

Sommario

[**1.** **Dominio del problema** 1](#_Toc53160477)

[**2.** **Obiettivi** 2](#_Toc53160478)

[**3.** **Requisiti Funzionali** 2](#_Toc53160479)

[***3.1 RF 1 - Gestione Registrazione*** 2](#_Toc53160480)

[***3.2 RF 2 - Gestione Autenticazione*** 2](#_Toc53160481)

[***3.3 RF 3 - Gestione Utenti*** 2](#_Toc53160482)

[***3.4 RF 4 – Gestione prodotti in vendita*** 3](#_Toc53160483)

[**4.** **Requisiti non Funzionali** 4](#_Toc53160484)

[**4.1 RNF 1 - Usabilità** 4](#_Toc53160485)

[**4.2 RNF 2 - Affidabilità** 5](#_Toc53160486)

[**4.3 RNF 3 - Performance** 5](#_Toc53160487)

[**4.4 RNF 4 – Supportabilità** 5](#_Toc53160488)

[**4.5 RNF 5 - Implementazione** 6](#_Toc53160489)

[**4.6 RNF 6 - Legali** 6](#_Toc53160490)

[**4.7 RNF 7 - Sicurezza** 6](#_Toc53160491)

[**5.Ambiente di Destinazione** 7](#_Toc53160492)

1. **Dominio del problema**

Oggigiorno si sta sviluppando sempre più la tendenza ad effettuare acquisti online secondo le esigenze delle persone. L’espressione commercio elettronico, in inglese e-commerce, si riferisce all’insieme delle transazioni per la commercializzazione di beni e servizi tra produttore e consumatore, realizzate tramite Internet; quindi all’acquisto di beni e servizi attraverso il World Wide Web ricorrendo a server sicuri, con servizi di pagamento online, come ad esempio le autorizzazioni per il pagamento con carta di credito.

Il problema del nostro cliente è quindi quello di voler rendere visibile la propria attività e di ampliare la sua clientela. L’obiettivo che ci poniamo è quello di fornire una piattaforma online per l’acquisto di carni e salumi. L’e-commerce “NaFella” offre la possibilità all’utenza di visionare il catalogo di carni e di effettuare uno o più acquisti. La possibilità di acquistare uno o più carni è riservata solo agli utenti registrati. Il gestore ha invece la possibilità di inserire o rimuovere beni e mettere in offerta determinati articoli. L’obiettivo finale è quello di sviluppare un sistema efficace ed efficiente, a vantaggio del nostro cliente, e che sia fruibile all’utenza.

Per questo ci viene in aiuto NaFella, il sistema da noi pensato.

1. **Obiettivi**

Gli obiettivi del sistema NaFella sono:

1. Fornire un’interfaccia per ottenere informazioni relative ai singoli prodotti in vendita
2. Fornire un’infrastruttura che dia la possibilità all’utente di potersi registrare e usufruire del servizio di acquisto
3. Fornire un’interfaccia che permetta l’acquisto dei prodotti
4. Fornire un’interfaccia che permetta al gestore l’inserimento di informazioni relative ai prodotti che verranno poi inseriti nel catalogo
5. Fornire un’interfaccia che permetta al gestore di modificare i prezzi degli articoli;
6. Fornire un’interfaccia che permetta al gestore di manipolare il magazzino;
7. Permettere di ridurre file d’attesa presso il negozio
8. Evitare al cliente di esso esposto a assembramenti potenzialmente pericolosi
9. Semplicità di utilizzo anche agli utenti meno esperti
10. **Requisiti Funzionali**

Analizzando il servizio necessario alla visualizzazione e/o all’acquisto di uno prodotto presente nel catalogo, abbiamo individuato tre tipologie principali di utenti del sistema NaFella ossia amministratore, utente registrato e utente non registrato.

Le funzionalità offerte dal sistema in funzione alla tipologia d’utente sono:

## 3.1 RF 1 - Gestione Registrazione

Questa funzionalità consente ai visitatori di registrarsi a NaFella inserendo i propri dati.

**Attore: Utente**

**RF 1.1 - Registrazione utente:** Questa funzionalità permette all’ utente di registrarsi al sistema in modo da poter effettuare il login per gli accessi futuri.

## 3.2 RF 2 - Gestione Autenticazione

Questa funzionalità consente di gestire l’autenticazione degli utenti a “NaFella”. Permette di:

**RF 2.1 - Login:** questa funzionalità permette di far effettuare l’accesso al sistema autenticandosi ed avendo a disposizione le varie funzionalità offerte.

**RF 2.2 - Logout:** questa funzionalità permette di uscire dal sistema disconnettendosi.

## 3.3 RF 3 - Gestione Utenti

Questa funzionalità raccoglie tutte le operazioni necessarie per gestire gli utenti.

**Attore: Admin**

**RF 3.1 - Rimozione utente:** questa funzionalità permette di rimuovere un'utente da “NaFella“ cancellando il proprio profilo e le relative informazioni dal sistema.

**RF 3.2 – Ricerca utente:** questa funzionalità permette di ricercare un'utente di “NaFella” utilizzando un ordinamento in base ad un determinato attributo.

**Attore: Utente**

**RF 3.3 - Visualizza profilo:** questa funzionalità permette all'utente di accedere alla schermata di visualizzazione dei dati del proprio profilo.

**RF 3.4 – Inserimento indirizzo:** questa funzionalità permette all'utente di inserire un indirizzo.

**RF 3.5 – Visualizzazione indirizzo:** questa funzionalità permette all'utente di accedere alla schermata di visualizzazione degli indirizzi.

**RF 3.6 – Modifica indirizzo:** questa funzionalità permette all'utente di modificare un indirizzo.

**RF 3.7 – Rimozione indirizzo:** questa funzionalità permette all'utente di rimuovere un indirizzo.

**RF 3.8 – Visualizzazione ordine:** questa funzionalità permette all'utente di accedere alla schermata di visualizzazione degli ordini.

**RF 3.9 – Cancellazione ordine:** questa funzionalità permette all'utente di cancellare lo storico di un ordine effettuato.

## 3.4 RF 4 – Gestione prodotti in vendita

Questa funzionalità comprende tutte le operazioni che permettono di gestire i prodotti.

**Attore: Admin**

**RF 4.1 – Inserimento prodotto:** questa funzionalità permette all’amministratore di inserire un prodotto all’interno del catalogo.

**RF 4.2 - Visualizzazione prodotto:** questa funzionalità permette di visualizzare tutti i prodotti presenti all’interno del catalogo.

**RF 4.3 - Modifica prodotto:** questa funzionalità permette di modificare le informazioni relative ad un prodotto all’interno del catalogo.

**RF 4.4 - Rimozione prodotto:** questa funzionalità permette di rimuovere un prodotto all’interno del catalogo cancellandone le relative informazioni dal sistema.

**RF 4.5 - Ricerca prodotto:** questa funzionalità permette di ricercare rapidamente un prodotto utilizzando un ordinamento in base ad un determinato attributo.

**Attore: Utente non registrato**

**RF 4.6 - Visualizza catalogo:** questa funzionalità consente la visualizzazione di tutti gli strumenti musicali presenti sulla piattaforma.

**RF 4.7 - Visualizza informazioni relative ai singoli prodotti:** questa funzionalità permette all’ utente di visualizzare le informazioni relative ad un singolo prodotto selezionato.

**RF 4.8 – Aggiunta al carrello di un prodotto:** questa funzionalità permette ad un utente non registrato di andare ad aggiungere un prodotto presente nel catalogo all’interno del carrello.

**Attore: Utente registrato**

**RF 4.9 - Visualizza catalogo:** questa funzionalità consente la visualizzazione di tutti gli strumenti musicali presenti sulla piattaforma.

**RF 4.10 - Visualizza informazioni relative ai singoli prodotti:** questa funzionalità permette all’ utente di visualizzare le informazioni relative ad un singolo prodotto selezionato.

**RF 4.11 - Aggiunta al carrello di un prodotto:** questa funzionalità permette ad un utente non registrato di andare ad aggiungere un prodotto presente nel catalogo all’interno del carrello.

**RF 4.12 - Acquisto prodotto:** questa funzionalità permette di acquistare un determinato prodotto selezionato presente in catalogo.

# **Requisiti non Funzionali**

I requisiti non funzionali descrivono aspetti del sistema che non sono legati direttamente alle funzionalità del sistema, abbiamo due macrocategorie di requisiti non funzionali: Qualità e Vincoli.

## **RNF 1 – Usabilità**

\*\*\*\*\*[Responsive ovvero si adatterà gaficamente ad ogni tipo di dispositivo dal quale verrà usato ]

Il sistema deve rispettare i tre principi di usabilità, ovvero apprendibilità (la facilità con cui i nuovi utenti possono iniziare un’interazione efficace e raggiungere le massime prestazioni), flessibilità (si riferisce alle modalità in cui l’utente e il sistema si scambiano informazioni) e robustezza (il livello di supporto fornito all’utente nel determinare un comportamento di successo rispetto ai suoi obiettivi). Per ottenere questi risultati:

* Devono essere ridotti al minimo sia il golfo dell’esecuzione che quello della valutazione: il sistema a questo proposito deve mostrare chiaramente i vari pulsanti per le opzioni in ogni schermata, senza dare spazio ad altre possibili scelte che l’utente può immaginare.
* Dev’essere rispettato il paradigma WYSIWYG (What You See Is What You Get) della manipolazione diretta: “NaFella”, infatti, deve rendere diretta ed immediata l’interpretazione dell’interfaccia grafica da parte dell’utente, mostrando in ogni schermata solo le operazioni disponibili per quel contesto.
* “NaFella” soddisferà i principi di usabilità sfruttando anche due principi dell’ergonomia: la disposizione dei controlli sul display, raggruppati per funzione e sequenza d’uso e l’uso sapiente dei colori, rendendo chiara ed efficace l’interazione.
* Il sistema non deve essere percepito come dispersivo in modo da rispettare la predicibilità e la recuperabilità delle operazioni effettuate dall’utente, per questo la massima sequenza di operazioni permesse conterà al più 4 passaggi tra le schermate del sistema.
* Il sistema sarà infine testato su un campione di potenziali utenti per avere una valutazione oggettiva, basando tale test su metriche definite come il numero di secondi che intercorrono tra un’operazione e l’altra ed il numero di errori commessi per portare a termine un compito.

## **4.2 RNF 2 - Affidabilità**

Il sistema è basato sull’architettura Client-Server quindi entrambi le componenti devono essere affidabili ed essere in grado di poter mantenere i propri dati anche in caso di guasti (problemi di fornitura elettrica, usura dell'hardware del server, attacchi informatici, problemi legati al browser, interruzione della connettività alla rete ecc.).

Il sistema deve gestire regolarmente un disservizio, quindi ripristinare lo stato precedente al guasto. Il sistema dev’essere inoltre usufruibile 24 ore su 24.

Deve essere possibile pianificare dei backup periodici del database con cadenza trimestrale. In caso di guasto, un riavvio del sistema è tollerabile.

## **4.3 RNF 3 - Performance**

Il sistema deve essere in grado di gestire un numero elevato \*\*\*\*[ inserire il numero preciso ]di utenti contemporanei senza problemi.

Essendo sotto carico, il sistema deve avere i tempi di risposta all’incirca di 1 secondo per qualsiasi operazione effettuata dall’utenza.

Il sistema sarà visualizzabile su diverse tipologie di browser allo stesso modo utilizzando la grafica responsive, grazie alla tecnologia Bootstrap che permette appunto al layout delle pagine web di regolarsi dinamicamente, tenendo conto delle caratteristiche del dispositivo utilizzato, sia esso desktop, tablet o smartphone.

## **4.4 RNF 4 – Supportabilità**

“NaFella” verrà implementato seguendo un’architettura three-tier (MVC), in questo modo, ciascuno dei tre moduli può essere modificato o sostituito indipendentemente dagli altri conferendo manutenibilità all'applicazione.

Si realizzerà inoltre una matrice di tracciabilità dei requisiti che consentirà di capire l'impatto di una eventuale modifica al sistema.

## **4.5 RNF 5 - Implementazione**

La parte back-end del sistema, verrà implementata in linguaggio Java con l’uso del framework Eclipse. Per la persistenza dei dati verrà utilizzato il database relazionale MySQL. Il front-end verrà realizzato con il supporto di Bootstrap, HTML5, CSS3 e jQuery.

La scelta di utilizzare le suddette tecnologie è stata guidata dalle seguenti motivazioni:

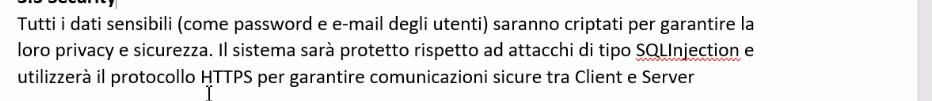
1. Java è un [linguaggio di programmazione ad alto livello](https://it.wikipedia.org/wiki/Linguaggio_di_programmazione_ad_alto_livello), [orientato agli oggetti](https://it.wikipedia.org/wiki/Programmazione_orientata_agli_oggetti) e a [tipizzazione statica](https://it.wikipedia.org/wiki/Tipizzazione_statica), che si appoggia sull'omonima [piattaforma](https://it.wikipedia.org/wiki/Piattaforma_Java) software di [esecuzione](https://it.wikipedia.org/wiki/Esecuzione_(informatica)), specificamente progettato per essere il più possibile indipendente dalla [piattaforma](https://it.wikipedia.org/wiki/Piattaforma_(informatica)) [hardware](https://it.wikipedia.org/wiki/Hardware) di [esecuzione](https://it.wikipedia.org/wiki/Esecuzione_(informatica))
2. Bootstrap è una raccolta di strumenti [liberi](https://it.wikipedia.org/wiki/Software_libero) per la creazione di [siti](https://it.wikipedia.org/wiki/Sito_web) e [applicazioni](https://it.wikipedia.org/wiki/Applicazione_web) per il [Web](https://it.wikipedia.org/wiki/Web). Essa contiene [modelli](https://it.wikipedia.org/wiki/Template) di progettazione basati su [HTML](https://it.wikipedia.org/wiki/HTML) e [CSS](https://it.wikipedia.org/wiki/CSS), sia per la [tipografia](https://it.wikipedia.org/wiki/Tipografia), che per le varie componenti dell'interfaccia, come [moduli](https://it.wikipedia.org/wiki/Form), pulsanti e navigazione, così come alcune estensioni opzionali di [JavaScript](https://it.wikipedia.org/wiki/JavaScript).
3. jQuery è una [libreria](https://it.wikipedia.org/wiki/Libreria_(software)) [JavaScript](https://it.wikipedia.org/wiki/JavaScript) per [applicazioni web](https://it.wikipedia.org/wiki/Applicazione_web), distribuita come [software libero](https://it.wikipedia.org/wiki/Software_libero), con l’obiettivo di semplificare la selezione, le manipolazione, la gestione degli eventi e l’animazione di elementi DOM in pagine HTML, nonché semplificare l'uso di funzionalità [AJAX](https://it.wikipedia.org/wiki/AJAX), la gestione degli eventi e la manipolazione dei [CSS](https://it.wikipedia.org/wiki/CSS).
4. Tutto lo stack si basa sul Model-View-Control, che offre la possibilità di separare la [logica di presentazione](https://it.wikipedia.org/wiki/Livello_di_presentazione#Altro) dei dati dalla [logica di business](https://it.wikipedia.org/wiki/Logica_di_business). Questo pattern si posiziona nel livello logico o di business e di presentazione in una [architettura multi-tier](https://it.wikipedia.org/wiki/Architettura_multi-tier).

## **4.6 RNF 6 - Legali**

“NaFella” è realizzato in maniera tale da garantire il rispetto delle norme vigenti sulla privacy, facendo riferimento alla Legge 31 dicembre 1996, n. 675 del Parlamento italiano, relativa alla tutela delle persone e di altri soggetti rispetto al trattamento dei dati personali.

## **RNF 7 – Sicurezza**

Protocoolo HTTPS PER d



Il sistema prevede un modulo di autenticazione, in modo da negare l’accesso da parte degli utenti non autorizzati. Inoltre è previsto L'MD5 \*\*\*[modificare con una descrizione piu approfondita ]ossia una funzione Hash crittografica che ci permette ancora di più di salvaguardare il sistema e i dati dei relativi utenti, durante la fase di autenticazione abbiamo il classico inserimento della password, che poi verrà confrontata attraverso un meccanismo interno di conversione con la stringa in formato MD5 all’interno del Database.

Il sistema dovrà consentire il monitoraggio degli accessi effettuati in un determinato arco temporale, all’interno di ogni sottosistema, con l’indicazione delle operazioni svolte.

I meccanismi di monitoraggio delle attività (log) tracceranno, dunque, le interazioni dell’utente, la data e l’operazione svolta con possibilità di visualizzazione o di stampa.

Il sistema deve rispettare gli standard di sicurezza OWASP, tecniche necessarie per la qualità del codice dove si va a rispettare il principio della “privacy by design” o di “protezione dei dati fin dalla progettazione”.

# **5.****Ambiente di Destinazione**

L’ambiente di destinazione del Software sarà senza nessun dubbio il Web, per aumentare la sua scalabilità e la facile distribuzione.

Essendo un’applicazione di tipo Web essa potrà essere accessibile da ogni tipologia di dispositivo: personal computer, smartphones e tablet collegati alla rete.

Per garantire le caratteristiche responsive dell’interfaccia si adotteranno appositi framework.\*\*\*modificare