

RAD Requirement Analysis Document

NewDM

|  |  |
| --- | --- |
| Riferimento |  |
| Versione | 0.8 |
| Data | 25/11/2020 |
| Destinatario | Prof.ssa F. Ferrucci |
| Presentato da | Cirillo Franco  Cirillo Luigi  Fusco Ciro  Aiello Vincenzo |
| Approvato da |  |

RAD Requirement Analysis Document

**NewDM**

|  |  |
| --- | --- |
| Riferimento |  |
| Versione | 0.8 |
| Data | 25/11/2020 |
| Destinatario | Prof.ssa F. Ferrucci |
| Presentato da | Cirillo Franco  Cirillo Luigi  Fusco Ciro  Aiello Vincenzo |
| Approvato da |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Cambiamenti | Autori |
| 12/11/2020 | Vr0.1 | Inserimento logo  Completamento introduzione | Luigi Cirillo |
| 15/11/2020 | Vr0.2 | Aggiornamenti colori, font, dimensione del font, come richiesto dalle specifiche | Lugi Cirillo |
| 17/11/2020 | Vr0.3 | Completamento sistema corrente |  |
| 19/11/2020 | Vr0.4 | Completamento Requisiti funzionali e non funzionali |  |
| 21/11/2020 | Vr0.5 | Completamento scenari, usecase |  |
| 22/11/2020 | Vr0.6 | Completamento object diagram |  |
| 23/11/2020 | Vr0.7 | Completamento sequence diagram |  |
| 25/11/2020 | Vr0.8 | Completamento mock-ups |  |

RevisionHistory

Sommario

1. [1. Introduzione 5](#_Toc57565255)

[1.1 Scopo del sistema 5](#_Toc57565256)

[1.2 Ambito del sistema 5](#_Toc57565257)

[1.3 Obiettivi e criteri di successo del sistema 5](#_Toc57565258)

[1.4 Definizioni, acronimi e abbreviazioni 5](#_Toc57565259)

[1.5 Riferimenti 5](#_Toc57565260)

[1.6 Panoramica 6](#_Toc57565261)

1. [2. Sistema corrente 6](#_Toc57565262)
2. [3. Sistema proposto 6](#_Toc57565263)

[3.1 Panoramica 6](#_Toc57565264)

[3.2 Requisiti funzionali 7](#_Toc57565265)

[3.3 Requisiti non funzionali 8](#_Toc57565266)

[3.3.1 Usabilità 8](#_Toc57565267)

[3.3.2 Affidabilità 8](#_Toc57565268)

[3.3.3 Prestazioni 9](#_Toc57565269)

[3.3.4 Supportabilità 9](#_Toc57565270)

[3.3.5 Implementazione 9](#_Toc57565271)

[3.3.6 Interfaccia 9](#_Toc57565272)

[3.3.7 Packaging 9](#_Toc57565273)

[3.3.8 Legali 9](#_Toc57565274)

[3.4 Modello di sistema 9](#_Toc57565275)

[3.4.1 Scenari 9](#_Toc57565276)

[3.4.2 Use Case Diagram 13](#_Toc57565277)

[3.4.2 Use Case 15](#_Toc57565278)

[3.4.4 Modello ad oggetti 21](#_Toc57565279)

[3.4.4.1 Object Diagram 21](#_Toc57565280)

[3.4.5 Modelli dinamici 26](#_Toc57565281)

[3.4.5.1 Sequence Diagram 26](#_Toc57565282)

[3.4.5.2 Statechart 31](#_Toc57565283)

[3.4.6 Interfaccia utente e mock-ups 32](#_Toc57565284)

[3.4.6.1 Mock\_Ups 32](#_Toc57565285)

[3.4.6.1 Navigation path 51](#_Toc57565286)

1. [4.Glossario 51](#_Toc57565287)

# 1. Introduzione

## 1.1 Scopo del sistema

La catena di supermercati intende incrementare la capacità di gestione dei magazzini, controllando in tempo reale la quantità di pezzi in stock di ogni prodotto e riuscire così a rifornire il magazzino con sufficiente anticipo. Intende riformare la fase di emissione degli scontrini per adeguarsi alla nuova normativa ministeriale. Intende altresì digitalizzare la fase di gestione post-vendita al fine di migliorare la qualità del servizio offerto.

## 1.2 Ambito del sistema

Il sistema una volta consegnato sarà di grande aiuto nella centralizzazione di tutte quelle attività che ogni giorno i dipendenti dei vari negozi eseguono per permettere ai clienti di usufruire del miglior servizio possibile, ad esempio:

* Gestione delle scorte
* Emissione degli scontrini
* Assistenza al cliente

Non è prevista la gestione dei volantini né la carta fedeltà per la suddetta catena di supermercati.

NewDM non gestirà nemmeno gli orari dei lavoro dei dipendenti, né quanto meno la loro paga.

## 1.3 Obiettivi e criteri di successo del sistema

L’elenco successivo mostra gli obiettivi primari del progetto:

1. Fornire uno strumento la gestione e l’informatizzazione dei punti vendita della catena di supermercati
2. Velocizzare il processo di pagamento attraverso una migliore gestione della cassa
3. Ottimizzare i tempi relativi all’aggiornamento dell’inventario del magazzino
4. Fornire uno strumento che aiuti nella scelta delle quantità di prodotti da acquistare
5. Rendere più facilmente accessibili i dati dei da parte dei dipendenti
6. Automatizzare le procedure successive alle richieste di assistenza da parte dei clienti

Criteri di successo:

1. Branch coverage dei casi di test: almeno 75%
2. Buona manutenibilità e di integrabilità

## 1.4 Definizioni, acronimi e abbreviazioni

- stock: un determinato quantitativo di prodotti

- device: dispositivo su cui è installato il software in oggetto

- modulo di IA: modulo di intelligenza artificiale per la previsione delle quantità di rifornimento consigliate

- RF: Requisito Funzionale

- NFR: Requisito Non Funzionale

- “det.” e “determin.”: determina e determinano

- sist. = Sistema

## 1.5 Riferimenti

Libro:

-- Object-Oriented Software Engineering (Using UML, Patterns, and Java) Third Edition

Autori:

-- Bernd Bruegge & Allen H. Dutoit

## 1.6 Panoramica

Al secondo punto del documento verrà presentato il sistema corrente. Al terzo punto verrà presentato il sistema proposto con i relativi requisiti funzionali e non funzionali. Attraverso scenari e use case verrà stabilito chi sono gli attori del sistema e come gli attori individuati interagiscono con il sistema. Verranno poi presentati mock-up dell’interfaccia utente ed i path navigazionali per l’applicazione proposta.

# 2. Sistema corrente

La catena di supermercati “I Giardinetti” utilizza un sistema poco informatizzato. L’acquisto dei prodotti, da parte del cliente, non è strettamente legato all’aggiornamento della quantità degli stock rimanenti all’interno del magazzino. Per questo motivo il magazziniere dovrà gestire manualmente questa situazione, eventualmente compilando un inventario ogni volta che l’occasione lo richiede. L’addetto al magazzino dovrà, inoltre, gestire la richiesta d’acquisto di nuovi prodotti basandosi sul conteggio e il calcolo delle scorte rimanenti e sui dati di vendita. La catena dei supermercati “I Giardinetti” non offre un’assistenza post-vendita per i clienti, anche se molto richiesta da questi ultimi.

Abbiamo ritenuto opportuno sviluppare un activity diagram relativi alle operazioni svolte dal punto vendita del supermercato per meglio percepire la differenza tra quello che è il sistema che proponiamo e quello che è il sistema attuale.

Pertanto, abbiamo ritenuto necessario realizzare un activity diagram relativo alla procedura per l’emissione di uno scontrino e alla richiesta di una nuova fornitura di prodotti.



# 3. Sistema proposto

## 3.1 Panoramica

Il sistema da noi proposto è un applicazione desktop, per informatizzare una catena di supermercati in modo da migliorare la gestione dei vari punti vendita. L’applicativo sarà disponibile su più dispositivi utili ai dipendenti dell’azienda stessa, sempre però garantendo un’integrità dei dati su tutti i device dello stesso punto vendita.

Gli utenti saranno i dipendenti che lavorano per il punto vendita in questione:

* Magazziniere
* Cassiere
* Addetto assistenza

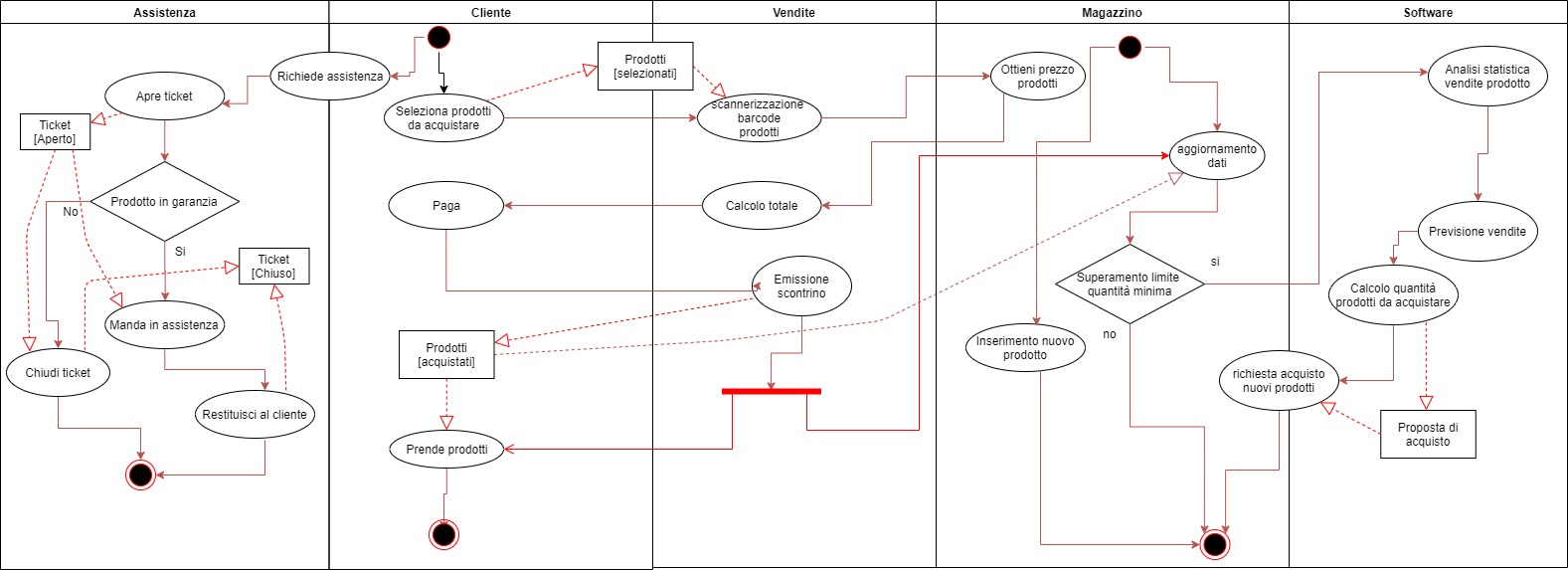
Gli utenti per accedere alle proprie mansioni dovranno effettuare un’autenticazione relativa alla propria area di lavoro.

Il cassiere potrà scansionare i prodotti selezionati dal cliente, calcolare il totale della spesa e stampare lo scontrino dopo aver ricevuto il pagamento. La stampa dello scontrino aggiornerà in automatico le quantità rimanenti in magazzino.

Il magazziniere potrà inserire stock di prodotti appena arrivati e potrà registrare anche un nuovo prodotto. Potrebbe altresì modificare il prezzo di un prodotto oppure fare una richiesta di rifornimento facendosi consigliare da una previsione della quantità di prodotto da acquistare.

L’addetto assistenza, al fine di soddisfare le richieste di un cliente, potrà aprire un nuovo ticket per l’assistenza relativa ad un prodotto, validarne degli aspetti ed eventualmente salvarlo.

Al fine di poter cogliere al meglio gli effetti del sistema che proponiamo, presentiamo un activity diagram relativo alle funzionalità principali messe a disposizione dalla piattaforma.



## 3.2 Requisiti funzionali

Di seguito i requisiti funzionali in cui la priorità va da 1 bassa a 3 alta

|  |  |  |
| --- | --- | --- |
| **Req.** | **Priorità** | **Descrizione** |
| RF\_1 | 3 | Il commesso deve essere in grado di inserire manualmente il codice di un prodotto |
| RF\_2 | 2 | Il commesso deve poter inserire manualmente la quantità di un prodotto |
| RF\_3 | 3 | Il commesso deve richiedere al sistema di emettere lo scontrino |
| RF\_4 | 2 | Il commesso deve poter modificare la lista prodotti |
| RF\_5 | 1 | Il commesso deve inserire il totale dato dal cliente |
| RF\_6 | 3 | Il commesso deve poter dare il resto |
| RF\_7 | 3 | Ogni dipendente per accedere alle funzionalità della propria area di lavoro deve necessariamente inserire nome utente e password |
| RF\_8 | 1 | Il cliente dovrebbe poter vedere il prezzo dell’ultimo prodotto “scannerizzato” |
| RF\_9 | 3 | Il cliente deve poter scegliere liberamente il metodo di pagamento tra quelli accettati |
| RF\_10 | 2 | Il cliente deve poter richiedere assistenza per ogni prodotto acquistato |
| RF\_11 | 3 | Il magazziniere deve poter inserire un nuovo prodotto |
| RF\_12 | 3 | Il magazziniere deve poter modificare la quantità di un prodotto |
| RF\_13 | 1 | Il magazziniere dovrebbe poter vedere la data di scadenza dei prodotti |
| RF\_14 | 3 | Il magazziniere deve poter modificare il prezzo |
| RF\_15 | 3 | Il magazziniere deve poter richiedere rifornimenti |
| RF\_16 | 2 | L'addetto all’assistenza deve poter aprire un ticket per ogni prodotto da inviare in assistenza |
| RF\_17 | 3 | Il modulo di IA deve essere in grado di fare una previsione sulla quantità di prodotti da acquistare |
| RF\_18 | 3 | Il sistema deve aggiornare la quantità dei prodotti in magazzino ogni volta che viene effettuato un acquisto |
| RF\_19 | 1 | L'addetto all’assistenza deve poter chiudere un ticket dopo la consegna del prodotto al cliente in questione |

## 3.3 Requisiti non funzionali

### 3.3.1 Usabilità

* RNF-U1: Il sistema deve permettere di effettuare le operazioni in modo semplice e immediato.
* RNF-U2: Il sistema deve consentire ai vari utenti di utilizzare le funzionalità offerte dal sistema senza consultare la documentazione, grazie all’utilizzo di etichette e pulsanti non ambigui, menù chiari e di facile gestione.
* RNF-U3: Il sistema deve avere una documentazione minimale ma esaustiva.

### 3.3.2 Affidabilità

* RNF-A1: Il sistema deve garantire l’affidabilità dei servizi proposti. Il prodotto software sarà sviluppato in modo tale da controllare accuratamente le informazioni inserite in input dagli utenti.
* RNF-A2: Il sistema dovrà evitare il verificarsi di più di un fault al giorno, in modo da garantire l’efficienza del servizio, ma soprattutto dovrà evitare ogni perdita di dati registrati.
* RNF-A3: Il sistema dovrà fornire un metodo di autenticazione sicuro in modo che i dati siano protetti da accessi fraudolenti.

### 3.3.3 Prestazioni

* RNF-P1: Il sistema dovrà garantire che la scannerizzazione e la visualizzazione del prezzo del prodotto devono avvenire in al più sec. 2, in modo da evitare il crearsi di lunghe file alla cassa.
* RNF-P2: Il sistema dovrà garantire che il calcolo totale e l’emissione dello scontrino devono avvenire in al più sec. 3.
* RNF-P3: Il sistema dovrà garantire che l’avanzamento del ticket per l’assistenza di un prodotto deve avvenire entro h. 24 al fine di soddisfare le richieste nel minor tempo possibile.
* RNF-P4: Il sistema deve essere sempre disponibile durante l’orario di lavoro (6am to 10pm).

### 3.3.4 Supportabilità

* RNF-S1: Il sistema prodotto dev’essere facilmente comprensibile da sviluppatori terzi che vorranno estenderne le funzionalità o risolverne i problemi
* RNF-S2: Il sistema prodotto deve offrire una buona modificabilità, in modo da poter modificare le funzionalità esistenti.
* RNF-S3: Il sistema prodotto deve offrire una buona estensibilità, in modo da poter aggiungere nuove funzionalità al sistema

### 3.3.5 Implementazione

Il sistema è stato concepito come un’applicazione Desktop in ambiente Windows che fornisce agli utenti interfacce e funzionalità a seconda della loro mansione.

### 3.3.6 Interfaccia

Il sistema deve interfacciarsi con l’utente attraverso la visualizzazione di pulsanti, menu e indicatori, in modo tale da avere una visione chiara della situazione e delle possibili azioni che può svolgere.

### 3.3.7 Packaging

Il sistema verrà installato su tutte le macchine da un team specializzato con adeguate conoscenze relative al funzionamento del prodotto.

### 3.3.8 Legali

N/D

## 3.4 Modello di sistema

### 3.4.1 Scenari

|  |  |  |
| --- | --- | --- |
| NOME SCENARIO | SC\_3: AcquistoProdotto | |
| ATTORI | Alberto:Cliente, Pasquale:Commesso. | |
| DESCRIZIONE | Il Cliente Alberto vuole acquistare dei prodotti nella filiale 1 del Gruppo “I Giardinetti” | |
| FLUSSO DEGLI EVENTI | *Attori* | *Sistema* |
| Alberto si avvicina alla cassa e deposita un barattolo di nutella, una confezione di fesa di tacchino, il petto di pollo e il pane sul nastro trasportatore |  |
| Pasquale prende singolarmente ogni prodotto e scannerizza il suo codice |  |
|  | Il sistema lo acquisisce e lo inserisce nella lista acquisti. |
| Pasquale, una volta terminati i prodotti, richiede al sistema di mostrare il totale |  |
|  | Il sistema mostra 25.50€ |
| Pasquale lo comunica ad Alberto |  |
| Alberto dà a Pasquale 30€ |  |
| Pasquale comunica al sistema l’importo versato da Alberto |  |
|  | Il sistema crea lo scontrino con la lista acquisti e comunica a Pasquale il resto da dare ad Alberto |
| Pasquale dà il resto ad Alberto insieme allo scontrino |  |
|  | Il sistema, emesso lo scontrino, procede con l’aggiornare i prodotti presenti nel magazzino. |

|  |  |  |
| --- | --- | --- |
| NOME SCENARIO | SC\_12: InserimentoProdottiRegistrati | |
| ATTORI | Giovanni:Magazziniere | |
| DESCRIZIONE | Il magazziniere Giovanni vuole inserire nel sistema uno stock di prodotti appena arrivati: acqua Lete 1.5lt, Coca Cola 33cl, Fanta 1.5lt. | |
| FLUSSO DEGLI EVENTI | *Attore* | *Sistema* |
| Giovanni accede all’area per l’inserimento di prodotti appena consegnati |  |
|  | Il sistema mostra un form per l’inserimento dei dati del rifornimento del prodotto |
| Giovanni inserisce il codice del prodotto acqua Lete cod: 80412021 e la quantità 100pz |  |
|  | Il sistema mostra di nuovo i dati e richiede conferma |
| Giovanni conferma e ripete la procedura per gli altri prodotti: Coca Cola cod: 5000112557664 75pz, Fanta cod: 5449000005090 50 pz. |  |
|  | Il sistema provvede ad aggiornare le quantità dei prodotti rimanenti. |

|  |  |  |
| --- | --- | --- |
| NOME SCENARIO | SC\_16: RichiediAssistenza | |
| ATTORI | Alberto:Cliente, Saverio:AddettoAssistenza | |
| DESCRIZIONE | Il cliente Alberto dopo aver acquistato un computer nel nostro punto vendita di Salerno ha bisogno di assistenza in quanto non si accende più. | |
| FLUSSO DEGLI EVENTI | *Attori* | *Sistema* |
| Alberto si reca presso il punto vendita in cui ha effettuato l’acquisto, richiede assistenza e consegna il computer e il relativo scontrino a Saverio |  |
| Saverio richiede l’apertura del ticket per la richiesta di assistenza |  |
|  | Il sistema mostra un form per l’inserimento dei dati anagrafici del cliente |
| Saverio inserisce nel ticket tutti i dati anagrafici di Alberto e in seguito controlla lo scontrino e osserva che la data di acquisto del computer risale al 10 ottobre 2019, quindi meno di 2 anni dalla data odierna. |  |
| Saverio a questo punto chiede ad Alberto una descrizione dettagliata del problema e fa una richiesta al sistema per inserirla |  |
|  | Il sistema mostra un form per l’inserimento del problema |
| Saverio inserisce il problema e conferma |  |
|  | Il sistema stampa una copia del ticket |
| Saverio in fine consegna ad Alberto una copia del ticket da utilizzare al momento del ritiro. |  |

|  |  |  |
| --- | --- | --- |
| NOME SCENARIO | SC\_15: Ordina | |
| ATTORI | Giovanni:Magazziniere | |
| DESCRIZIONE | Il magazziniere Giovanni riceve dal sistema una notifica che la quantità minima del prodotto Coca Cola 1.5L è stata raggiunta. | |
| FLUSSO DEGLI EVENTI | *Attori* | *Sistema* |
| Giovanni accede all’ area dedicata per ottenere le previsioni di acquisto per i prodotti |  |
|  | Il sistema mostra il form per l’inserimento del codice del prodotto |
| Giovanni inserisce il codice della Coca Cola 1.5L: cod: 5000112557664 |  |
|  | Il sistema acquisisce il codice e propone il quantitativo consigliato |
| Giovanni riceve il risultato e decide di richiedere l’acquisto delle quantità proposte |  |
|  | Il sistema prende in carica la richiesta |

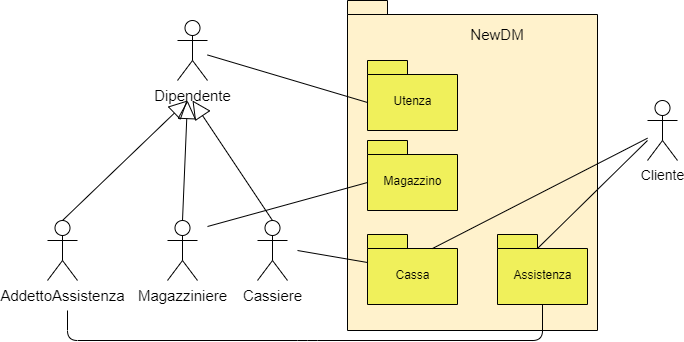
|  |  |  |
| --- | --- | --- |
| NOME SCENARIO | SC\_14: ModificaPrezzoProdotto | |
| ATTORI | Giovanni:Magazziniere | |
| DESCRIZIONE | Il magazziniere Giovanni deve aggiornare il prezzo del computer “Lenovo yoga slim 7 14are05”. | |
| FLUSSO DEGLI EVENTI | *Attori* | *Sistema* |
| Giovanni accede all’area per la modifica delle informazioni sul prodotto |  |
|  | Il sistema mostra un form per l’inserimento del codice prodotto |
| Giovanni inserisce il codice del computer |  |
|  | Il sistema mostra un form modificabile contenente tutte le informazioni sul computer “Lenovo yoga slim 7 14are05”. |
| Giovanni aggiorna il prezzo del prodotto a 1000€ e conferma la modifica. |  |
|  | Il sistema provvede ad aggiornare le informazioni del prodotto sul database |

|  |  |  |
| --- | --- | --- |
| NOME SCENARIO | SC\_11: InserimentoNuovoProdotto | |
| ATTORI | Giovanni:Magazziniere | |
| DESCRIZIONE | Il magazziniere Giovanni deve aggiungere il nuovo prodotto “Zuppa di fagioli” che da domani verrà venduto presso il punto vendita di Scafati. | |
| FLUSSO DEGLI EVENTI | *Attori* | *Sistema* |
| Giovanni accede all’area per l’inserimento dei nuovi prodotti |  |
|  | Il sistema mostra un form per l’inserimento dei dati del prodotto |
| Giovanni inserisce il nome del prodotto “zuppa di fagioli Trinità” il codice del prodotto cod: 80859632, la quantità 100pz, e il prezzo per confezione di 2.50€ |  |
|  | Il sistema mostra di nuovo i dati e richiede conferma |
| Giovanni conferma |  |
|  | Il sistema provvede a registrare tutte le informazioni inserite |

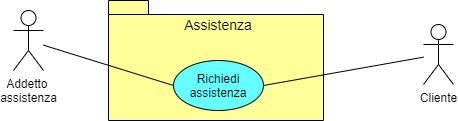
|  |  |  |
| --- | --- | --- |
| NOME SCENARIO | SC\_7: Login | |
| ATTORI | Giovanni:Magazziniere | |
| DESCRIZIONE | Il magazziniere Giovanni vuole accedere alla sua area riservata relativa alla sua sezione di lavoro. | |
| FLUSSO DEGLI EVENTI | *Attori* | *Sistema* |
| Giovanni raggiunge la schermata relativa del sistema |  |
|  | Il sistema mostra 3 possibili aree di lavoro: Magazzino, cassa, assistenza |
| Giovanni seleziona magazzino |  |
|  | Il sistema mostra una schermata per l’inserimento delle credenziali |
| Giovanni inserisce come nome utente: “Giovanni99” e come password: “trottolino\_amoroso” |  |
|  | Il sistema verifica le credenziali e notifica all’utente che l’operazione di accesso è stata effettuata con successo |
|  | Il sistema porta l’utente alla propria area personale |

### 3.4.2 Use Case Diagram

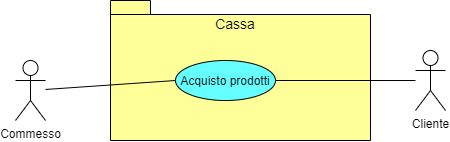
UCD\_Sistema



UCD\_Assistenza



UCD\_Cassa



UCD\_Magazzino



UCD\_Utenza



### 3.4.2 Use Case

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Identificativo  *UC\_1* | | | | *Acquisto prodotti* | *Data* | | *03/10/2020* |
| *Vers.* | *0.01.003* | |
| *Autore* | *Vincenzo Aiello* | |
| Descrizione | | | | *Il Cliente vuole acquistare dei prodotti* | | | |
| Attore Principale | | | | Cliente  Acquista i prodotti in un qualsiasi punto vendita della catena i giardinetti | | | |
| Attori secondari | | | | Commesso  Utilizza il sistema per registrare l’acquisto dei prodotti | | | |
| Entry Condition | | | | Il cliente seleziona i prodotti da acquistare | | | |
| Exit condition  On success | | | | Il commesso consegna lo scontrino e i prodotti al cliente | | | |
| Exit condition  On failure | | | | Il cliente non riceve lo scontrino e i prodotti | | | |
| Rilevanza/User Priority | | | | Elevata | | | |
| Frequenza stimata | | | | 700/giorno | | | |
| Extension point | | | | NA | | | |
| Generalization of | | | | NA | | | |
| Flusso di Eventi Principale/Main Scenario | | | | | | | |
| 1 | Cliente: | | deposita i prodotti da acquistare alla cassa | | | | |
| 2 | Commesso: | | prende singolarmente ogni prodotto e lo scannerizza | | | | |
| 3 | Sistema: | | acquisisce i dati di ogni prodotto e lo inserisce nella lista acquisti | | | | |
| 4 | Commesso: | | richiede al sistema di mostrare il totale | | | | |
| 5 | Sistema: | | mostra il totale | | | | |
| 6 | Commesso: | | comunica al cliente il prezzo totale | | | | |
| 7 | Cliente: | | effettua il pagamento | | | | |
| 8 | Commesso: | | inserisce somma versata dal cliente | | | | |
| 9 | Sistema: | | mostra il resto da dare al cliente | | | | |
| 10 | Sistema: | | emette lo scontrino | | | | |
| 11 | Commesso: | | consegna il resto, lo scontrino e i prodotti | | | | |
| 12 | Cliente: | | ritira il resto, i prodotti e lo scontrino | | | | |
| 13 | Sistema: | | aggiorna i prodotti presenti nel magazzino | | | | |
| I Scenario/Flusso di eventi Alternativo: il prodotto non può essere scannerizzato | | | | | | | |
| 2.1 | | Commesso: | | digita manualmente il codice. | | | |
|  | | | |  | | | |
| I Scenario/Flusso di eventi di ERRORE: codice inserito non corretto | | | | | | | |
| 3.1 | | Sistema: | | mostra una notifica di errore. | | | |
| 3.2 | | Sistema: | | mostra di nuovo il form per l’inserimento del codice | | | |
| 3.3 | | Commesso: | | riprova ad inserire il codice | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Identificativo  *UC\_2* | | | | *Inserimento prodotti già registrati* | *Data* | | *03/11/20* |
| *Vers.* | *0.01.002* | |
| *Autore* | *Franco Cirillo* | |
| Descrizione | | | | *Il magazziniere vuole inserire nel sistema uno stock di prodotti appena arrivati.* | | | |
| Attore Principale | | | | Magazziniere  Si interessa di inserire prodotti appena arrivati. | | | |
| Attori secondari | | | | NA  NA | | | |
| Entry Condition | | | | È arrivato uno stock di prodotti presso un punto vendita | | | |
| Exit condition  On success | | | | Tutte le quantità relative ai prodotti arrivati sono aggiornate | | | |
| Exit condition  On failure | | | | Le quantità dei prodotti nel sistema non sono state aggiornate | | | |
| Rilevanza/User Priority | | | | Elevata | | | |
| Frequenza stimata | | | | 2/giorno | | | |
| Extension point | | | | NA | | | |
| Generalization of | | | | NA | | | |
| Flusso di Eventi Principale/Main Scenario | | | | | | | |
| 1 | Magazziniere: | | Accede all’area per l’inserimento dei prodotti | | | | |
| 2 | Sistema: | | Mostra un form per l’inserimento dei dati:   * Codice prodotto * Quantità | | | | |
| 3 | Magazziniere: | | Inserisce i dati del prodotto e sottomette | | | | |
| 4 | Sistema: | | Chiede al magazziniere se desidera inserire altri prodotti | | | | |
| 5 | Magazziniere: | | Se desidera inserire altri prodotti riprende dal punto 2, altrimenti sottomette | | | | |
| 4 | Sistema: | | Acquisisce i dati, mostra un elenco riassuntivo e richiede conferma | | | | |
| 5 | Magazziniere: | | Controlla i dati e conferma | | | | |
| 6 | Sistema: | | Aggiorna le quantità relative ai prodotti inseriti | | | | |
| I Scenario/Flusso di eventi di ERRORE: codice inserito non corretto | | | | | | | |
| 4.1 | | Sistema: | | mostra una notifica di errore. | | | |
| 4.2 | | Sistema: | | mostra di nuovo il form per l’inserimento dei dati | | | |
| 4.3 | | Magazziniere: | | riprova ad inserire i dati | | | |
|  | | | |  | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Identificativo  *UC\_3* | | | | *Richiedi assistenza* | *Data* | | *05/11/2020* |
| *Vers.* | *0.02.010* | |
| *Autore* | *Luigi Cirillo* | |
| Descrizione | | | | *Un cliente dopo aver acquistato un prodotto ha bisogno di assistenza* | | | |
| Attore Principale | | | | Cliente  Richiede assistenza su un prodotto acquistato | | | |
| Attori secondari | | | | Commesso  Deve prendere in carico la richiesta di assistenza | | | |
| Entry Condition | | | | Ha acquistato il prodotto in un punto vendita della catena. | | | |
| Exit condition  On success | | | | Richiesta accettata. | | | |
| Exit condition  On failure | | | | Richiesta rifiutata. | | | |
| Rilevanza/User Priority | | | | Media | | | |
| Frequenza stimata | | | | 10/giorno | | | |
| Extension point | | | | NA | | | |
| Generalization of | | | | NA | | | |
| Flusso di Eventi Principale/Main Scenario | | | | | | | |
| 1 | Cliente: | | Si presenta allo sportello e richiede assistenza su un prodotto mostrando lo scontrino | | | | |
| 2 | Commesso: | | Accede all’area assistenza ed apre la sezione per la creazione di un nuovo ticket | | | | |
| 3 | Sistema: | | Mostra il form per la creazione del ticket:   * nome e cognome cliente * numero di telefono * codice fiscale * indirizzo di residenza | | | | |
| 4 | Commesso: | | Compila il form | | | | |
| 5 | Sistema: | | Salva i dati e mostra form per le informazioni sul prodotto:   * tipo prodotto * nome prodotto * codice prodotto * numero di serie * numero scontrino acquisto * data scontrino acquisto | | | | |
| 6 | Commesso: | | Inserisce i dati del prodotto | | | | |
| 7 | Sistema: | | Salva i dati e conferma che il prodotto è stato acquistato meno di 2 anni fa | | | | |
| 8 | Sistema: | | Mostra form per l’inserimento dei dettagli del problema | | | | |
| 9 | Commesso: | | Chiede al cliente maggiori dettagli sul problema | | | | |
| 10 | Cliente: | | Spiega al commesso il problema | | | | |
| 11 | Commesso: | | Annota il problema nel form | | | | |
| 12 | Sistema: | | Salva i dettagli del problema e inoltra il ticket al reparto assistenza | | | | |
| 13 | Commesso: | | Ritira il prodotto, fornisce al cliente copia del ticket e lo avvisa che verrà ricontattato. | | | | |
| I Scenario/Flusso di eventi ERRORE: Cliente senza scontrino | | | | | | | |
| 1.1 | | Commesso: | | Avvisa il cliente che senza lo scontrino non ha diritto all’assistenza | | | |
| II Scenario/Flusso di eventi di ERRORE: Lo scontrino ha più di 2 anni | | | | | | | |
| 6.1 | | Sistema: | | Avvisa il commesso che il prodotto è fuori garanzia. | | | |
| 6.2 | | Sistema: | | Cancella il ticket corrente. | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Identificativo  *UC\_4* | | | | *Ordina* | *Data* | | *04/11/20* |
| *Vers.* | *0.01.002* | |
| *Autore* | *Vincenzo Aiello* | |
| Descrizione | | | | *Il magazziniere dopo aver ricevuto una notifica deve richiedere l’acquisto di prodotti che hanno raggiunto la quantità minima* | | | |
| Attore Principale | | | | Magazziniere  Si interessa di gestire i prodotti rimanenti in magazzino e richiedere i rifornimenti | | | |
| Attori secondari | | | | NA | | | |
| Entry Condition | | | | Il magazziniere riceve dal sistema una notifica che la quantità minima di un prodotto è stata raggiunta. | | | |
| Exit condition  On success | | | | Richiesta acquisto prodotti effettuata | | | |
| Exit condition  On failure | | | | Richiesta acquisto prodotti non effettuata | | | |
| Rilevanza/User Priority | | | | Elevata | | | |
| Frequenza stimata | | | | 2/giorno | | | |
| Extension point | | | | NA | | | |
| Generalization of | | | | NA | | | |
| Flusso di Eventi Principale/Main Scenario | | | | | | | |
| 1 | Magazziniere: | | Accede all’area dedicata per ottenere le previsioni di acquisto per i prodotti | | | | |
| 2 | Sistema: | | Mostra un form per l’inserimento del codice del prodotto | | | | |
| 3 | Magazziniere: | | Inserisce il codice del prodotto | | | | |
| 4 | Sistema: | | *Calcola quantità prodotti da acquistare* | | | | |
| 5 | Sistema: | | *Mostra il quantitativo consigliato da acquistare* | | | | |
| 6 | Magazziniere: | | *Conferma la richiesta d’acquisto dei prodotti nella quantità consigliata* | | | | |
| I Scenario/Flusso di eventi di ERRORE: Codice inserito non corretto | | | | | | | |
| 4.1 | | Sistema: | | Mostra una notifica di errore | | | |
| 4.2 | | Sistema: | | Mostra di nuovo il form per l’inserimento del codice | | | |
| 4.3 | | Magazziniere: | | Riprova ad inserire il codice | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Identificativo  *UC\_5* | | | | *Modifica prezzo prodotto* | | *Data* | | *06/11/20* |
| *Vers.* | *0.01.001* | |
| *Autore* | *Franco Cirillo* | |
| Descrizione | | | | *Il magazziniere deve aggiornare il prezzo di un prodotto* | | | | |
| Attore Principale | | | | Magazziniere  Aggiorna prezzi dei prodotti | | | | |
| Attori secondari | | | | NA  NA | | | | |
| Entry Condition | | | | Il magazziniere accede all’area per la modifica del prezzo di un prodotto. | | | | |
| Exit condition  On success | | | | Prezzo aggiornato | | | | |
| Exit condition  On failure | | | | Prezzo non aggiornato | | | | |
| Rilevanza/User Priority | | | | Media | | | | |
| Frequenza stimata | | | | 10/giorno | | | | |
| Extension point | | | | NA | | | | |
| Generalization of | | | | NA | | | | |
| Flusso di Eventi Principale/Main Scenario | | | | | | | | |
| 1 | Sistema: | | Mostra un form per l’inserimento del codice relativo al prodotto | | | | | |
| 2 | Magazziniere: | | Inserisce codice prodotto | | | | | |
| 3 | Sistema: | | Mostra prezzo attuale del prodotto | | | | | |
| 4 | Sistema: | | *Mostra un form per l’inserimento del nuovo prezzo* | | | | | |
| 5 | Magazziniere: | | *Inserisce il nuovo prezzo* | | | | | |
| 6 | Sistema: | | *Aggiorna le informazioni relative al prodotto* | | | | | |
| I Scenario/Flusso di eventi di ERRORE: Codice prodotto non corretto | | | | | | | | |
| 3.1 | | Sistema: | | Mostra una notifica di errore | | | | |
| 3.2 | | Sistema: | | Mostra di nuovo un form per l’inserimento del codice | | | | |
| 3.3 | | Magazziniere: | | Riprova ad inserire il codice | | | | |
| II Scenario/Flusso di eventi di ERRORE: Prezzo non positivo | | | | | | | | |
| 6.1 | | Sistema: | | | Mostra una notifica di errore | | | |
| 6.2 | | Sistema: | | | Mostra di nuovo un form per l’inserimento del prezzo | | | |
| 6.3 | | Magazziniere | | | Riprova ad inserire il prezzo | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Identificativo  *UC\_6* | | | | *Inserimento nuovi prodotti* | | *Data* | | *08/11/2020* |
| *Vers.* | *0.01.001* | |
| *Autore* | *Ciro Fusco* | |
| Descrizione | | | | *Il magazziniere vuole inserire nel sistema un nuovo prodotto appena arrivato.* | | | | |
| Attore Principale | | | | Magazziniere  Si interessa di inserire il prodotto appena arrivato. | | | | |
| Attori secondari | | | | NA  NA | | | | |
| Entry Condition | | | | È arrivato uno stock di un nuovo prodotto presso un punto vendita | | | | |
| Exit condition  On success | | | | Il nuovo prodotto risulta correttamente inserito nel sistema | | | | |
| Exit condition  On failure | | | | Il nuovo prodotto non viene inserito | | | | |
| Rilevanza/User Priority | | | | Elevata | | | | |
| Frequenza stimata | | | | 3/settimana | | | | |
| Extension point | | | | NA | | | | |
| Generalization of | | | | NA | | | | |
| Flusso di Eventi Principale/Main Scenario | | | | | | | | |
| 1 | Magazziniere: | | Accede all’area per l’inserimento di un nuovo prodotto | | | | | |
| 2 | Sistema: | | Mostra un form per l’inserimento dei dati:   * Codice prodotto * Nome Prodotto * Quantità * Prezzo | | | | | |
| 3 | Magazziniere: | | Inserisce i dati del prodotto e chiede di procedere | | | | | |
| 4 | Sistema: | | Acquisisce i dati, mostra un elenco riassuntivo e richiede conferma | | | | | |
| 5 | Magazziniere: | | Controlla i dati e conferma | | | | | |
| 6 | Sistema: | | Inserisce il nuovo prodotto nel sistema | | | | | |
| I Scenario/Flusso di eventi Errore: Codice prodotto già esistente | | | | | | | | |
| 4.1 | | Sistema: | | | Mostra una notifica di errore | | | |
| 4.2 | | Sistema: | | | Mostra di nuovo il form per l’inserimento dati | | | |
| 4.3 | | Magazziniere: | | | Riprova ad inserire i dati | | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Identificativo  *UC\_7* | | | | *Login* | *Data* | | *20/11/2020* |
| *Vers.* | *0.01.001* | |
| *Autore* | *Franco Cirillo* | |
| Descrizione | | | | *Login permette al dipendente di accedere alla propria area di lavoro* | | | |
| Attore Principale | | | | Dipendente  vuole accedere all’aerea personale per svolgere le sue mansioni, può essere un magazziniere, un cassiere o un addetto all’assistenza | | | |
| Attori secondari | | | | NA | | | |
| Entry Condition | | | | Dipendente localizzato sulla schermata relativa alle aree di lavoro | | | |
| Exit condition  On success | | | | Login effettuato con successo | | | |
| Exit condition  On failure | | | | Login non effettuato | | | |
| Rilevanza/User Priority | | | | Elevata | | | |
| Frequenza stimata | | | | 50/giorno | | | |
| Extension point | | | | NA | | | |
| Generalization of | | | | NA | | | |
| Flusso di Eventi Principale/Main Scenario | | | | | | | |
| 1 | Dipendente: | | raggiunge la schermata relativa alle aree di lavoro | | | | |
| 2 | Sistema: | | mostra le diverse aree di lavoro:   * Magazzino * Cassa * Assistenza | | | | |
| 3 | Dipendente: | | seleziona la sua area di lavoro | | | | |
| 4 | Sistema: | | mostra un form per l’inserimento delle credenziali:  nome Utente e password | | | | |
| 5 | Dipendente: | | inserisce le sue credenziali e sottomette | | | | |
| 6 | Sistema: | | verifica la correttezza e reindirizza all’area di competenza | | | | |
| I Scenario/Flusso di eventi di ERRORE: credenziali errate | | | | | | | |
| 6.1 | | Sistema: | | rimanda l’utente al punto 4 | | | |

### 3.4.4 Modello ad oggetti

### 3.4.4.1 Object Diagram

***OD\_1***

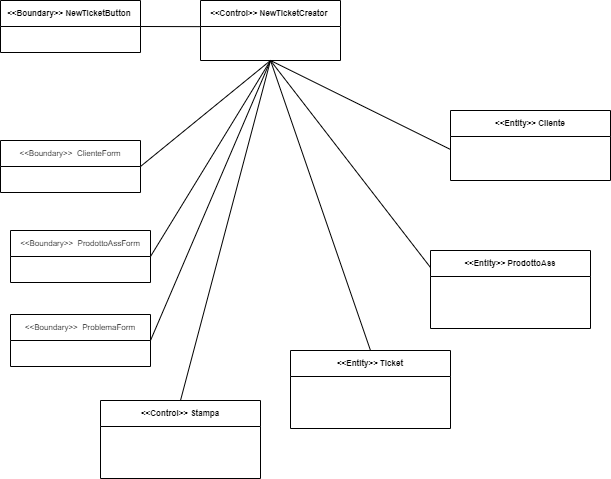
|  |  |  |
| --- | --- | --- |
| **Nome oggetto** | **Tipologia** | **Descrizione** |
| ScannerProdotti | Boundary | Boundary che permette di scannerizzare un prodotto per aggiungerlo alla lista dei prodotti che vuole acquistare il cliente |
| StampaScontrino | Boundary | Boundary che mostra la lista dei prodotti presenti sullo scontrino di acquisto |
| ScannerProdottiControl | Control | Control che coordina le operazioni relative formazione e alla stampa di uno scontrino |
| AggiornaMagazzinoControl | Control | Control che coordina le operazioni relative alla gestione dei prodotti |
| Prodotto | Entity | Entità che modella un prodotto |
| Scontrino | Entity | Entità che modella uno scontrino |

***OD\_2***

******

|  |  |  |
| --- | --- | --- |
| **Nome oggetto** | **Tipologia** | **Descrizione** |
| FormProdottiMag | Boundary | Boundary che permette al magazziniere di inserire il codice di un prodotto per modificarne le quantità rimanenti |
| FormRiassuntoProd | Boundary | Boundary che mostra le quantità aggiornate e che permette di confermare la modifica |
| AggiornaMagazzinoControl | Control | Control che coordina le operazioni relative alla gestione dei prodotti |
| Prodotto | Entity | Entità che modella un prodotto |

***OD\_3***

******

|  |  |  |
| --- | --- | --- |
| **Nome oggetto** | **Tipologia** | **Descrizione** |
| NewTicketButton | Boundary | Boundary che permette all’ addetto assistenza di avviare la procedura di creazione di un ticket |
| ClientForm | Boundary | Boundary che permette l’inserimento di dati relativi di un cliente |
| ProdottoAssForm | Boundary | Boundary che permette l’inserimento di dati relativi al prodotto da assistere |
| ProblemaForm | Boundary | Boundary che permette l’inserimento di dati relativi al problema che ha il prodotto |
| NewTicketCreator | Control | Control che coordina le operazioni relative all’apertura e salvataggio di un ticket per l’assistenza di un prodotto |
| Ticket | Entity | Entity che modella un ticket |
| Cliente | Entity | Entity che modella un cliente |
| ProdottoAss | Entity | Entity che modella un prodotto assistenza |

***OD\_4***

******

|  |  |  |
| --- | --- | --- |
| **Nome oggetto** | **Tipologia** | **Descrizione** |
| PrevediFornituraForm | Boundary | Boundary che permette al magazziniere di inserire il codice di un prodotto per prevedere le quantità di acquisto consigliate |
| ConfermaConsiglioFornitura | Boundary | Boundary che permette di confermare la richiesta di acquisto |
| PrevediFornituraControl | Control | Control che si occupa di calcolare le quantità di acquisto consigliate |
| ConfermaAcquistoControl | Control | Control che coordina le operazioni relative alle richieste di acquisto |
| RichiestaAcquisto | Entity | Entità che modella una richiesta di acquisto |

***OD\_5***

|  |  |  |
| --- | --- | --- |
| **Nome oggetto** | **Tipologia** | **Descrizione** |
| VisualizzaPrezzoForm | Boundary | Boundary che permette al magazziniere di inserire il codice di un prodotto per visualizzarne il prezzo |
| ModificaPrezzoForm | Boundary | Boundary che mostra il prezzo relativo al prodotto e che permette al magazziniere di inserire un prezzo al fine di poterlo modificare |
| ModificaPrezzoControl | Control | Control che coordina le operazioni relative alla modica dei dati di un prodotto |
| Prodotto | Entity | Entità che modella un prodotto |

***OD\_6***

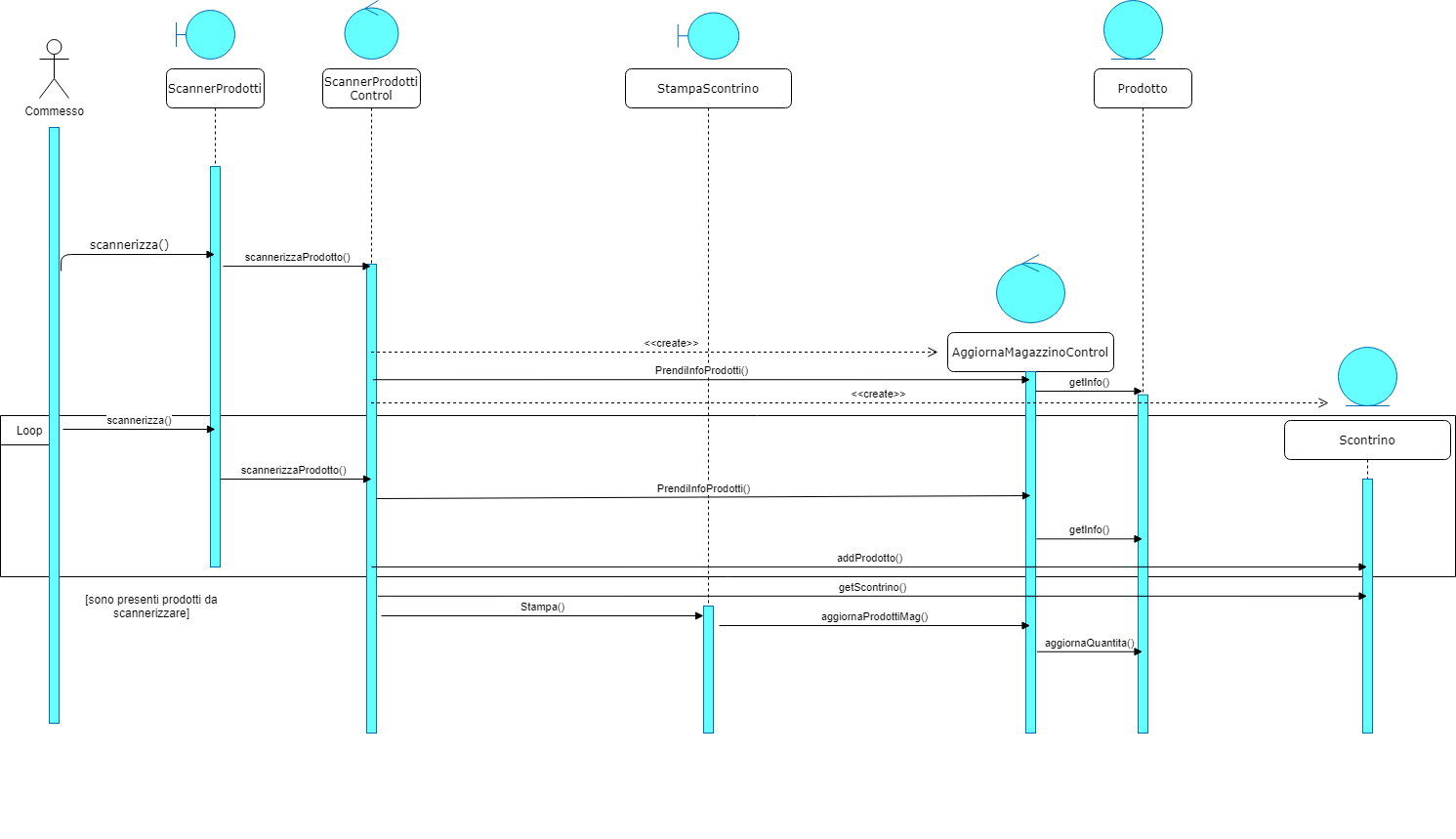
******

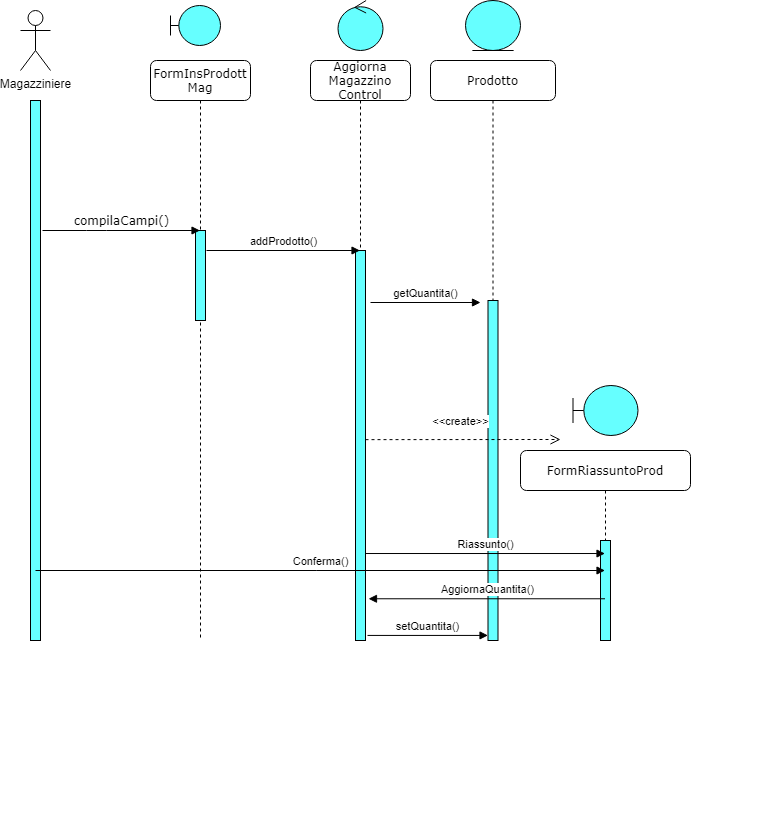
|  |  |  |
| --- | --- | --- |
| **Nome oggetto** | **Tipologia** | **Descrizione** |
| NewProduct | Boundary | Boundary che permette al magazziniere di inserire i dati di un nuovo prodotto per aggiungerlo |
| ViewElencoProdottiInseriti | Boundary | Boundary che mostra il riassunto del prodotto da inserire e che permette di confermare l’inserimento |
| InsertProdotto | Control | Control che coordina le operazioni relative all’inserimento di un nuovo prodotto |
| Prodotto | Entity | Entità che modella un prodotto |

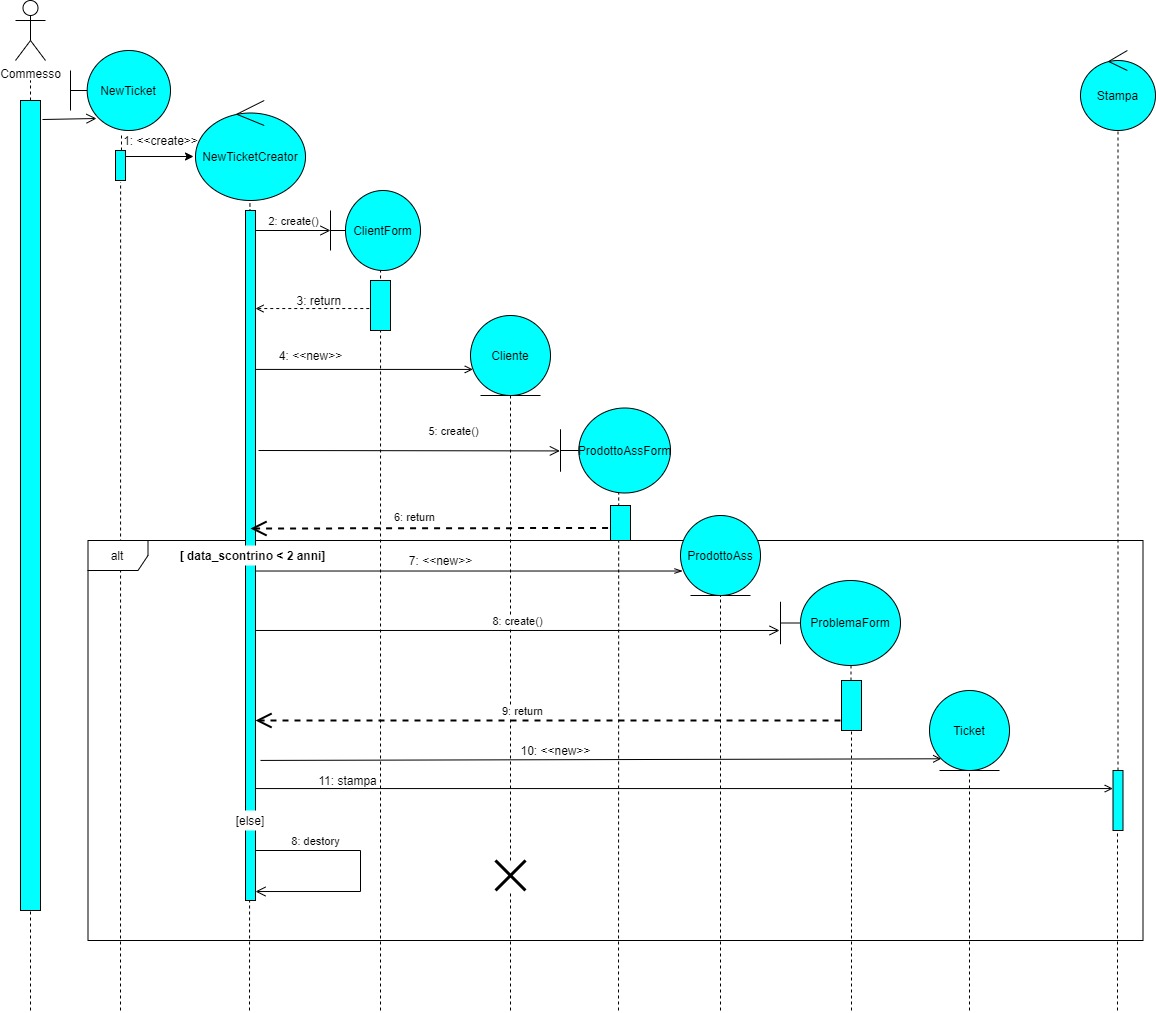
### 3.4.5 Modelli dinamici

### 3.4.5.1 Sequence Diagram

***SD\_1***

******

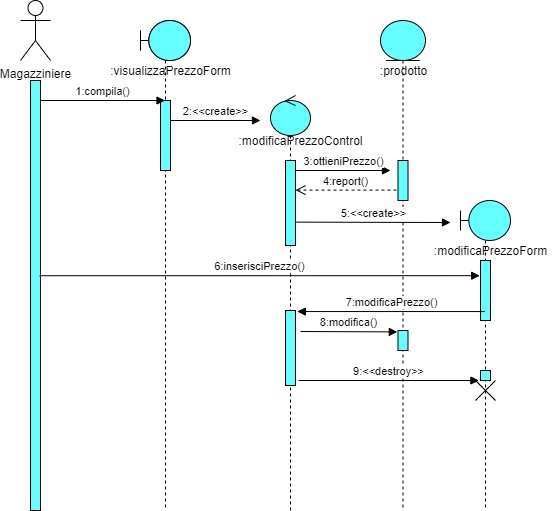
***SD\_2***

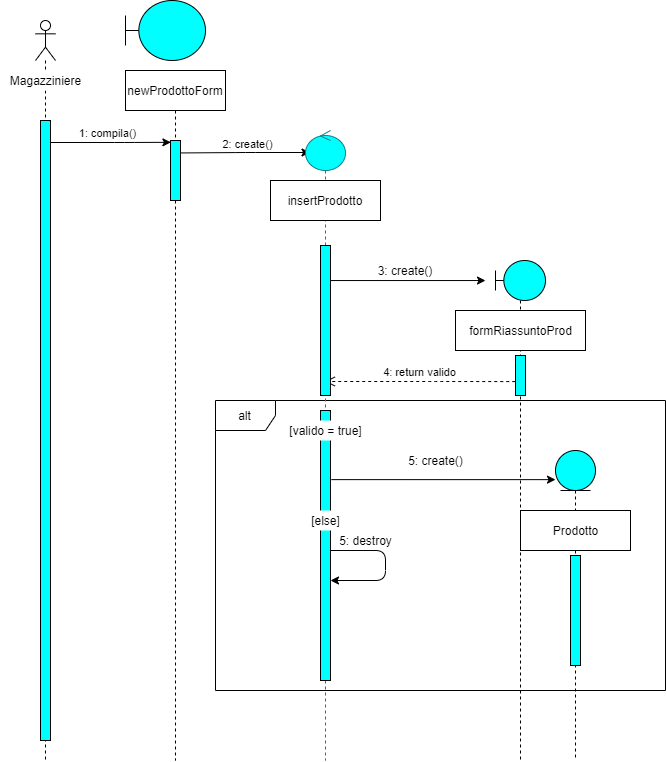
***SD\_3***

***SD\_4***

******

***SD\_5***

******

***SD\_6***

### 3.4.5.2 Statechart

SCD: Ticket**

SCD: scontrino

**

### 3.4.6 Interfaccia utente e mock-ups

### 3.4.6.1 Mock\_Ups

UI\_1:Dashboard******

UI\_2:Login****

UI\_3:Dashboard(magazzino)

UI\_4:InserisciProdotto

UI\_5:RiepilogoInserisciProdotto**Immagine che contiene testo

Descrizione generata automaticamente**

UI\_6:InserisciNuovoProdotto****

UI\_7:InserisciNuovoProdotto(riepilogo)**Immagine che contiene testo

Descrizione generata automaticamente**

UI\_8:OrdinaProdotto****

UI\_9:FormOrdinaProdotto

UI\_10:InserisciQuantitàProdotto******

UI\_11:ModificaPrezzoProdotto******

UI\_12:CambiaPrezzoProdotto******

UI\_13:InformazioneCliente(assistenza)**

UI\_14:InformazioneProdotto(assistenza)**

UI\_15:DettagliProblema(assistenza)*Immagine che contiene testo

Descrizione generata automaticamente*

UI\_16:Cassa ***Immagine che contiene testo

Descrizione generata automaticamente***

UI\_17:InserisciProdottoCliente(cassa)**

UI\_18:RiepilogoTotaleCliente(cassa)**

UI\_19:RiepilogoResto(cassa)*Immagine che contiene testo

Descrizione generata automaticamente*

### 3.4.6.1 Navigation path



# 4.Glossario

*Requisito funzionale*:funzionalità del sistema

*Requisito non funzionale*:tutti quei requisiti che non riguardano direttamente una funzionalità del sistema bensì riguardano caratteristiche di qualità come ad esempio Usabilità, Performance, Sopportabilità ed Affidabilità

*Use case*:viene utilizzato per esprimere l’interazione che avviene tra utente e sistema.

*Diagrammi dei casi d’uso (UCD)*:modellano il comportamento esterno di un sistema in termini delle funzioni che esso mette a disposizione agli attori che interagiscono con essi (utenti, altri sistemi software…). Gli UCS sono il diagramma principale nella vista dei casi d’uso.

*Sequence diagram*:diagramma previsto dall’UML utilizzato per descrivere uno use case descrivendo le relazioni che intercorrono in termini di messaggi tra attori, oggetti di business, oggetti o entità del sistema.

*Statechart diagram*:diagramma previsto dall’UML utilizzato per descrivere il comportamento di oggetti del sistema in termini di stato. In altre parole, modella quello che è il ciclo di vita di un oggetto.

*Mock-ups*:prototipazione digitale che permette la visione complessiva e generale di un’interfaccia utente.

*Navigation path*:grafico per la vista complessiva delle pagine presenti nel sistema e dei percorsi possibili per il loro raggiungimento.

*Form*:è un modulo elettronico che tramite un’interfaccia di un programma consente a un utente di inserire ed inviare uno o più dati.