

Indice

1. Obiettivi

2. Requisiti Funzionali

3. Requisiti non Funzionali

4. Design Front-end

5. Design Back-end

Obiettivi

In un periodo in cui i prezzi delle materie prime sono molto volatili, PaninaroNext si propone di aggiungere alla semplice gestione di un ristorante un sistema già usato nei supermercati, automatizzare i prezzi, chi userà PaninaroNext, avrà i prezzi dei suoi piatti modificati automaticamente al prezzo delle materie prime. Il sistema deve permettere di:

- Al cliente:
 - Accedere al menu con dispositivi esterni al sistema (Es. dispositivi mobili in possesso del cliente).
 - Prenotare un posto o prendere take-away ad una determinata ora.
 - Inviare Feedback al ristorante.
 - Pagare online o in locale.
 - Vedere quale materia prima ha influenzato maggiormente il prezzo del piatto comprato.
- Al Ristorante:
 - Visualizzare ordini e transazioni.
 - Visualizzare dati sulle vendite delle materie prime (per vedere quali sono le più utilizzate).
 - Creare ordini (in caso un cliente non abbia la possibilità di ordinare da telefono).
 - Aggiungere materie prime al DB.
 - Aggiungere nuovi oggetti al menu.
 - Modificare il costo di materie prime nel DB.
 - Modificare oggetti nel menu.
 - Cancellare ordini.
 - Modificare la visibilità dei piatti nel menu (i dati dei piatti sono salvati ma non accessibili al cliente).

Requisiti Funzionali

Dalla parte del cliente

- ☐ RF1 Presentazione del sistema:
 - Il sistema deve essere presentato al cliente tramite una pagina Web.
- ☐ RF2 Browser supportati:

- La pagina Web deve essere utilizzabile nei **3 browser piu utilizzati su sistemi mobili** e nei **3 browser piu utilizzati su desktop** in questo momento (Chrome, Safari, Samsung Internet, Edge, Firefox) con circa il 96% della copertura mobile e l'93% della copertura desktop.
- ☐ RF3 Interazioni con la pagina Web:
 - La pagina Web sarà utilizzabile indifferentemente dalle dimensioni e dall'aspect-ratio dello schermo.
- ☐ RF4 Livelli d'accesso del cliente:
 - La pagina Web avrà due livelli d'accesso: 1. Cliente in loco, accederà alla pagina Web attraverso un codice QR presente su ogni tavolo, così facendo il sistema sa anche a che tavolo consegnare l'ordine. 2. Cliente non in loco, accederà ad una pagina Web generica.
- ☐ RF5 In loco:
 - La pagina Web deve permettere ad un cliente in loco ristorante di visualizzare il menu e ordinare un piatto.
- ☐ RF6 Non in loco:
 - La pagina Web generica deve permettere ai clienti non in loco di prenotare un posto al ristorante, visualizzare il menu, ordinare un piatto con take-away.
- ☐ RF7 Feedback:
 - La pagina Web deve permettere ai clienti di inviare feedback al ristorante indifferentemente dal livello d'accesso.
- ☐ RF8 Visualizzazione del menu:
 - La pagina Web deve permettere al cliente di vedere quale materia prima ha influenzato di più il costo di un piatto.
- ☐ RF9 Pagamenti:
 - La pagina Web deve permettere al cliente di poter pagare il servizio ordinato.
- ☐ RF10 Metodi di interazione con il sistema alternativi:
 - In caso il cliente non abbia la possibilità di interfacciarsi al sistema con le funzioni sopra elencate può comunicare con il sistema attraverso lo staff del ristorante.

Dalla parte del ristorante

- ☐ RF11 Accesso admin:
 - Il sistema deve avere un accesso admin per lo staff addetto al sistema.
- ☐ RF12 Visualizzazione dell'andamento dell'attività:
 - Il sistema deve mostrare gli ordini e le transazioni eseguite.
- ☐ RF13 Ordini fuori dal sistema:
 - Il sistema deve permettere allo staff di generare ordini per clienti che non hanno i mezzi di ordinare.
- ☐ RF14 Popolare la lista delle materie prime:
 - Il sistema deve permettere allo staff di aggiungere nuove materie prime nel sistema.
- ☐ RF15 Modificare la lista delle materie prime:
 - Il sistema deve permettere allo staff di modificare i dati delle materie prime nel sistema.
- ☐ RF16 Eliminare materie prime dalla lista:
 - Il sistema deve permettere allo staff di eliminare materie prime dal sistema.
- ☐ RF17 Aggiungere piatti al menu:
 - Il sistema deve permettere allo staff di aggiungere nuovi piatti al menu specificando con quali materie prime viene preparato, e il guadagno atteso.
- ☐ RF18 Modificare piatti esistenti:

- Il sistema deve permettere allo staff di modificare i dettagli di un determinato piatto.
- ☐ RF19 Rimuovere piatti dal menu:
 - Il sistema deve permettere allo staff di rimuovere i dati di un piatto dal DB.
- ☐ RF20 Visibilità di un piatto:
 - Il sistema deve permettere allo staff di "Rimuovere" dal menu un determinato piatto e successivamente aggiungerlo di nuovo al menu visibile ai clienti.
- ☐ RF21 Cancellare ordini:
 - Il sistema deve permettere allo staff di cancellare ordini con la possibilità di rimborso al cliente.

Requisiti Funzionali generali

- ☐ RF22 Prezzo dei piatti:
 - Il sistema dovrà automaticamente modificare i prezzi nel menu in base al costo delle sue materie prime e il guadagno atteso del piatto.
- ☐ RF23 Codici QR:
 - L'applicazione dovrà generare i codici QR univoci per ogni tavolo del ristorante.

Requisiti non Funzionali

- ☐ RNF1 Limitazione di memoria:
 - il sito web non deve occupare più di 4Gb di memoria RAM.
- ☐ RNF2 Uptime:
 - L'applicazione può essere down per motivi legati alla sua struttura per un massimo di 0,547945205% di un anno (2 giorni), non ci prendiamo la responsabilità se l'applicazione smette di funzionare per cali di corrente, mancanza di accesso ad Internet, sabotaggio da parte dello staff o da parte dei clienti.
- ☐ RNF3 Latenza:
 - La latenza del servizio non deve eccedere i 5 secondi per ogni servizio offerto, se l'area del cliente ha disponibilità di traffico limitata, questo requisito cade.
- ☐ RNF4 Lingua:
 - L'interfaccia dell'applicazione deve essere disponibile sia in inglese che in italiano.
- ☐ RNF5 Accessibilità:
 - L'applicazione deve utilizzare un font ad alta leggibilità (Helvetica, Courier, Arial, Verdana)
- ☐ RNF6 Privacy:
 - L'applicazione non dovrà ritenere dati sensibili sui clienti, ma solo sugli ordini per poter assistere lo staff durante l'acquisto delle materie prime

Front-end

Ecco come sarà la parte grafica dell'applicazione:

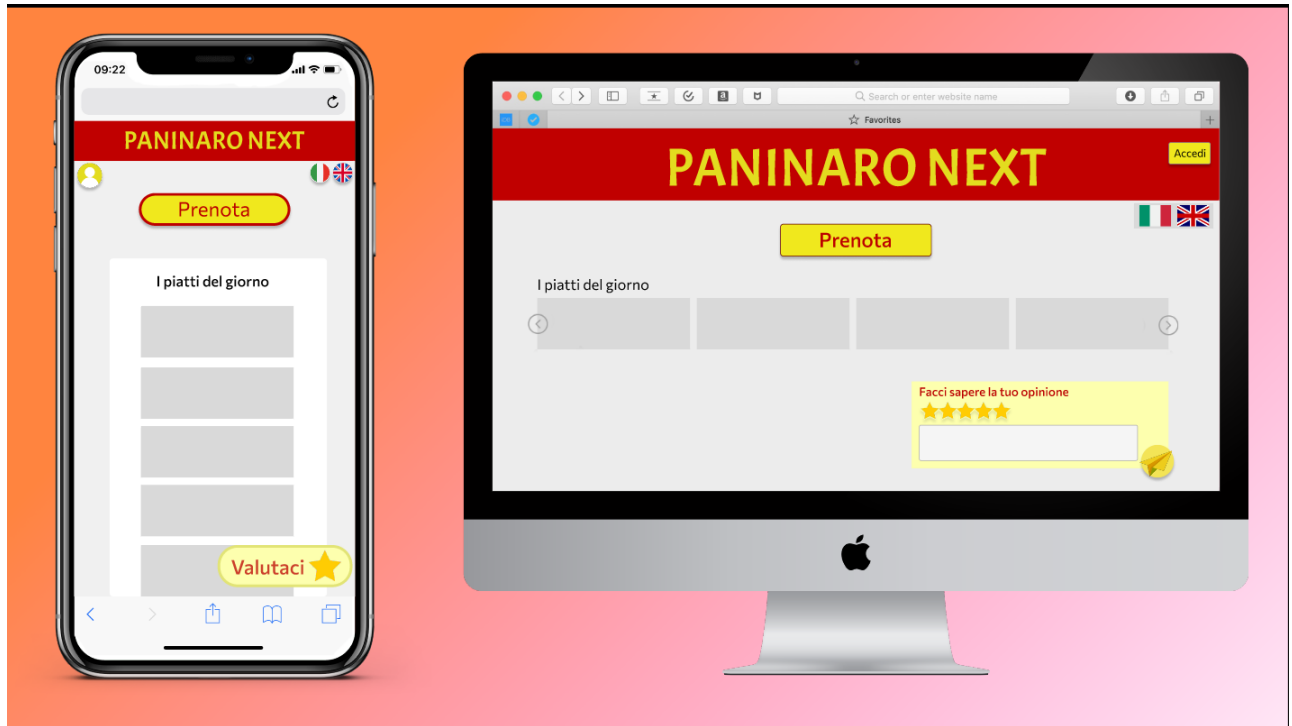
- Cliente che vuole prenotare o prentere un take-away (pagina generica).



- Cliente in loco dopo aver scannerizzato il codice QR del proprio tavolo, con possibilita di pagamento.



- Cliente ha la possibilità di mandare un feedback al ristorante.



- Login dello staff.



- Statistica del ristorante e modifica del database.



Back-end

Le tecnologie utilizzate per l'applicazione sono:

- [MongoDB](#):
 - Un sistema di gestione dati non relazionale.
- [NodeJS](#):
 - Permette di utilizzare JavaScript per costruire la parte logica dell'applicazione da parte del server, la parte logica dell'applicazione dovrà far comunicare il front-end con la base di dati in MongoDB.
- [goqr.me](#):
 - Utilizzata per creare i QR code per i tavoli.
- [Paypal](#)
 - Utilizzato come metodo di pagamento online.