

# SRS Poldo – Sistema di Prenotazione per il Bar Scolastico

---

## Sommario

### 1. Introduzione

1. Scopo del documento
2. Ambito del progetto
3. Descrizione del sistema
4. Team di sviluppo
5. Riferimenti
6. Glossario

### 2. Descrizione generale

1. Contesto e panoramica
2. Prospettiva del prodotto
3. Funzionalità principali
4. Profili utente
5. Ambiente operativo
6. Vincoli e dipendenze
7. Assunzioni e prerequisiti

### 3. Requisiti funzionali

1. Autenticazione e gestione utenti
2. Gestione prodotti
3. Visualizzazione menu e prodotti
4. Gestione ordini utente
5. Gestione ordini bar
6. Reporting e statistiche

#### 4. Requisiti di interfaccia

1. Interfacce utente
2. Interfacce hardware
3. Interfacce software
4. Interfacce di comunicazione

#### 5. Requisiti non funzionali

1. Performance
2. Usabilità
3. Affidabilità
4. Sicurezza
5. Manutenibilità
6. Portabilità

#### 6. Altri requisiti

1. Requisiti legali
2. Requisiti di localizzazione

#### 7. Architettura del sistema

1. Stack tecnologico
2. Componenti principali
3. Modello dei dati

#### 8. Appendici

1. Priorità dei requisiti
2. Casi d'uso
3. Prototipi delle interfacce
4. Funzionalità escluse

---

## 1. Introduzione

### 1.1 Scopo del documento

Questo documento descrive i requisiti funzionali e non funzionali dell'applicazione web "Poldo", sviluppata dagli studenti dell'ITI G. Marconi. Il documento è rivolto agli sviluppatori, agli stakeholder e ai futuri manutentori del sistema, e serve come punto di riferimento per la progettazione, lo sviluppo e il testing dell'applicazione.

## 1.2 Ambito del progetto

Il progetto Poldo ha l'obiettivo di sviluppare un'applicazione web che permetta a studenti e professori dell'ITI G. Marconi di ordinare cibi e bevande dal bar della scuola prima degli intervalli, ottimizzando i tempi di preparazione e ritiro. Il sistema include funzionalità di gestione ordini, gestione prodotti, e reporting, con diverse modalità di accesso in base al ruolo dell'utente.

## 1.3 Descrizione del sistema

Poldo è un'applicazione web sviluppata per risolvere il problema delle code durante gli intervalli scolastici, ottimizzando il processo di ordinazione e ritiro di cibi e bevande dal bar della scuola. L'applicazione facilita la gestione degli ordini sia dal lato utente (studenti e professori) che dal lato fornitore (gestore del bar), aggregando gli ordini per classe e consentendo una preparazione anticipata e un ritiro efficiente.

## 1.4 Team di sviluppo

Il sistema è stato progettato e sviluppato da un team di studenti dell'ITI G. Marconi:

- **Marta Genovese**
- **Giacomo Marconi**
- **Francesco Miranda**

## 1.5 Glossario

Termine	Definizione
<b>Scheda prodotto</b>	Elemento all'interno di una pagina che descrive un prodotto tramite immagine, etichette, pulsanti per aggiunta e rimozione dal carrello, disponibilità. Sul retro, accessibile tramite tocco sulla scheda, sono disponibili i dettagli del prodotto.
<b>Paninaro</b>	Utente con la possibilità di contrassegnare ogni singolo ordine come validato per comporre l'ordine di classe, inoltre ha la possibilità di visualizzare il QR-Code per ritirare l'ordine.
<b>Classe</b>	Classe di studenti all'interno dell'istituto.
<b>Ordine individuale</b>	Ordine effettuato da un singolo utente (studente o professore).

Termine	Definizione
Ordine di classe	Aggregazione di ordini individuali di una stessa classe, gestito dal paninaro.
Terminale	Totem per l'ordinazione di prodotti installato all'interno della scuola.

## 2. Descrizione generale

### 2.1 Contesto e panoramica

L'ITI G. Marconi ospita centinaia di studenti che, durante gli intervalli, devono acquistare la loro merenda presso il bar della scuola. Questo crea code e congestione che riducono il tempo effettivo di pausa degli studenti e aumentano il carico di lavoro per il personale del bar.

Poldo risolve queste problematiche permettendo a studenti e professori di ordinare in anticipo, con il bar che può preparare gli ordini prima dell'intervallo. Gli ordini vengono poi ritirati da un rappresentante per classe ("paninaro"), riducendo ulteriormente le code e il caos.

### 2.2 Prospettiva del prodotto

Poldo è un prodotto standalone che non sostituisce né si integra con altri sistemi preesistenti nell'istituto. È progettato specificamente per il contesto dell'ITI G. Marconi, ma la sua architettura modulare potrebbe consentire adattamenti per altri istituti in futuro. Il sistema si integra con i servizi di autenticazione Google per sfruttare gli account istituzionali già esistenti.

### 2.3 Funzionalità principali

#### Per gli studenti:

- Effettuare ordini individuali che vengono aggregati per classe
- Visualizzare i prodotti disponibili con dettagli e prezzi
- Per i paninari: validare gli ordini di classe e ritirare i prodotti tramite QR code

#### Per i professori:

- Effettuare ordini individuali
- Visualizzare i prodotti disponibili

#### Per il gestore del bar:

- Gestire il catalogo prodotti (aggiunta, modifica, rimozione)

- Monitorare e gestire gli ordini ricevuti
- Confermare preparazione
- Confermare consegna tramite QR code
- Generare report sulle vendite

#### Per l'amministratore:

- Gestire utenti e ruoli
- Configurare parametri di sistema (turni)
- Accedere a tutte le funzionalità degli altri ruoli

## 2.4 Profili utente

Ruolo	Descrizione	Autenticazione	Responsabilità
Studente	Utente base che può effettuare ordini personali aggregati alla sua classe	Google	Ordinare prodotti
Paninaro	Studente con privilegi estesi per gestire ordini di classe	Google	Validare ordini individuali, gestire ordini di classe, ritirare prodotti
Professore	Utente che può effettuare ordini individuali	Google	Ordinare prodotti
Gestore del Bar	Responsabile della gestione prodotti e ordini	Username/Password	Gestire catalogo, monitorare ordini, confermare consegne
Admin	Supervisore con accesso completo al sistema	Google	Gestire utenti, configurare sistema, accesso completo

## 2.5 Ambiente operativo

L'applicazione sarà:

- Web-based, accessibile tramite browser
- Responsive per l'utilizzo su dispositivi mobili e desktop
- Integrata con i servizi di autenticazione Google

Tecnologie utilizzate:

- **Frontend:** Vue.js
- **Backend:** Node.js
- **DBMS:** MySQL

## 2.6 Vincoli e dipendenze

- L'applicazione richiede una connessione internet funzionante
- Gli utenti devono avere un account Google istituzionale
- Il sistema deve operare con le risorse hardware disponibili nell'istituto
- Il ritiro richiede la capacità di generare e scansionare codici QR

## 2.7 Assunzioni e prerequisiti

- Si assume che tutti gli studenti e i professori abbiano accesso a dispositivi con connessione internet
- Si assume che il bar scolastico abbia almeno un dispositivo per gestire gli ordini e scansionare i QR code
- Si presuppone che i gestori del bar abbiano conoscenze informatiche di base

# 3. Requisiti funzionali

## 3.1 Autenticazione e gestione utenti

ID	Requisito	Descrizione	Priorità
F1.1	Autenticazione Google	Autenticazione tramite Google per studenti, professori e admin	MUST
F1.2	Autenticazione locale	Autenticazione locale (username/password) per i gestori del bar	MUST

ID	Requisito	Descrizione	Priorità
F1.3	Attivazione paninaro	Attivazione del ruolo "Paninaro" tramite token ottenuto a inizio anno	MUST
F1.4	Gestione ruoli admin	Possibilità per gli admin di modificare i ruoli degli utenti	MUST
F1.5	Token JWT	Le sessioni utente devono utilizzare token JWT con scadenza dopo 30 minuti di inattività	SHOULD
F1.6	Gestione terminali	Ordinazione su terminali installati all'interno della scuola	WON'T

## 3.2 Gestione prodotti

ID	Requisito	Descrizione	Priorità
F2.1	Aggiunta prodotti	Aggiunta nuovi prodotti specificando nome, prezzo, immagine, ingredienti, tipo e disponibilità	MUST
F2.2	Modifica prodotti	Modificare le informazioni dei prodotti esistenti	MUST
F2.3	Disponibilità prodotti	Impostare la disponibilità giornaliera dei prodotti	MUST
F2.4	Visibilità prodotti	Nascondere/mostrare prodotti nel menu	SHOULD
F2.5	Prodotti temporanei	Designare prodotti come "temporanei" (validi solo per il giorno corrente)	SHOULD
F2.6	Gestione buoni	Gestione/acquisto di buoni del bar	WON'T

## 3.3 Visualizzazione menu e prodotti

ID	Requisito	Descrizione	Priorità
F3.1	Visualizzazione menu	Visualizzare i prodotti disponibili in un menu con schede prodotto	MUST
F3.2	Dettagli prodotto	Visualizzare i dettagli del prodotto toccando la scheda	MUST
F3.3	Riepilogo ordine	Visualizzare un riepilogo dell'ordine corrente	MUST
F3.4	Filtro prodotti	Consentire agli utenti di filtrare i prodotti per nome, prezzo, tipo, disponibilità e ingredienti	SHOULD
F3.5	Etichette prodotti	Visualizzare etichette grafiche per specificare il tipo di prodotto	SHOULD
F3.6	Suddivisione menu	Suddividere il menu in prodotti normali e temporanei	COULD
F3.7	Preferiti	Salvare prodotti preferiti per un accesso rapido	COULD
F3.8	Template ordini	Salvare fino a tre template di ordine	COULD
F3.9	Personalizzazione prodotti	Rimuovere ingredienti di un determinato prodotto nel proprio ordine	WON'T

## 3.4 Gestione ordini utente

ID	Requisito	Descrizione	Priorità
F4.1	Gestione carrello	Aggiungere/rimuovere prodotti dal carrello	MUST
F4.2	Conferma ordine	Confermare l'ordine per i Paninari	MUST
F4.3	Aggregazione ordini	Aggregare gli ordini individuali in ordini di classe	MUST
F4.4	Validazione ordini	Validare gli ordini individuali per comporre l'ordine di classe per i Paninari	MUST



ID	Requisito	Descrizione	Priorità
F4.5	QR code ritiro	Generare un codice QR univoco per il ritiro dell'ordine	SHOULD
F4.6	Cronologia ordini	Visualizzare la cronologia dei propri ordini	COULD
F4.7	Invio QR professori	Consentire ai professori di inviare il proprio QR code a un paninaro a scelta	COULD

## 3.5 Gestione ordini bar

ID	Requisito	Descrizione	Priorità
F5.1	Visualizzazione ordini	Visualizzare gli ordini suddivisi per classe o per prodotto	MUST
F5.2	Ordini preparati	Marcare gli ordini come "preparati"	MUST
F5.3	Prodotti preparati	Marcare i prodotti come "preparati"	MUST
F5.4	Conferma ritiro	Confermare il ritiro degli ordini tramite scansione del QR code	SHOULD

## 3.6 Reporting e statistiche

ID	Requisito	Descrizione	Priorità
F6.1	Report vendite	Generare report di vendite per finestra di tempo	COULD
F6.2	Esportazione report	Consentire l'esportazione dei report in formato PDF/CSV	COULD
F6.3	Grafici vendite	Visualizzare grafici sull'andamento delle vendite per prodotto	COULD
F6.4	Classifica classi	Calcolare e visualizzare una classifica delle classi per spesa totale e ordini	COULD

ID	Requisito	Descrizione	Priorità
F6.5	Aggiornamento classifica	Aggiornare la classifica delle classi una volta al giorno	COULD

## 4. Requisiti di interfaccia

### 4.1 Interfacce utente

ID	Requisito	Priorità
UI1.1	Interfaccia per l'autenticazione degli utenti (studenti, profoessori, admin, gestori bar)	MUST
UI1.2	Interfaccia per la gestione del catalogo prodotti	MUST
UI1.3	Interfaccia per la visualizzazione e ordinazione dei prodotti	MUST
UI1.4	Interfaccia per la gestione degli ordini (lato utente)	MUST
UI1.5	Interfaccia per la gestione degli ordini (lato bar)	MUST
UI1.6	Interfaccia amministrativa per la gestione utenti	MUST
UI1.7	Interfaccia per la generazione e visualizzazione di report	COULD

### 4.2 Interfacce hardware

ID	Requisito	Priorità
HW1.1	Compatibilità con dispositivi mobili (smartphone, tablet)	MUST
HW1.2	Compatibilità con computer desktop/laptop	MUST
HW1.3	Supporto per la scansione di codici QR tramite fotocamera	SHOULD
HW1.4	Compatibilità con stampanti per l'eventuale stampa di report	COULD

## 4.3 Interfacce software

ID	Requisito	Priorità
SW1.1	Integrazione con l'API di autenticazione Google	MUST
SW1.2	Compatibilità con i principali browser web (Chrome, Firefox, Safari, Edge)	MUST
SW1.3	API RESTful per la comunicazione client-server	MUST
SW1.4	Integrazione con librerie per la generazione e lettura di codici QR	SHOULD

## 4.4 Interfacce di comunicazione

ID	Requisito	Priorità
CM1.1	Comunicazione client-server tramite protocollo HTTPS	MUST
CM1.2	API documentata per eventuali integrazioni future	SHOULD
CM1.3	Websocket per aggiornamenti in tempo reale degli ordini	COULD

# 5. Requisiti non funzionali

## 5.1 Performance

ID	Bisogno	Descrizione	Priorità
NF1.1	Utilizzo simultaneo	Supportare l'utilizzo simultaneo da parte di più utenti	MUST
NF1.2	Tempo di risposta	Tempo di risposta del server inferiore a 2 secondi per operazioni standard	SHOULD
NF1.3	Gestione risorse	Gestione efficiente delle risorse di sistema (CPU, memoria)	SHOULD

ID	Bisogno	Descrizione	Priorità
NF1.4	Ottimizzazione DB	Ottimizzazione delle query al database	SHOULD

## 5.2 Usabilità

ID	Bisogno	Descrizione	Priorità
NF2.1	Interfaccia responsive	Interfaccia responsive compatibile con dispositivi mobili e desktop	MUST
NF2.2	Supporto touch	Supporto per operazioni touch su dispositivi mobili	MUST
NF2.3	Feedback azioni	Feedback delle azioni (conferme, errori)	SHOULD
NF2.4	Design coerente	Interfaccia intuitiva con elementi di design coerenti	SHOULD
NF2.5	Navigazione semplificata	Navigazione semplificata con pochi passaggi per completare un ordine	SHOULD

## 5.3 Affidabilità

ID	Requisito	Descrizione	Priorità
NF3.1	Disponibilità	Il sistema deve essere disponibile durante l'orario scolastico	MUST
NF3.2	Gestione errori	Gestione appropriata degli errori con messaggi informativi	SHOULD
NF3.3	Recupero	Capacità di recupero in caso di interruzioni	SHOULD
NF3.4	Backup	Backup periodico del database	SHOULD

## 5.4 Sicurezza

ID	Requisito	Descrizione	Priorità
NF4.1	Autenticazione sicura	Autenticazione sicura per tutti gli utenti	MUST
NF4.2	Controllo accessi	Controllo degli accessi basato sui ruoli	MUST
NF4.3	Cifratura	Cifratura delle password e dei dati sensibili	MUST
NF4.4	Protezione attacchi	Protezione contro attacchi comuni (XSS, CSRF, SQL Injection)	SHOULD

## 5.5 Manutenibilità

ID	Bisogno	Descrizione	Priorità
NF5.1	Registrazione log	Registrazione log	SHOULD
NF5.2	Documentazione codice	Codice ben documentato e commentato	SHOULD
NF5.3	Architettura modulare	Architettura modulare per facilitare future modifiche	SHOULD
NF5.4	Versioning	Versioning del codice tramite sistema di controllo versione	SHOULD

## 5.6 Portabilità

ID	Requisito	Descrizione	Priorità
NF6.1	Indipendenza piattaforma	Indipendenza dalla piattaforma (funzionamento su diversi sistemi operativi)	MUST
NF6.2	Deployment	Facilità di deployment e configurazione	SHOULD
NF6.3	Compatibilità browser	Compatibilità con diverse versioni dei browser supportati	SHOULD

## 6. Architettura del sistema

### 6.1 Stack tecnologico

- **Frontend:**
  - Vue.js come framework JavaScript
  - HTML, CSS per la struttura e lo stile
  - Responsive design
- **Backend:**
  - Node.js come runtime JavaScript
  - JWT per l'autenticazione
  - Integrazione con Google OAuth
- **Database:**
  - MySQL come DBMS relazionale
- **Deployment:**
  - Server interno dell'istituto

## 7. Appendici

### 7.1 Priorità dei requisiti

I requisiti sono classificati secondo il metodo MoSCoW:

- **MUST:** Requisito obbligatorio, il sistema non funziona senza
- **SHOULD:** Requisito importante ma non critico
- **COULD:** Requisito desiderabile ma non essenziale
- **WON'T:** Requisito non implementato in questa versione