

|  |  |  |  |
| --- | --- | --- | --- |
| FRANCO CIRILLO  LUIGI CIRILLO  CIRO FUSCO VINCENZO AIELLO |  |  |  |

RAD Requirement Analysis Document

**NewDM**

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versione | Cambiamenti | Autori |
| 20/11/2020 | Vr 0.1 |  | Luigi Cirillo |
|  |  |  |  |

Sommario

1. [1. Introduzione 3](#_Toc433975253)

[1.1 Scopo del sistema](#_Toc433975254)

[1.2 Ambito del sistema](#_Toc433975255)

[1.3 Obiettivi e criteri di successo del sistema](#_Toc433975256)

[1.4 Definizioni, acronimi e abbreviazioni](#_Toc433975257)

[1.5 Riferimenti](#_Toc433975258)

[1.6 Panoramica](#_Toc433975259)

1. [2. Sistema corrente 3](#_Toc433975260)
2. [3. Sistema proposto 3](#_Toc433975261)

[3.1 Panoramica](#_Toc433975262)

[3.2 Requisiti funzionali](#_Toc433975263)

[3.3 Requisiti no funzionali](#_Toc433975264)

[3.4 Modello di sistema](#_Toc433975265)

1. [4. Glossario 3](#_Toc433975266)

# 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

## 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 al 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 clientiCriteri di successo:

1.Branch coveragedei casi di test: almeno 75%

2.Buona manutenibilità e di integrabilità

## 1.4 Definizioni, acronimi e abbreviazioni

Nel seguente documento, viene utilizzato sia il termine “la mail” che “l’email” per indicare la parola e-mail

- RF: Requisito Funzionale

- NFR: Requisito Non Funzionale

- +: Vincoli o pseudo-requisiti

- “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

# 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.

# 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.

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

* Magazziniere
* Cassiere
* Assistente alla vendita

Da continuare…

## 3.2 Requisiti funzionali

|  |  |  |
| --- | --- | --- |
| Req.1 | 3 | Il commesso deve essere in grado di inserire manualmente il codice di un prodotto |
| Req.2 | 2 | Il commesso deve poter inserire manualmente la quantità di un prodotto |
| Req.3 | 3 | Il commesso deve richiedere al sistema di emettere lo scontrino |
| Req.4 | 2 | Il commesso deve poter modificare la lista prodotti |
| Req.5 | 1 | Il commesso deve inserire il totale dato dal cliente |
| Req.6 | 3 | Il commesso deve poter dare il resto |
| Req.7 | 3 | Ogni dipendente per accedere alle funzionalità della propria area di lavoro deve necessariamente inserire il proprio codice segreto, tale codice deve essere univoco |
| Req.8 | 1 | Il cliente dovrebbe poter vedere il prezzo dell’ultimo prodotto “scannerizzato” |
| Req.9 | 3 | Il cliente deve poter scegliere liberamente il metodo di pagamento tra quelli accettati |
| Req.10 | 2 | Il cliente deve poter richiedere assistenza per ogni prodotto acquistato |
| Req.11 | 3 | Il magazziniere deve poter inserire un nuovo prodotto |
| Req.12 | 3 | Il magazziniere deve poter modificare la quantità di un prodotto |
| Req.13 | 1 | Il magazziniere dovrebbe poter vedere la data di scadenza dei prodotti |
| Req.14 | 3 | Il magazziniere deve poter modificare il prezzo |
| Req.15 | 3 | Il magazziniere deve poter richiedere rifornimenti |
| Req.16 | 2 | L'addetto all’assistenza deve poter aprire un ticket per ogni prodotto da inviare in assistenza |
| Req.17 | 3 | Il modulo di IA deve essere in grado di fare una previsione sulla quantità di prodotti da acquistare |
| Req.18 | 3 | Il sistema deve aggiornare la quantità dei prodotti in magazzino ogni volta che viene effettuato un acquisto |
| Req.19 | 2 | 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

Il sistema deve:

* permettere di effettuare le operazioni in modo semplice e immediato, grazie a un’interfaccia user-friendly.
* 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.
* avere una documentazione minimale ma esaustiva.

### 3.3.2 Affidabilità

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. 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. Il sistema dovrà inoltre fornire un metodo di autenticazione sicuro in modo che i dati siano protetti da accessi fraudolenti.

### 3.3.3 Prestazioni

La scannerizzazione e la visualizzazione del prezzo del prodotto deve avvenire in al più sec. 2, in modo da evitare il crearsi di lunghe file alla cassa. Allo stesso modo il calcolo totale e l’emissione dello scontrino devono avvenire in al più sec. 3. 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. Il sistema deve essere sempre disponibile durante l’orario di lavoro (6am to 10pm).

### 3.3.4 Supportabilità

Il sistema deve essere di facile manutenzione, in modo tale da individuare rapidamente bug o errori di sistema e di correggerli in modo altrettanto rapido, e deve potersi adattare ai cambiamenti portati dalla realtà, da nuove esigenze da parte degli utenti e dalle nuove tecnologie.

### 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

[Esempi di utilizzo, informali e completi]

### 3.4.2 Use Case

[PER OGNI REQUISITO FUNZIONALE produrre uno use case.   
INFO VALUTAZIONE: ogni componente del team deve comunicare alle PM il numero di use case prodotti e la stima del tempo impiegato. Sottolineiamo che il tempo necessario per capire come gli use case sono strutturati e le prove fatte per produrne alcuni di esempio NON deve essere contemplato nel tempo dedicato alla realizzazione degli use case per il progetto.]

|  |  |
| --- | --- |
| **Nome** |  |
| **Attore** |  |
| **Condizione di Ingresso** |  |
| **Flusso degli Eventi** |  |
| **Condizione di Uscita** |  |
| **Condizioni di Errore** |  |
| **Requisiti di Qualità** |  |

### 3.4.3 Modello ad oggetti

<CONVENZIONE NOMI:   
 entity: usare nome usato dall’utente finale e in accordo al dominio dell’applicazione  
 control: [nome\_use\_case]Control (di solito dovremmo avere un oggetto control in funzione di ogni attore coinvolto in ogni use case. Dato che l’attore è sempre il presidente di area didattica noi avremo un solo oggetto control per use case)  
 boundary: strumenti che fungono da interfaccia con l’utente per erogare la funzionalità richiesta dall’utente.

TABELLA RIASSUNTIVA DI TUTTI GLI OGGETTI:>

|  |  |  |
| --- | --- | --- |
| Nome oggetto | Tipologia | Descrizione |
|  | [boundary, entity, control] |  |
|  |  |  |

<PER OGNI USE CASE RIPORTARE IL DIAGRAMMA DELLE RELAZIONI TRA GLI OGGETTI ENTITY; BOUNDARY E CONTROL COINVOLTI NELLO USE CASE CORRENTE. Per gli oggetti Entity specificare gli attributi, per gli oggetti control specificare le operazioni/metodi>

### 3.4.4 Modelli dinamici

[Diagramma di sequenza o diagrammi della attività.   
INFO VALUTAZIONE: Comunicare il tempo impiegato per la produzione del diagrammi di sequenza o attività. Sottolineiamo che il tempo necessario per capire come gli use case sono strutturati e le prove fatte per produrne alcuni di esempio NON deve essere contemplato nel tempo dedicato alla realizzazione dei diagrammi per il progetto]

### 3.4.5 Interfaccia utente e mock-ups

### [CONSIGLIO: i programmi che abbiamo usato per la produzione dei mock up sono balsamiq (free per 30 giorni) e Pencil. Nel momento in cui sviluppate i mockup tenete presente che il sistema dovrebbe essere una web application.]

# 4. Glossario

[Inserire qui tutti i termini utilizzati nel documento in modo da disambiguarli.]