# **Modello dei Dati per DispensAPP**

## **1️⃣ Obiettivo del Modello**

Il database in SQLite deve gestire in modo semplice la dispensa, la lista della spesa e il monitoraggio dei prodotti in scadenza. I dati verranno successivamente esportati in MariaDB per analisi avanzate.

## **2️⃣ Tabelle Principali**

### **📌 Dimensioni (Tabelle di riferimento)**

Le tabelle dimensionali contengono informazioni statiche e descrittive.

#### **1. Prodotti (Products)**

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo** | **Descrizione** |
| Product\_ID | INTEGER PK | Identificativo univoco del prodotto |
| Barcode | TEXT | Codice a barre del prodotto |
| Nome | TEXT | Nome del prodotto |
| Marca | TEXT | Marca del prodotto |
| Quantita\_Min | INTEGER | Quantità minima desiderata in dispensa |
| Immagine | TEXT | Percorso immagine del prodotto |

#### **2. Punti Vendita (Shops)**

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo** | **Descrizione** |
| Shop\_ID | INTEGER PK | Identificativo del negozio |
| Nome | TEXT | Nome del negozio |
| Nota | TEXT | Note generiche |

#### **3. Categorie (Categories)**

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo** | **Descrizione** |
| Category\_ID | INTEGER PK | Identificativo della categoria |
| Nome | TEXT | Nome della categoria |
| Nota | TEXT | Note generiche |

### **🔹 Tabella Transazionale (Movimentazione Prodotti)**

Questa tabella registra le variazioni di quantità dei prodotti nel tempo.

#### **4. Movimenti Prodotti (Product\_Transactions)**

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo** | **Descrizione** |
| Transaction\_ID | INTEGER PK | ID della transazione |
| Product\_ID | INTEGER FK | Prodotto acquistato |
| Shop\_ID | INTEGER FK | Negozio di acquisto |
| Category\_ID | INTEGER FK | Categoria del prodotto |
| Data\_Inserimento | DATE | Data di inserimento del prodotto in dispensa |
| Data\_Scadenza | DATE | Data di scadenza del prodotto |
| Data\_Consumo | DATE NULL | Data di consumo (se null → prodotto non consumato) |

⏳ **Regola di mantenimento:** I record "chiusi" (con Data\_Consumo valorizzata) vengono eliminati dopo un mese.

### **🛒 Liste di Supporto (Aggiornamento automatico giornaliero)**

#### **5. Lista della Spesa (Shopping\_List)**

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo** | **Descrizione** |
| Product\_ID | INTEGER FK | Prodotto necessario |
| Shop\_ID | INTEGER FK | Negozio preferito |
| Quantita | INTEGER | Quantità da acquistare |

📌 **Regola di aggiornamento:** Ogni sera i prodotti con quantità inferiore alla Quantita\_Min vengono aggiunti.

#### **6. Prodotti in Scadenza (Expiring\_Products)**

|  |  |  |
| --- | --- | --- |
| **Campo** | **Tipo** | **Descrizione** |
| Product\_ID | INTEGER FK | Prodotto in scadenza |
| Data\_Scadenza | DATE | Data di scadenza |

📌 **Regola di aggiornamento:** Ogni sera vengono inseriti i prodotti che scadono entro 2 settimane.

## **3️⃣ Flusso dei Dati**

1. **Acquisto Prodotto** → Inserito in Product\_Transactions.
2. **Consumo Prodotto** → Si aggiorna Data\_Consumo e viene rimosso dopo un mese.
3. **Aggiornamento Serale**:
   1. **Lista della Spesa**: Controllo Quantita\_Min e aggiunta prodotti mancanti.
   2. **Prodotti in Scadenza**: Aggiunta prodotti con scadenza entro 14 giorni.

## **4️⃣ Transizione verso MariaDB**

* **SQLite** gestisce le operazioni giornaliere.
* **MariaDB** riceverà periodicamente i dati per analisi avanzate con Power BI.
* **I dati storici** vengono cancellati da SQLite dopo il trasferimento a MariaDB.

📌 **Benefici di questa struttura**: ✅ Database leggero per l’applicazione. ✅ Eliminazione automatica dei dati obsoleti. ✅ Possibilità di analisi dettagliata tramite MariaDB & Power BI.

## **5️⃣ Prossimi Passi**

🔲 Creazione schema SQLite 🔲 Sviluppo query di aggiornamento serale 🔲 Integrazione con MariaDB 🔲 Ottimizzazione interfaccia utente

🎯 **Obiettivo finale**: Un sistema pratico, semplice e scalabile! 🚀