**3.3.6 RICERCACLIENTE**

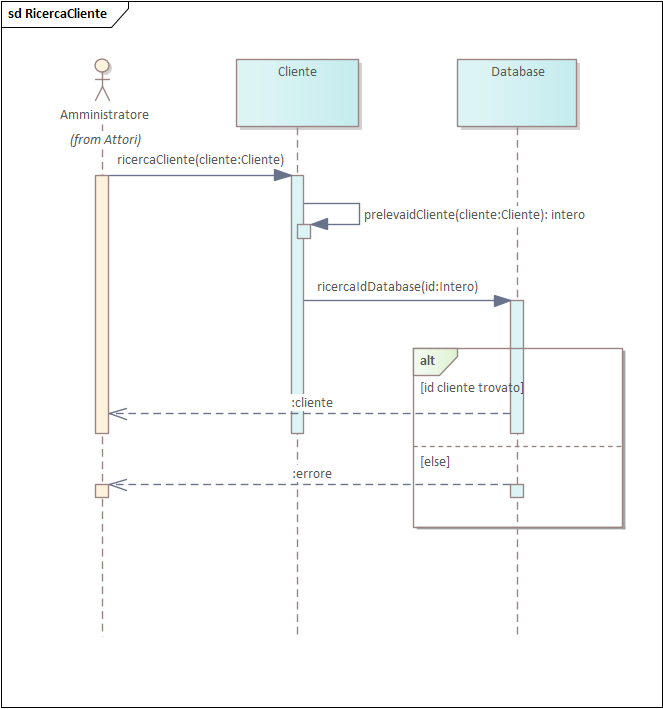


Figura NUMERO: Diagramma di sequenza: Ricerca Cliente

**3.3.7 RICERCAPRODOTTO**

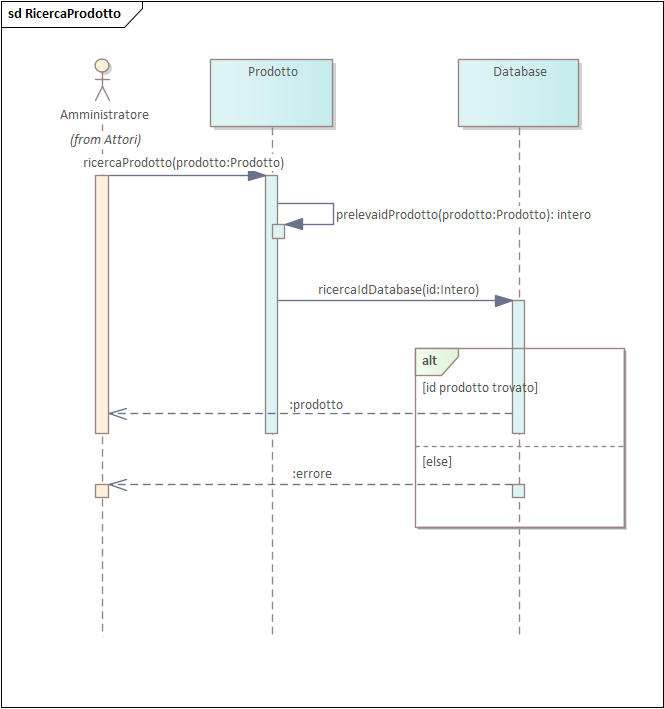


Figura NUMERO: Diagramma di sequenza: RicercaProdotto

**3.3.8 VENDIPRODOTTO**

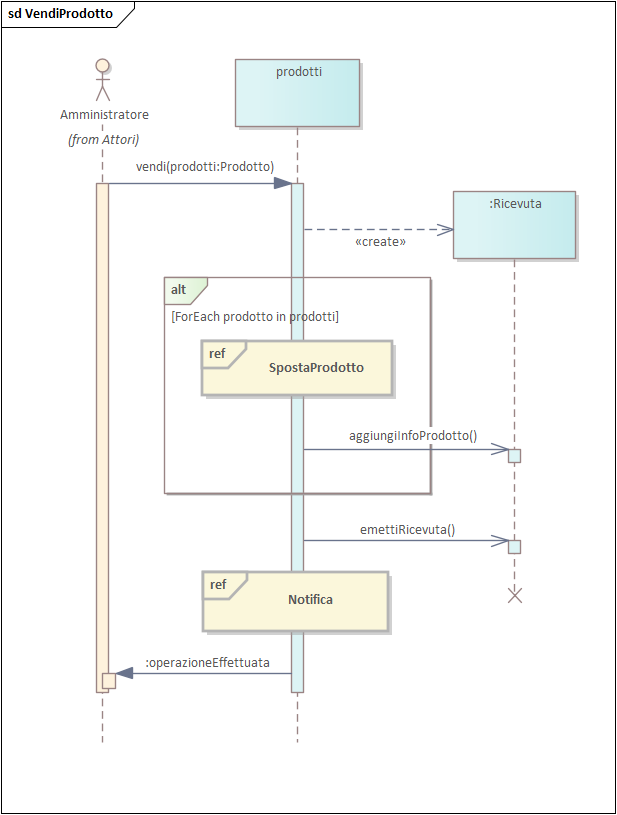


Figura NUMERO: Diagramma di sequenza: VendiProdotto

**3.3.9 VISUALIZZASTATISTICHE**

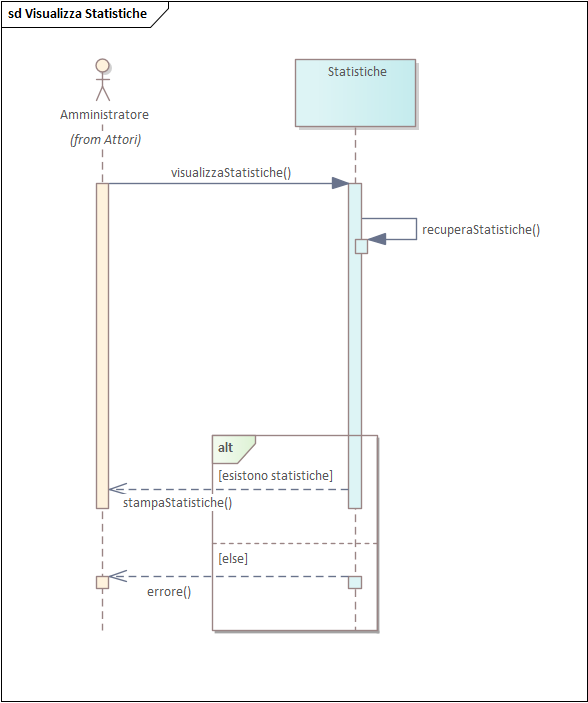


Figura NUMERO: Diagramma di sequenza: Visualizza Statistiche

**4 DIAGRAMMI DI PROGETTAZIONE**

**4.1 DIAGRAMMI DELLE CLASSI DI PROGETTAZIONE**

Immagine che contiene testo

Descrizione generata automaticamente

Figura NUMERO: Diagramma del package di Progettazione

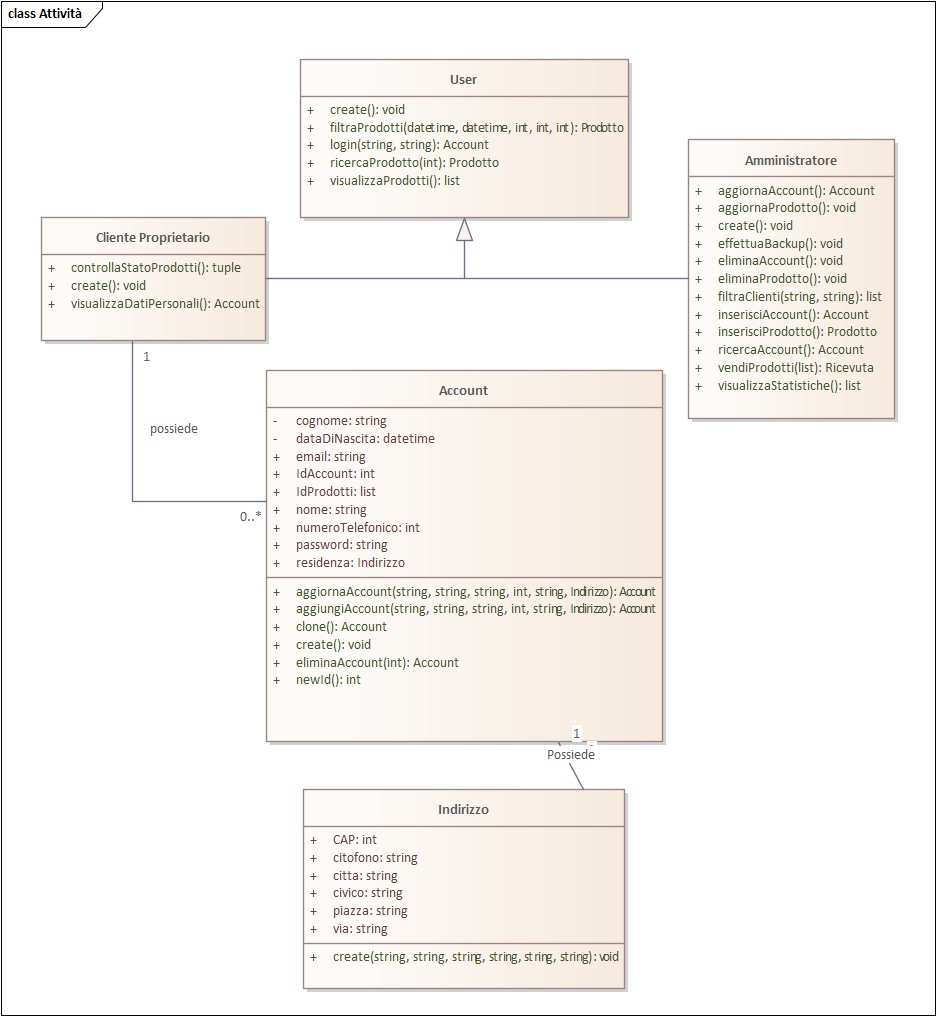


Figura NUMERO: Diagramma delle classi di Attività

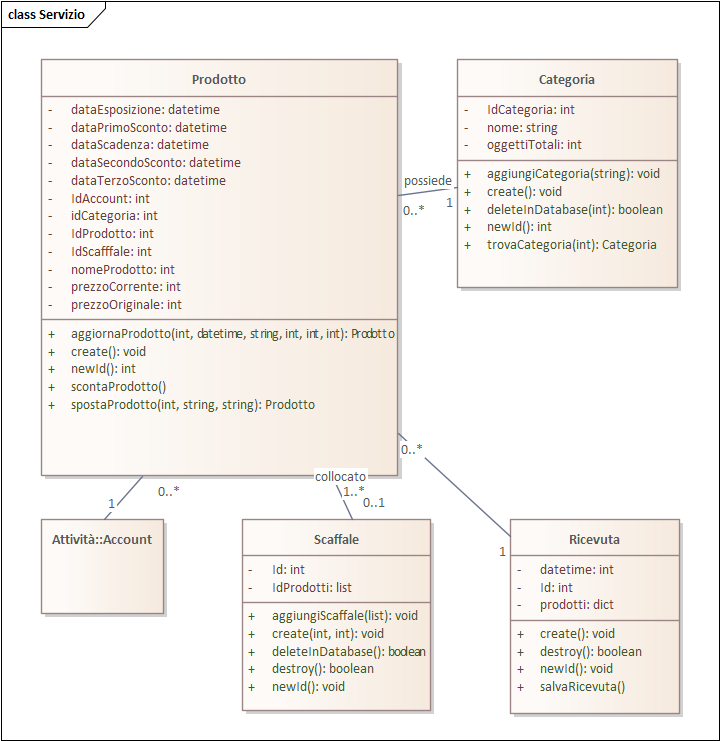


Figura NUMERO: Diagramma delle classi di Servizio

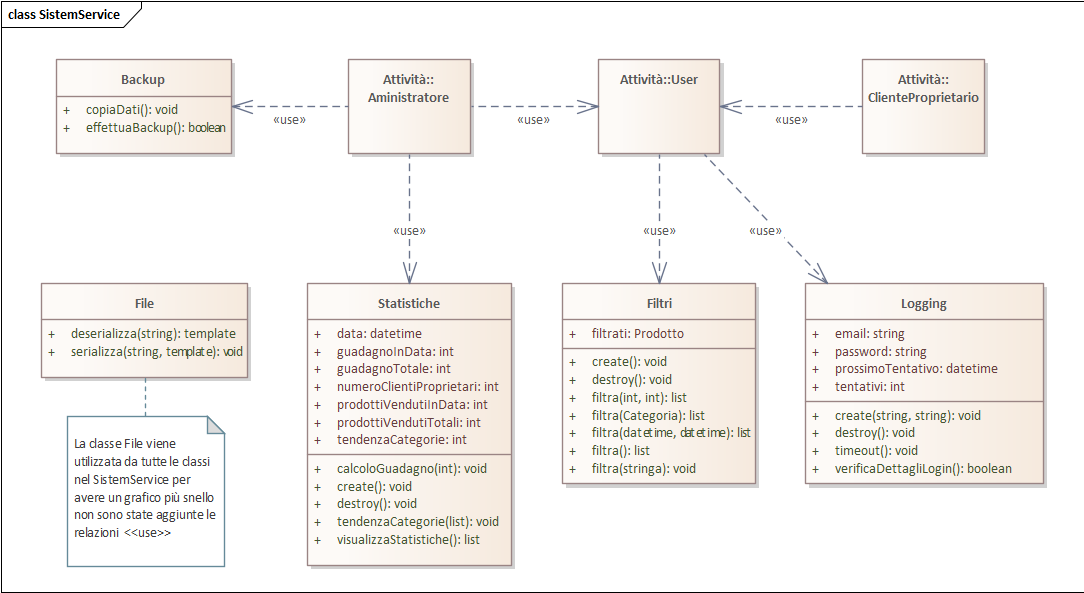


Figura NUMERO: Diagramma delle classi di SistemService

**4.2 DIAGRAMMI DEI COMPONENTI**

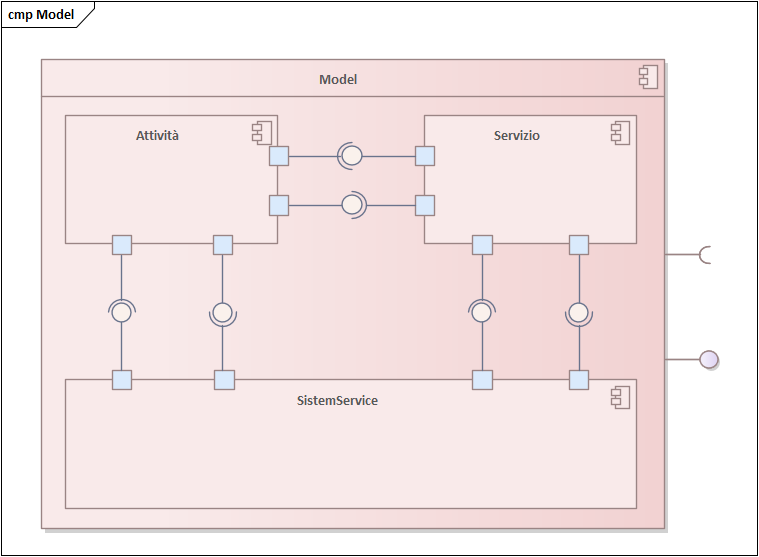


Figura NUMERO: Diagramma dei componenti: Model

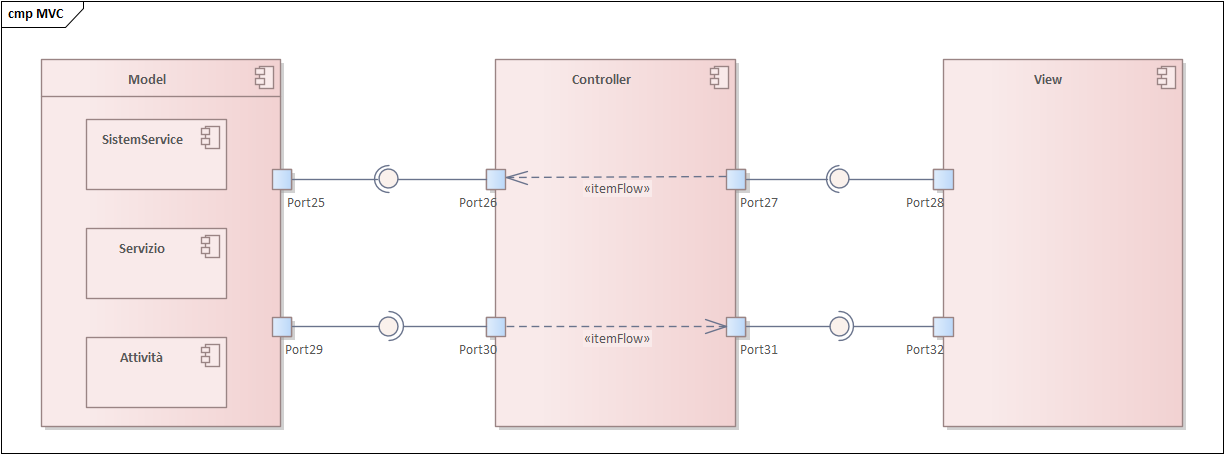


Figura NUMERO: Diagramma dei componenti: MVC

**4.3 DIAGRAMMI DELLE MACCHINE A STATI**

**4.3.1 MACCHINE A STATI: BACKUP**

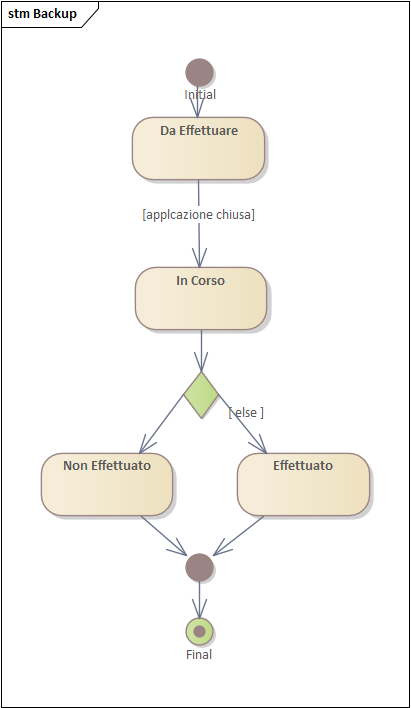


Figura NUMERO: Diagramma della macchina a stati: Backup

**4.3.2 MACCHINE A STATI: CATEGORIA**

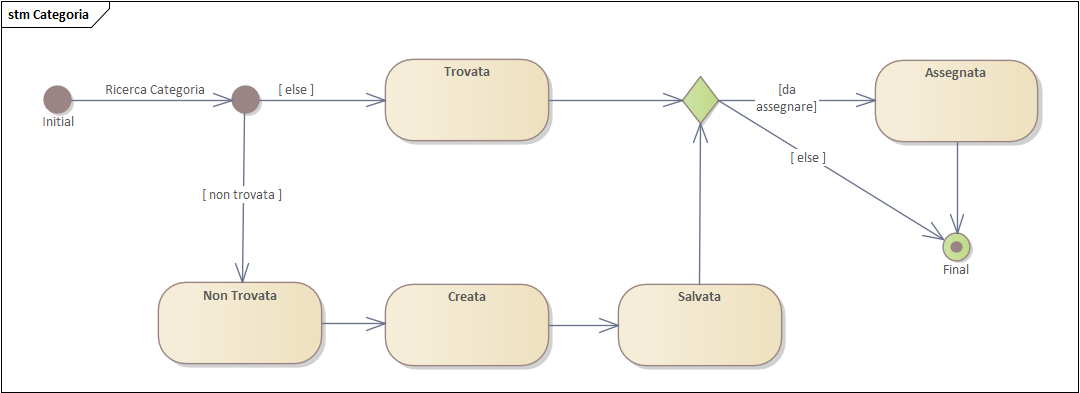


Figura NUMERO: Diagramma della macchina a stati: Categoria

**4.3.3 MACCHINE A STATI: LOGIN**

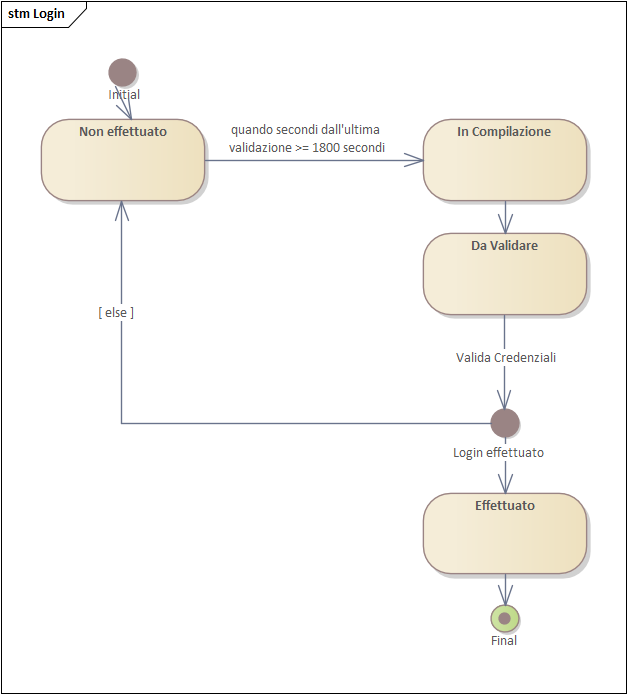


Figura NUMERO: Diagramma della macchina a stati: Login

**4.3.4 MACCHINE A STATI: PRODOTTO**

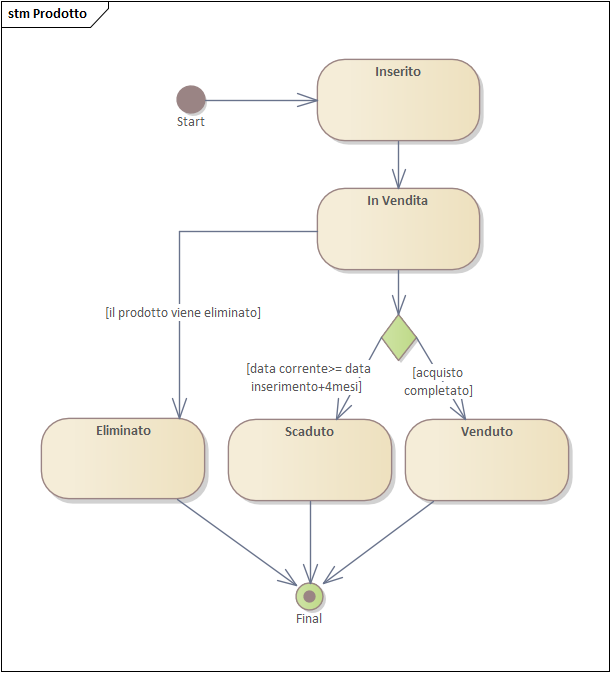


Figura NUMERO: Diagramma della macchina a stati: Prodotto

**4.3.5 MACCHINE A STATI: RICEVUTA**

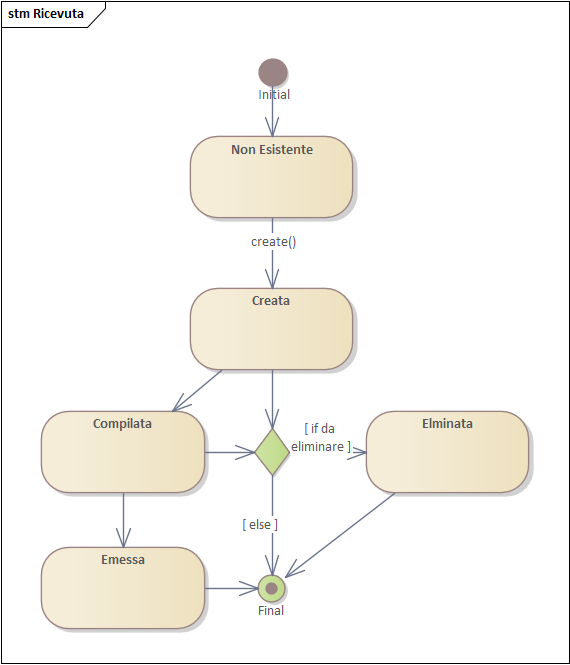


Figura NUMERO: Diagramma della macchina a stati: Ricevuta

**5 IMPLEMENTAZIONE**

**5.1 DIAGRAMMA DI DEPLOYMENT**

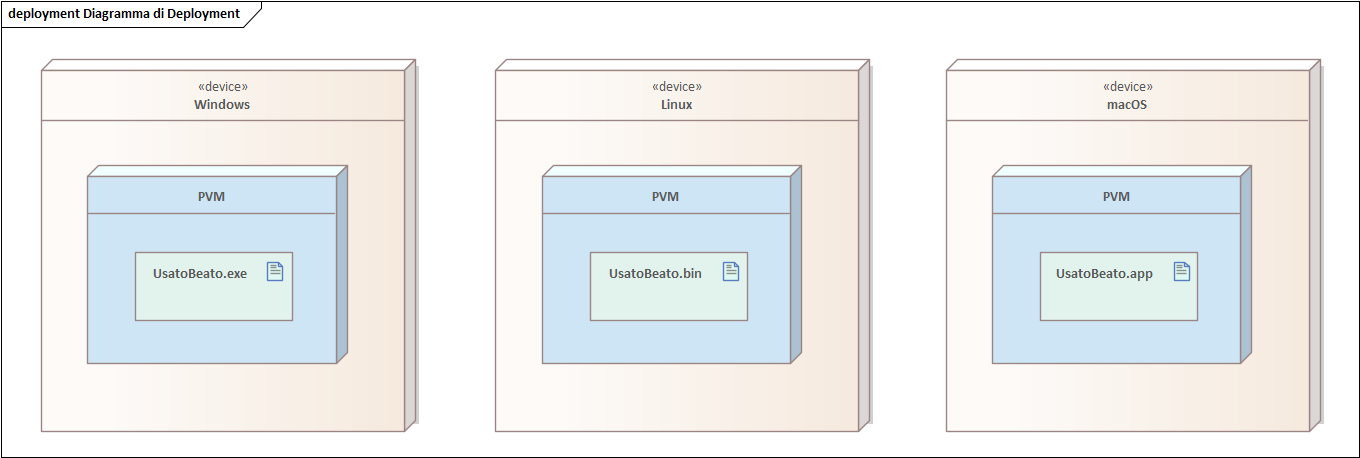


Figura NUMERO: Diagramma di deployment

**6 MOCKUP**

I segueti Mockup sono stati realizzati con il software: **Qt designer**

Il fine è stato rendere la progettazione delle interfacce grafiche dinamiche e quindi evitare una struttura monolitica. Per questo motivo è stato utilizzato per la progettazione dei Mockup il suddetto software. Vi è una 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 versione di UsatoBeato citata in tale documento.