**Requirements Analysis Document**

**• Introduction**

*• Purpose of the system*

Il sistema punta a tenere traccia delle finanze di una pasticceria. L’obbiettivo principale è calcolare il food cost per ogni attività registrata al sistema e suggerire loro prezzi di vendita per elevare i guadagni della stessa. Il sistema sarà di supporto nella gestione dell’inventario del magazzino e nella gestione delle ricette.

*• Scope of the system*

L’idea è quella di gestire per mezzo di un sistema informatico le finanze relative ad una pasticceria, ramo in cui le materie prime e manodopera sono tra le attività che richiedono più spese.

*• Objectives and success criteria of the project*

Ci si concentra ad isolare l’azienda dal dispendioso impegno di realizzare conti e gestire nel più ottimizzato modo l’organizzazione di finanze, prodotti e ricette.

*• Definitions, acronyms, and abbreviations*

*• References*

*• Overview*

**• Current system**

Oggi, la maggior parte delle pasticcerie, non utilizza software gestionali legati al food-cost per gestire le proprie finanze. Tante operazioni ripetitive, se effettuate, vengono svolte da persone che potrebbero occupare il loro tempo per compiti molto più importanti ed aumentare la produzione riuscendo magari a soddisfare clienti che fino ad ora sono stati respinti per il poco tempo a disposizione. Non tenere conto degli sprechi o degli scarti è un errore frequente quando si analizzano le spese.

**• Proposed system**

*• Overview*

Il sistema proposto sarà ospitato su un web server che fornirà ad ogni utente connesso i servizi di cui ha bisogno.

*• Functional requirements*

Il sistema dovrà essere in grado di:

• Gestire l’inventario dei prodotti presenti in magazzino

**• RF\_1)** Si potranno salvare nel sistema tutti gli ingredienti utilizzati nel magazzino con le relative operazioni di aggiornamento.

**• RF\_2)** Il sistema memorizzerà informazioni utili al calcolo delle finanze sugli stessi (prezzo, quantità ecc.).

• Gestire il ricettario

**• RF\_3)** L’utente finale potrà memorizzare all’interno del sistema ricette predefinite selezionando tra i prodotti presenti in magazzino quali fanno parte della ricetta.

**• RF\_4)** Saranno possibili tutte le operazioni di aggiornamento.

• Gestire entrate e uscite economiche

**• RF\_5)** Verranno generati dei grafici sul guadagno (mensile, annuale) rapportando i costi di un prodotto rispetto al suo costo di vendita.

**• RF\_6)** L’utente finale dovrà selezionare il tipo di ricetta, la quantità di produzione e quella di vendita.

**• RF\_7)** Il sistema recupera i dati salvati nel ricettario e ne calcolerà il costo della ricetta.

• Calcolare il costo complessivo di un prodotto finale e/o ricetta

**• RF\_8)** Avendo a disposizione i prezzi delle singole materie prime e la composizione di una certa ricetta il sistema sarà in grado di definire un prezzo di produzione per ogni ricetta memorizzata nel sistema.

**• RF\_9)** L’utente finale potrà definire una percentuale di carico sul prodotto. Il sistema suggerirà quindi un prezzo di vendita.

*• Nonfunctional requirements*

*• Usability*

*• Reliability*

**NF\_1)** Il riavvio del sistema in caso di fallimento è consentito;  
**NF\_2)** Ogni password associata ad un account deve essere alfanumerica, di almeno 8 caratteri di cui uno speciale e una lettera maiuscola;  
**NF\_3)** La password viene criptata secondo algoritmi di crittografia scelti in fase di implementazione;  
**NF\_4)** Il protocollo usato per l’accesso al sito sarà HTTPS, ovvero il sito web girerà su protocollo SSL;

*• Performance*

**NF\_5)** Deve garantire un tempo di risposta per la generazione dei grafici delle finanze che non superi i 60 s;  
**NF\_6)** Deve garantire un tempo di risposta per tutte le operazioni aggiornamento sul DB inferiore ad 1 s;  
**NF\_7)** Deve garantire scalabilità in termini di numero di clienti;

*• Supportability*

*• Implementation*

**NF\_8)** Implementa la parte server in linguaggio Java;  
**NF\_9)** Implementa la parte client in HTML5 e JSP page;  
**NF\_10)** Formatta i dati dell’interfaccia utente utilizzando CSS e frame-work comeBootstrap e AngularJS;  
**NF\_12)** Gestisce i dati con l’uso del DBMS MySQL;

*• Interface*

**NF\_13)** L’unico formato accettato per l’upload delle fatture in formato elettronico è .xlsx in cui il numero di colonne è pari a 4 nel seguente ordine: codice identificativo, nome prodotto, quantità e prezzo;

*• Packaging*

*• Legal*

*• System models*

*• Scenarios*

***GESTIONE FINANZE***

Ogni mattina Giuseppe (G), amministratore del sistema, si collega con il suo laptop, tablet o pc al sistema digitando nella barra degli indirizzi di un suo comune browser il nome del sistema. Per permettere l’autenticazione e mostrare i dati relativi alla azienda di G, il sistema mostra un form con due campi di testo nei quali inserire una username e una password. G inserisce la sua username e la sua password e clicca sul pulsante <LOGIN>. Il sistema non riconosce i dati e mostra un alert avvisando G che i dati immessi sono errati. G ripete l’inserimento dei dati. Il sistema riconosce l’accesso e mostra l’area personale di G come indicato in **figura 1.1**. G clicca sul pulsante <GESTIONE FINANZE>. Il sistema mostra una interfaccia che presenta due grafici (uno per le entrate, l’altro per le uscite della sua attività), due pulsanti dedicati all’inserimento di una nuova entrata e/o uscita occasionale e una tabella sulle informazioni dei prodotti con i relativi prezzi di acquisto e di vendita come rappresentato in **figura 1.2**. G sceglie di inserire una nuova entrata. Il sistema mostra un form con 3 campi di testo nei quali inserire descrizione, data e importo. G inserisce i dati e clicca sul pulsante salva per memorizzarli. Il sistema aggiorna i grafici con la nuova entrata. G osservando il grafico si accorge di dover aumentare il prezzo di un prodotto. Nella colonna “Prezzo di vendita” della tabella, G clicca sul pulsante + per aumentare il suo valore di una unità. G spunta la checkbox per vedere in anteprima l’aggiornamento del grafico.

**PRENOTAZIONE**

Carlo (C) vuole prenotare un prodotto. Accede al sistema con le sue credenziali inserendo una username e una password. C non è registrato e il sistema non riconosce i dati. C si registra compilando il form come in **figura 2.1.** Cinserisce un indirizzo email non valido. Il sistema avvisa C che la mail non è riconosciuta. C ripete l’inserimento. C accede al sistema il quale gli mostra l’elenco di tutte le pasticcerie registrate come mostrato in **figura 2.2**. C seleziona una pasticceria e il sistema mostra in una tabella tutti prodotti che quella pasticceria propone indicando il relativo prezzo come in **figura 2.3**. C può leggere dettagli sul prodotto cliccando su di esso. Il sistema mostra un paragrafo dedicato alla descrizione del prodotto e alla sua composizione. In fondo viene mostrato il numero di telefono della pasticceria (nel caso in cui C voglia prenotare chiamando) e un form con 2 caselle nelle quali C inserisce la quantità desiderata (Es. 2) e la data di ritiro (Es. 12/10/2019) e un pulsante <INVIA PRENOTAZIONE> che memorizza nel sistema la prenotazione. C pagherà poi in negozio.

**INVENTARIO**

Ogni mattina Mario (M), magazziniere, accede al sistema. Una tabella indica ad M tutti i prodotti presenti in magazzino con i relativi attributi come mostrato in **figura 3.1**. Farina 00 e uova sono sotto scorta e il sistema imposta un background di colore rosso. M vuole cercare la nutella e immette nella barra di ricerca laterale il nome del prodotto. Arrivano nuovi prodotti nel magazzino e M li memorizza cliccando sul pulsante <INSERISCI NUOVO PRODOTTO>. Il sistema mostra un form dove inserire il nome del prodotto, un codice identificativo, il prezzo e la quantità come mostrato in **figura 3.2**. M sbaglia l’inserimento di un prodotto e deve eliminarlo. Clicca sull’icona del cestino. Il sistema chiede se eliminare definitivamente il prodotto con una alert di conferma. Entro fine giornata M deve ordinare i prodotti mancanti e/o nuovi prodotti. Consulta la sezione Prenotazioni la quale mostra ad M (come in **figura 3.3**) i prodotti sotto scorta da acquistare per la preparazione delle prenotazioni in sospeso.

***RICETTARIO***

Pasquale (P) è il pasticciere del laboratorio e per preparare le sue ricette accede al sistema con le sue credenziali. Il sistema riconosce il suo ruolo e gli mostra una tabella con tutte le ricette memorizzate indicando per ognuna di esse il nome e il tempo medio impiegato per la realizzazione e un segno “+” come in **figura 4**. P clicca sul segno “+” e il sistema mostra la composizione di quella ricetta indicando tutti i prodotti e le materie primi utili ai fini della realizzazione del prodotto. P inizia la realizzazione di una ricetta e al termine clicca sul pulsante <FATTA> per permettere al sistema di aggiornare le quantità del magazzino. P vuole modificare la quantità di farina della ricetta della ricotta e pera. P clicca sull’icona della matita. Il sistema permette la modifica dei campi del prodotto. Al termine P clicca sul pulsante <SALVA> per il salvataggio dei nuovi dati.

Figura 1.1

Figura 1.2

Figura 2.1

Figura 2.2

Figura 2.3

Figura 3.1

Figura 3.2

Figura 3.3

Figura 4

*• Use case model*

*• Object model*

*• Dynamic model*

*• User interface navigational paths and screen mock-ups*

**• Glossary**