

Specifica dei Requisiti

Gabriele Chignoli

Maggio 2025

Contents

1	Ingegneria dei Requisiti	2
1.1	Elicitazione	2
1.2	Utenti	2
1.3	Definizioni	3
1.4	Abbreviazioni	3
1.5	Priorità	3
1.6	Implementazione	3
1.7	Requisiti	4
2	UML	6
2.1	State Machine Diagram	6
3	Qualità del Software	7

1 Ingegneria dei Requisiti

1.1 Elicitazione

I requisiti sono stati elicitati operando direttamente sul team di sviluppo poiché l'idea originale del software è nata da reali necessità riscontrate nella vita personale dei membri. Si può attribuire a tale metodologia un carattere etnografico, in cui lo sviluppatore ha vissuto in prima persona l'ambiente del cliente, ed è quindi in grado di avere una buona comprensione dei requisiti necessari e delle loro priorità, senza farsi confondere da "fischietti e campanelli".

1.2 Utenti

L'applicativo è costruito per l'utilizzo in un ambiente ridotto, in cui il numero di prodotti da gestire/catalogare è limitato; difatti si considera l'ambiente casalingo il più adatto (e sensato) in cui utilizzare l'applicazione. Per questo la tipologia di utente con cui entrerà in contatto il software saranno persone a cui non è richiesta alcuna preparazione tecnica, se non una minima familiarità con sistemi digitali forniti di interfaccia grafica.

Viene illustrato nella *Figure 1* il diagramma dei casi d'uso, volto a mostrare le interazioni degli utenti (attori) con l'intero sistema e le operazioni che possono eseguire (casi d'uso).

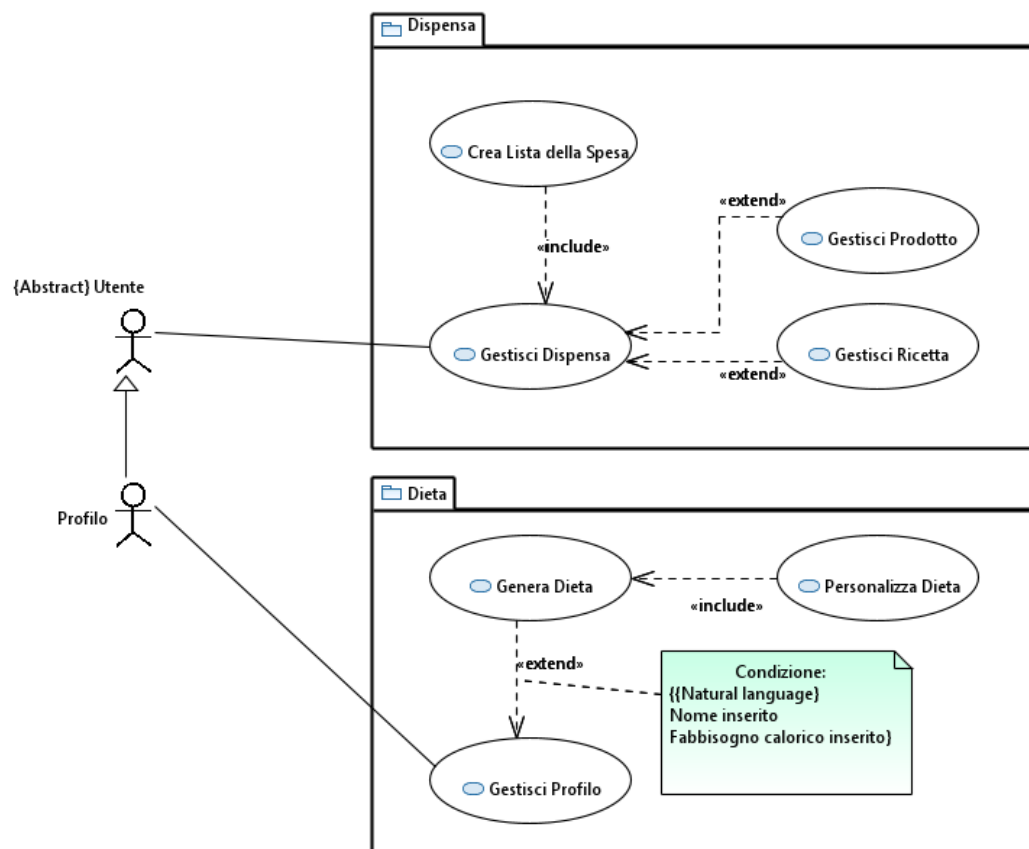


Figure 1: Diagramma dei casi d'uso

1.3 Definizioni

Chilocaloria [kcal]	Rappresenta la quantità di calore necessaria per innalzare la temperatura di 1 Kg di acqua distillata da 14,5°C a 15,5°C. Viene utilizzata in ambito alimentare come metrica per il potere calorico contenuto negli alimenti, cioè per misurare quanta "energia" è contenuta nell'alimento.
Dispensa (Pantry)	Nominativo con cui ci si riferisce all'inventario attuale dell'utente (e quindi all'insieme dei prodotti alimentari, di qualsiasi genere, di cui l'utente vuole tenere traccia).
Schermata Home	Si riferisce alla prima schermata dell'interfaccia grafica che l'applicazione mostra all'utente all'avvio.
Servizio Web Esterno	Si riferisce all'utilizzo di una connessione Internet per ricercare in rete un'informazione mancante, attraverso l'invio di una richiesta (HTTP) ad un ente in grado di fornire il dato.
Unità	Si riferisce alla più piccola parte ("atomica") di un prodotto, al livello di dettaglio che interessa all'utente, cioè al singolo barattolo/bottiglia/confezione/sacchetto/busta/frutto/vegetale/cassetta/scatola etc.
Profilo	Si riferisce ad un'entità digitale identificata da un nome, che rappresenta un utente reale.

1.4 Abbreviazioni

FUNZ	Requisiti funzionali
GUI	Requisiti dell'interfaccia grafica

1.5 Priorità

Nella tabella dei requisiti della sezione 1.7 viene specificata, oltre all'ID del requisito ed alla sua descrizione, la priorità attribuitagli, secondo lo schema *MoSCoW*, per il quale si possono suddividere i requisiti in

- **Must Have** - requisiti assolutamente necessari. Verrà indicato con **M**
- **Should Have** - requisiti importanti, ma non assolutamente necessari per un sistema utilizzabile. Verrà indicato con **S**
- **Could Have** - requisiti che vengono implementati solo se il tempo lo consente. Verrà indicato con **C**
- **Won't Have** - requisiti non richiesti che rimarranno per la prossima iterazione. Verrà indicato con **W**

1.6 Implementazione

Nella tabella dei requisiti della sezione 1.7 viene segnato (attraverso il simbolo: ✓) se il requisito è stato implementato nella versione rilasciata all'utente.

1.7 Requisiti

Funzionali

ID	Descrizione	Priorità	Impl.
<i>FUNZ.1</i>	Il sistema dovrebbe utilizzare una <i>Dispensa</i> per tenere traccia dei prodotti in possesso dell'utente.	M	✓
<i>FUNZ.2</i>	Il sistema dovrebbe permettere di aggiungere/modificare/rimuovere un prodotto dalla <i>Dispensa</i> .	M	✓
<i>FUNZ.3</i>	Il sistema dovrebbe tenere traccia, per ogni prodotto, delle seguenti informazioni: <ul style="list-style-type: none"> • Nome del prodotto • Data di scadenza • Quantità (peso, in chilogrammi/grammi [<i>kg,g</i>], volume, in litri/millilitri [<i>L,ml</i>] e/o numero di <i>Unità</i>) • Calorie per quantità (in chilocalorie per 100g di prodotto [<i>kcal/100g</i>]) 	M	✓
<i>FUNZ.4</i>	Il sistema dovrebbe contenere già al suo interno le informazioni per prodotto riportate nel requisito <i>FUNZ.3</i> degli alimenti più comuni.	C	
<i>FUNZ.5</i>	Il sistema dovrebbe poter recuperare le informazioni per prodotto riportate nel requisito <i>FUNZ.3</i> utilizzando un <i>Servizio Web Esterno</i> , indipendente dall'applicazione.	W	
<i>FUNZ.6</i>	Il sistema dovrebbe permettere all'utente di aggiornare/modificare/cancellare le informazioni riportate nel requisito <i>FUNZ.3</i> per ogni prodotto.	M	✓
<i>FUNZ.7</i>	Il sistema dovrebbe permettere di aggiungere/modificare/eliminare ricette. Per ogni ricetta il sistema memorizza i seguenti dati <ul style="list-style-type: none"> • Quantità (per ingrediente) • Note di preparazione • Tempo di preparazione (in minuti [<i>min</i>]) 	S	
<i>FUNZ.8</i>	Il sistema dovrebbe permettere di impostare l'obiettivo calorico giornaliero per l'utente principale.	C	

FUNZ.9	Il sistema dovrebbe essere in grado di calcolare (in modo approssimativo) il fabbisogno calorico necessario per l'utente, utilizzando i seguenti parametri (inseriti dall'utente): <ul style="list-style-type: none"> • Peso dell'individuo (in chilogrammi) • Altezza dell'individuo (in centimetri) • Attività fisica svolta <ul style="list-style-type: none"> – Tipo di attività – Durata dell'esercizio (minuti) 	W	
FUNZ.10	Il sistema dovrebbe permettere di creare dei <i>Profili</i> che identificano un individuo di cui si vuole impostare un obiettivo calorico.	C	
FUNZ.10.1	Il sistema dovrebbe permettere di impostare l'obiettivo calorico giornaliero per ogni Profilo.	C	
FUNZ.11	Il sistema dovrebbe permettere di creare una <i>Lista della spesa</i> in cui l'utente può aggiungere prodotti che ha intenzione/necessità di acquistare.	W	
FUNZ.12	Il sistema dovrebbe permettere di recuperare ricette attraverso un Servizio Web esterno, e proporle all'utente.	W	
FUNZ.13	Il sistema dovrebbe notificare l'utente quando un prodotto sta per scadere.	M	✓
FUNZ.13.1	La notifica verrà inviata 3 giorni prima della scadenza.	S	✓

Graphic User Interface (GUI)

ID	Descrizione	Priorità	Impl.
GUI.1	L'interfaccia utente dovrebbe rispondere in modo reattivo agli input dell'utente, fornendo un feedback visivo.	S	✓
GUI.2	La data di scadenza dei prodotti sarà segnata <ul style="list-style-type: none"> • In giallo per i prodotti in scadenza (cioè a 3 giorni dalla data limite del prodotto) • In rosso per i prodotti che hanno superato la data di scadenza 	S	
GUI.3	Il sistema dovrebbe permettere all'utente di rimuovere velocemente i prodotti che ha utilizzato.	S	✓
GUI.4	L'interfaccia dovrebbe disporre di una schermata principale detta " <i>Schermata Home</i> " (da qui in poi detta semplicemente "Home").	M	✓
GUI.4.1	La Home dovrebbe mostrare i prodotti e disporli in un ordinamento a matrice o ad elenco.	M	✓
GUI.4.2	Dovrebbero venire mostrate le immagini dei prodotti.	S	

GUI.4.3	Dovrebbero venire mostrati i nomi dei prodotti.	S	✓
GUI.5	In caso l'applicazione vada in background, o venga sospesa, quando viene riattivata deve tornare all'ultima schermata su cui era prima della sospensione.	W	
GUI.6	Il requisito GUI.3 dovrebbe essere soddisfatto ponendo un pulsante "+" e un pulsante "-" sotto le immagini dei prodotti.	S	✓

2 UML

2.1 State Machine Diagram

Si rappresentano di seguito i diagrammi UML delle Macchine a Stati Finiti che modellano alcune parti del software; in particolare vengono rappresentati gli stati in cui dovrebbe trovarsi l'interfaccia grafica per l'utente (GUI, *Figure 3*), e i possibili stati di un prodotto della dispensa (*Figure 2*).

La visione dei modelli UML è disponibile anche dal repository Github del progetto.

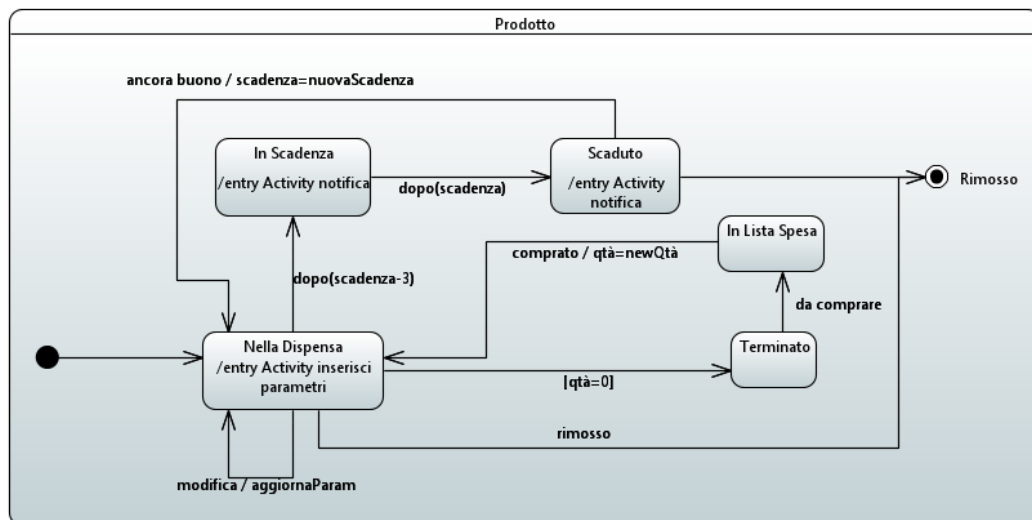


Figure 2: Product State Machine Diagram

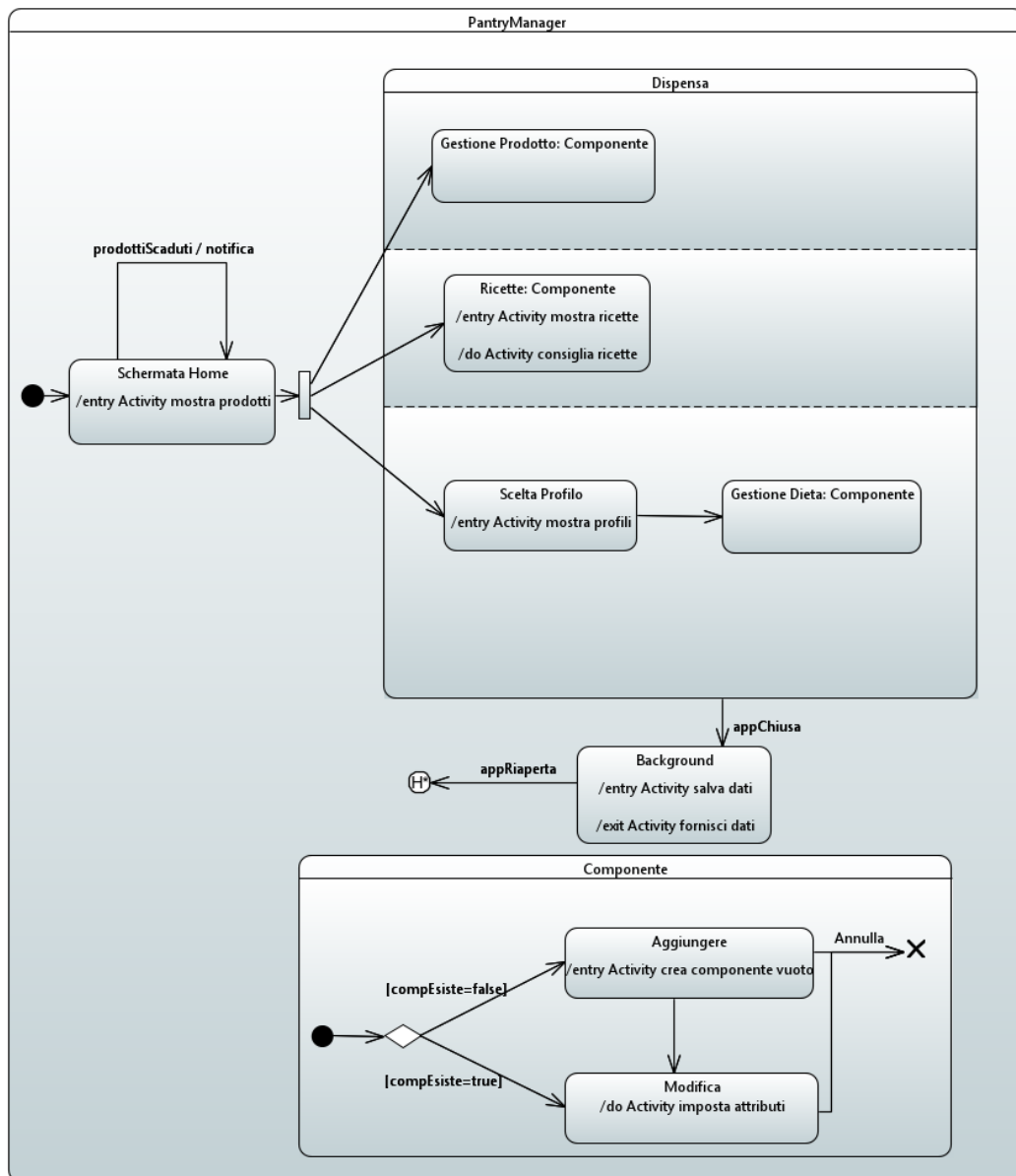


Figure 3: GUI State Machine Diagram

3 Qualità del Software

Si è deciso di non adottare uno standard predefinito per la definizione dei requisiti di qualità, poiché si ritiene che l'effettiva verifica tecnica del soddisfacimento della qualità richieda non risulti fattibile; si è comunque deciso di definire dei "principi" generici che hanno guidato lo sviluppo.

- **Correttezza** - si vuole costruire un software che fornisca all'utente le funzionalità che promette, producendo l'output corretto e atteso.
- **Usabilità** - si intende sviluppare un'interfaccia sufficientemente reattiva e comprensibile, che abbia un design semplice e magari meno appetibile, ma comunque immediato e intuitivo.

- **Riutilizzabilità** - si intende costruire un software le cui componenti siano facilmente riutilizzabili per lo sviluppo di un altro applicativo che sia nello stesso dominio (software che sia in grado di salvare elementi sul dispositivo, ne tenga traccia, e permetta di aggiungerli, modificarli o toglierli attraverso un interfaccia grafica).