

# **Progetto Ingegneria del Software: Gestionale “Sweet & Sour”**

Matteo Casarotto Sant’Ana, Alvin Preidt, Enrico Zasso

# Indice:

<b>Progetto Ingegneria del Software: Gestionale “Sweet &amp; Sour”</b> .....	<b>1</b>
<b>Indice:</b> .....	<b>2</b>
<b>1. OBIETTIVI DEL PROGETTO</b> .....	<b>3</b>
1.1. Flusso standard di un prodotto:.....	3
<b>2. REQUISITI FUNZIONALI</b> .....	<b>4</b>
2.1. Login.....	4
2.2. Utente autenticato.....	4
2.3. Utente “dipendente”.....	4
2.3.1. Operazioni sui prodotti:.....	4
2.3.2. Operazioni sui container.....	5
2.3.3. Movimenti.....	5
2.4. Utente “amministratore”.....	6
2.4.1. Gestione utenti.....	6
2.4.2. Statistiche.....	6
<b>3. REQUISITI NON FUNZIONALI</b> .....	<b>6</b>
3.1. Usabilità.....	6
3.2. Affidabilità.....	7
3.3. Prestazioni.....	7
3.4. Sicurezza.....	7
3.5. Scalabilità.....	7
3.6. Manutenzione.....	7
3.7. Compatibilità.....	8
3.8. Backup e Ripristino.....	8
3.9 Conformità normativa.....	8
<b>4. DESIGN BACK-END</b> .....	<b>8</b>
4.1 Integrazione con Sistemi Esterni:.....	8
4.2 Funzionalità CRUD (Create, Read, Update, Delete):.....	9

# 1. OBIETTIVI DEL PROGETTO

Nel bar Akka, attualmente, la gestione dell'inventario dei prodotti e i loro spostamenti nei banconi e nel magazzino vengono tracciati manualmente con un sistema cartaceo. Il personale a fine serata (o prima dell'inizio del turno successivo) conta i prodotti mancanti nei banconi, compila una lista e rifornisce i banconi dal magazzino. Quest'ultimo viene poi a sua volta approvvigionato, durante il fine settimana, attraverso ordini inviati dal manager/direttore ai vari fornitori. Questo sistema richiede una considerevole quantità di tempo per il censimento dei prodotti mancanti (sia nei banconi per il rifornimento che nel magazzino per gli ordini).

L'obiettivo del progetto è quindi la realizzazione di un software gestionale di magazzino per il bar Akka. Il software dovrà offrire:

- Una conta dei prodotti all'interno dei containers<sup>1</sup>;
- Metodi per muovere e consumare i prodotti.
- Delle statistiche sul movimento e sul consumo dei prodotti;
- Funzioni che automatizzano i movimenti svolti più di frequente (consumo, carichi e acquisto).

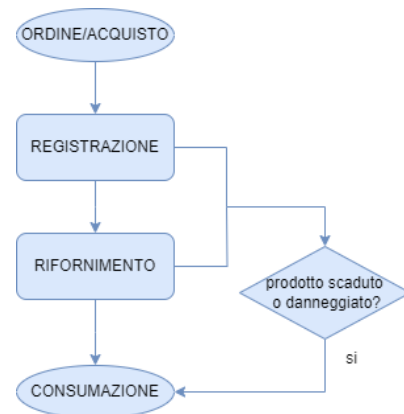
Il corretto soddisfacimento di questi obiettivi garantirebbe al bar:

- Un risparmio di tempo e denaro;
- Una riduzione degli errori e degli sprechi;
- Una semplificazione del lavoro dei dipendenti;
- Una migliore soddisfazione dei clienti;

## 1.1. Flusso standard di un prodotto:

Tipicamente all'interno del gestionale un prodotto segue questo percorso:

- Viene acquistato attraverso ordini fatti presso i fornitori;
- Viene registrato nel magazzino;
- Viene spostato dal magazzino per rifornire i vari banconi;
- Viene venduto, quindi consumato.



<sup>1</sup> **Container:** per il sistema si definiscono “container” gli spazi fisici in cui avviene lo stoccaggio dei prodotti, i container fisici avranno una corrispondenza virtuale nel sistema.

## 2. REQUISITI FUNZIONALI

### 2.1. Login

L'utente può effettuare il login, specificando nome utente e password, per poter accedere alle funzionalità dell'applicazione. Tutte le altre operazioni messe a disposizione dal sistema sono riservate all'utente autenticato.

### 2.2. Utente autenticato

Un utente autenticato rappresenta all'interno del gestionale una persona reale o una postazione di lavoro.

Esistono due tipi di utenti:

- Amministratore
- Dipendente

Ad ognuno di essi deve essere associata una lista di autorizzazioni che consentono l'accesso totale o parziale alle funzionalità messe a disposizione dall'applicazione.

### 2.3. Utente “dipendente”

L'utente “dipendente” è il principale utilizzatore del sistema e ha quindi la gran parte delle funzioni, essi infatti devono poter fare:

#### 2.3.1. Operazioni sui prodotti:

- Visualizzare la lista di tutti i prodotti esistenti nel sistema con i rispettivi parametri.
- Effettuare la ricerca di un prodotto tramite parametri (nome, riempimento<sup>2</sup>, categoria, fornitore).
- Aggiungere un nuovo tipo di prodotto al sistema, definendone anche le capacità per ogni container.
- Visualizzare tutte le informazioni che descrivono un prodotto e della relativa quantità presente in ogni container.
- Modificare le informazioni che descrivono un prodotto.
- Eliminare dal sistema un prodotto.

---

<sup>2</sup> **Riempimento di un prodotto in un container:** Si definisce come il rapporto tra la quantità di prodotto nel container e la capacità del container relativa a quel prodotto.

**Riempimento di un prodotto:** Si definisce come il rapporto tra la somma delle quantità di prodotto in ogni container e la somma delle capacità di ogni container relative a quel prodotto..

### 2.3.2. Operazioni sui container

- Visualizzare una lista dei container.
- Visualizzare una lista contenente tutti i prodotti presenti all'interno di un container.
- Effettuare la ricerca di un prodotto all'interno di un container tramite parametri (nome, riempimento, categoria, fornitore).

### 2.3.3. Movimenti

- Visualizzare la cronologia di tutti i movimenti eseguiti all'interno del gestionale.
- Effettuare la ricerca di un movimento tramite parametri (prodotto, motivazione<sup>3</sup>, autore, container di origine, container di destinazione).
- Creazione di un movimento specificando i suoi attributi (prodotto da muovere, motivazione, quantità, container di origine, container di destinazione e descrizione del movimento). La creazione di un movimento ha l'effetto di sottrarre la quantità di prodotto selezionata dal container di origine e aggiungerla a quello di destinazione.
- Modificare un movimento.
- Eliminare un movimento ne annulla gli effetti.
- Eseguire delle funzioni che velocizzano e semplificano i tipi di movimenti svolti più di frequente:
  - **Carico:**
    - L'utente seleziona due containers: provenienza e destinazione, diversi tra di loro.
    - Il sistema permette di visualizzare una lista che indichi la quantità mancante di ogni prodotto necessaria per soddisfare la capacità<sup>4</sup> del container di destinazione di quel specifico prodotto.
    - Il sistema effettua automaticamente lo spostamento di tutti i prodotti nella lista dal container di provenienza al container di destinazione.
      - Il sistema deve indicare all'utente quando, per alcuni prodotti, la quantità nel container di provenienza non è sufficiente a coprire la quantità mancante nel container di destinazione.
  - **Consumazione Fast:**
    - L'utente seleziona un container per la "consumazione Fast".
    - L'utente scannerizza il codice a barre di prodotto con il lettore di codice a barre.
    - Il sistema crea automaticamente un movimento per rimuovere dal gestionale una quantità unitaria del prodotto scansionato.

---

<sup>3</sup> Le motivazioni disponibili devono essere: comprato, consumato, rotto, scaduto, carico, aggiustamento, furto, sconosciuto.

<sup>4</sup> Ogni container ha una specifica capacità totale di uno specifico prodotto (esempio: nel banco grande ci possono essere stoccate 150 bottigliette d'acqua mentre in quello piccolo 50), questa capacità viene scelta/modificata anche sulla base dei report statistici.

- **Acquisto:**
  - L'utente carica nel sistema una fattura elettronica, indicando il fornitore e un container di destinazione.
  - Il sistema effettua un'analisi della fattura, individuando la quantità e i nomi dei prodotti acquistati. Durante l'analisi, se un prodotto all'interno della fattura risulta associato a un prodotto nel gestionale, il sistema crea automaticamente un movimento per aggiungere la quantità di prodotto indicata nella fattura all'interno del container di destinazione.
  - L'utente può definire delle regole per il riconoscimento automatico dei prodotti.

## 2.4. Utente “amministratore”

Deve esistere almeno un utente con i permessi di amministratore, il quale può accedere a tutte le funzionalità dell'utente "dipendente" e inoltre ha i seguenti permessi:

### 2.4.1. Gestione utenti

- Creare un nuovo utente.
- Visualizzare i permessi di ogni singolo utente.
- Modificare i permessi di ogni utente non amministratore.
- Eliminare un utente.

### 2.4.2. Statistiche

- Il gestionale deve fornire report statistici e grafici sui consumi dei vari prodotti al fine di ottimizzare gli acquisti e ridurre al minimo lo spreco di spazio del magazzino.

## 3. REQUISITI NON FUNZIONALI

### 3.1. Usabilità

- **Interfaccia Utente Intuitiva:** Il sistema deve avere un'interfaccia utente intuitiva e facile da utilizzare per consentire ai membri del personale del bar di sfruttare facilmente il software.
- **Documentazione Completa:** Deve essere fornita una documentazione completa per gli utenti e per il personale addetto alla manutenzione, che spieghi chiaramente come utilizzare il sistema e come risolvere eventuali problemi comuni.
- **Gestione degli errori:** Si deve tener conto che l'ambiente di uso del sistema è molto frenetico quindi bisogna dar modo al personale di gestire in maniera ottimale gli errori rendendo modificabile ogni operazione.

## 3.2. Affidabilità

- **Disponibilità Elevata:** Il sistema deve essere sempre disponibile durante l'orario di apertura del bar, minimizzando il tempo di inattività causato da eventuali guasti.
- **Accessi concorrenti:** Il sistema deve gestire in maniera ottimale gli accessi concorrenti per garantire l'affidabilità visto che il bar dispone di vari banchi e molti addetti che ci lavorano anche contemporaneamente.
- **Integrità e coerenza dei dati:** Il sistema deve gestire in maniera opportuna le operazioni eseguite dagli utenti evitando che queste vadano ad intaccare l'integrità dei dati.

## 3.3. Prestazioni

- **Tempo di risposta:** Il sistema deve rispondere rapidamente alle richieste degli utenti, specialmente durante le ore di punta del bar, per evitare ritardi nel servizio. In particolare i barcode scanners devono svolgere la lettura di un codice a barre e la corrispettiva immissione nel software in meno di un secondo.

## 3.4. Sicurezza

- **Protezione dei Dati:** Il sistema deve garantire la sicurezza dei dati sensibili, come informazioni sugli utenti, dati di inventario e transazioni. Deve prevedere un sistema di autenticazione sicuro e autorizzazioni basate sui ruoli per proteggere l'accesso non autorizzato.

## 3.5. Scalabilità

- **Scalabilità:** Il sistema deve essere in grado di gestire un aumento del volume di dati nel tempo, considerando sia una scalabilità verticale (aumento di prodotti, transazioni o utenti dello stesso locale) che orizzontale (cioè l'apertura di nuovi locali in futuro)

## 3.6. Manutenzione

- **Manutenzione:** Il software deve essere progettato in modo da consentire aggiornamenti e correzioni di bug, che devono essere portati a termine durante gli orari/giorni di chiusura del bar, in modo da non compromettere il flusso di lavoro.

### 3.7. Compatibilità

- **Compatibilità dei Dispositivi:** Il software deve essere accessibile tramite browser. In particolare deve essere compatibile, sempre tramite browser, con i tablet android messi a disposizione del cliente.
- **Barcode scanners:** Deve essere possibile collegare delle pistole per codici a barre, tramite USB OTG, ai dispositivi che si collegano al sistema. Queste devono essere configurate in modo che, una volta acquisito un codice a barre, agiscano come una tastiera collegata al dispositivo: prima scrivono uno alla volta i caratteri numerici presenti nel barcode, infine simulano la pressione del tasto invio.

### 3.8. Backup e Ripristino

- **Sistema di Backup:** È previsto un sistema di backup periodici in modo da poter ripristinare il sistema in caso di guasti o errori irreversibili.

### 3.9 Conformità normativa

- **Conformità alle Normative:** Il sistema deve essere progettato per essere conforme a tutte le leggi e i regolamenti locali e nazionali che disciplinano la gestione degli inventari in un contesto di bar. Inoltre, deve garantire la protezione dei dati sensibili, implementando misure di sicurezza adeguate, inclusa la crittografia e il controllo degli accessi basati su ruoli, per rispettare le normative sulla privacy e la sicurezza dei dati personali.

## 4. DESIGN FRONT-END

### 4.1 Layout Organizzato

Il design del front-end del sistema "Sweet & Sour" è stato attentamente pianificato per massimizzare l'usabilità e migliorare l'esperienza dell'utente. I principi guida includono:

#### 4.1.1 Menu Laterale Intuitivo

È stato implementato un menu laterale che fornisce accesso diretto a tutte le sezioni del sistema, tra cui la lista dei prodotti, i movimenti, il carico dei container e altro ancora. Questo menu laterale è progettato per essere intuitivo e facile da navigare, garantendo un accesso rapido alle funzionalità chiave.



### **4.1.2 Pulsanti Ben Visibili**

Tutti i pulsanti all'interno dell'applicazione sono stati progettati per essere ben visibili e facilmente distinguibili. Abbiamo utilizzato colori contrastanti e dimensioni ottimali per garantire che gli utenti possano individuare facilmente i pulsanti e interagire con essi senza sforzo.

### **4.1.3 Organizzazione delle Schermate**

Ogni schermata è stata organizzata con attenzione per mettere in evidenza le informazioni più importanti e le azioni più comuni. Ciò significa che gli utenti possono concentrarsi sulle attività principali senza essere distratti da elementi superflui.

### **4.1.4 Flussi di Utilizzo Intuitivi**

Abbiamo lavorato per garantire che i flussi di utilizzo all'interno del programma siano chiari e intuitivi. Gli utenti possono facilmente accedere alle funzionalità relative a un container specifico o a un prodotto senza dispersione di tempo o confusione.

## **4.2 Feedback Utente**

### **4.2.1 Conferme Immediate**

Abbiamo implementato feedback visivi per fornire agli utenti conferme immediate sulle azioni eseguite. Ad esempio, dopo l'aggiunta di un prodotto a un container, verrà mostrato un messaggio di conferma o un'icona di successo, migliorando così la percezione dell'utente.

## **4.3 Responsive Design**

Il design del front-end è stato ottimizzato per garantire una perfetta fruizione del sistema su dispositivi mobili, tablet e desktop. Questo assicura un'esperienza uniforme su tutte le piattaforme, consentendo al personale del bar di utilizzare il sistema in modo flessibile.

## 5. DESIGN BACK-END

Il back-end del sistema "Sweet & Sour" deve essere progettato utilizzando un'architettura scalabile e modulare. Alcuni principi importanti includono:

### 5.1 Integrazione con Sistemi Esterni:

**Ordini Automatici:** Il sistema "Sweet & Sour" può automatizzare l'invio di ordini ai fornitori in base alle scorte di magazzino. Questo significa che il bar non dovrà più preoccuparsi di tenere traccia manualmente degli approvvigionamenti. Il processo di ordine diventa più rapido ed efficiente.

**Integrazione con Fatture Elettroniche:** implementare un sistema che possa leggere e analizzare i file CSV delle fatture elettroniche dai fornitori. Questo sistema dovrebbe essere in grado di identificare i prodotti acquistati, crearne automaticamente i movimenti e aggiornare l'inventario di conseguenza.

### 5.2 Funzionalità CRUD (Create, Read, Update, Delete):

"Sweet & Sour" offre un set di funzionalità CRUD che comprende quattro operazioni principali:

- **Creazione (Create):** Per aggiungere nuovi dati, come nuovi prodotti.
- **Lettura (Read):** Per visualizzare informazioni esistenti, ad esempio dettagli sui prodotti.
- **Aggiornamento (Update):** Per apportare modifiche ai dati esistenti, come cambiare il prezzo di un prodotto.
- **Eliminazione (Delete):** Per rimuovere dati obsoleti o non più necessari, come prodotti esauriti.