**Progetto:**

**Paninaro Next**

**Titolo del documento:**

**Obiettivi e Requisiti**

**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 gia usato nei supermercati, automatizzare i prezzi. Paninaro Next 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.
* Inviare Feedback al ristorante.
* Pagare online.

Al Ristorante:

* Visualizzare ordini e transazioni.
* Aggiungere materie prime al DB.
* Agigungere nuovi piatti al menu.
* Modificare il costo di materie prime nel DB.
* Modificare oggetti nel menu.
* Cancellare ordini.

Requisiti Funzionali

Dalla parte del cliente:

* **RF1 Presentazione del sistema**: Il sistema deve essere prestantato al cliente tramite una pagina Web.
* **RF2 Browser supportati**: La pagina Web deve essere utilizzabile nei 3 browser piu utilizzati su sitestemi 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 Accesso del cliente**: La pagina Web avrà come accesso per il cliente una pagina Web attraverso un codice QR presente su ogni tavolo, cosi facendo il sistema sa anche a che tavolo consegnare l'ordine.
* **RF5 Ordini cliente**: La pagina Web deve permettere ad un cliente in loco di visualizzare il menu e ordinare un piatto.
* **RF6 Feedback**: La pagina Web deve permettere ai clienti di inviare feedback al ristorante.
* **RF7 Pagamenti**: La pagina Web deve permettere al cliente di poter pagare il servizio ordinato con contanti o Paypal.

Dalla parte del ristorante:

* **RF8 Accesso gestore**: Il sistema deve avere un accesso admin per lo staff addetto al sistema.
* **RF9 Visualizzazione dell'andamento dell'attività**: Il sistema deve mostrare gli ordini e le transazioni eseguite.
* **RF10 Popolare la lista delle materie prime**: Il sistema deve permettere allo staff di aggiungere nuove materie prime nel sistema.
* **RF11 Modificare la lista delle materie prime**: Il sistema deve permettere allo staff di modificare i dati delle materie prime nel sistema.
* **RF12 Eliminare materie prime dalla lista**: Il sistema deve permettere allo staff di eliminare materie prime dal sistema.
* **RF13 Aggiungere piatti al menu**: Il sistema deve permettere allo staff di agigungere nuovi piatti al menu specificando con quali
* materie prime viene preparato.
* **RF14 Modificare piatti esistenti**: Il sistema deve permettere allo staff di modificare i dettagli di un determinato piatto.
* **RF15 Rimuovere piatti dal menu**: Il sistema deve permettere allo staff di rimuovere i dati di un piatto dal DB.
* **RF16 Cancellare ordini**: Il sistema deve permettere allo staff di cancellare ordini con la possibilita di rinborso al cliente.

Requisiti non Funzionali

* **RNF1 Limitazione di memoria**: Il sito web non deve occuprare piu di 4Gb di memoria RAM.
* **RNF2 Uptime**: L'applicazione puo essere down per motivi legati alla sua struttura per un massimo di 0,547945205% di un anno (2 giorni).
* **RNF3 Latenza**: La latenza del servizio non deve eccedere i 5 secondi per ogni servizio offerto, se il cliente ha disponibilità di traffico limitata, questo requisito cade.
* **RNF4 Accessibilità**: L'applicazione deve utilizzare un font ad alta leggibilità (Helvetica, Courier, Arial, Verdana)
* **RNF5 Privacy**: L'applicazione non dovrà ritenere dati sensibili sui clienti, ma solo sugli ordini.

Front-end

Ecco come sarà la parte grafica dell’applicazione:

* Cliente che vuole ordinare

Immagine che contiene testo, schermata, software, multimediale

Descrizione generata automaticamente

* Cliente che paga

Immagine che contiene testo, schermata, multimediale, software

Descrizione generata automaticamente

* Cliente che può mandare feedback al ristorante

Immagine che contiene testo, schermata, multimediale, software

Descrizione generata automaticamente

* Login del gestore

Immagine che contiene testo, multimediale, schermata, software

Descrizione generata automaticamente

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 dovra far comunicare il front-end con la base di dati in MongoDB.
* **Paypal**: Utilizzato come metodo di pagamento.