1. DIAGRAMMA DEI RUOLI

* Utente generico: generalizzazione di tutti gli utenti del sistema; si tratta di un qualunque utente (registrato e no), che si trova a navigare nel nostro sito web con la possibilità di scorrere tra le pagine, ricercare prodotti, registrarsi o accedere.
* Utente guest: è un utente che non ha effettuato la registrazione, ma può farlo in qualunque momento.
* Utente registrato: può effettuare il login in qualunque momento, visualizzare il proprio profilo e carrello, finalizzare acquisto.
* Admin: colui che gestisce la web app: rimozione ed aggiunta prodotti, gestione ordini, visualizzazione degli utenti.

1. REQUISITI FUNZIONALI
2. Utente registrato:
   1. RF 1: Il sistema deve fornire la possibilità di connettersi alla piattaforma.
   2. RF 2: Il sistema deve fornire la possibilità di effettuare il login tramite la compilazione di un form dedicato.
   3. RF 3: Il sistema deve fornire la possibilità di effettuare il log out dal proprio profilo tramite un bottone dedicato, che farà scadere la sessione attuale.
   4. RF 4: Il sistema deve fornire la possibilità di consultare l’area personale dell’utente, attraverso una pagina dedicata.
   5. RF 5: Il sistema deve fornire la possibilità di modificare la password attraverso un campo dedicato.
   6. RF 6: Il sistema deve fornire la possibilità di visualizzare le categorie di tutti i prodotti disponibili sul sito.
   7. RF 7: Il sistema deve fornire la possibilità di visualizzare i prodotti nelle specifiche pagine di ogni categoria.
   8. RF 8: Il sistema deve fornire la possibilità di ricercare un prodotto in modo veloce, attraverso una barra di ricerca.
   9. RF 9: Il sistema deve fornire la possibilità di inserire prodotti nel proprio carrello.
   10. RF 10: Il sistema deve fornire la possibilità di eliminare prodotti dal proprio carrello.
   11. RF 11: Il sistema deve fornire la possibilità di visualizzare i prodotti nel carrello col rispettivo totale da pagare.
   12. RF 12: Il sistema deve fornire la possibilità di scegliere il metodo di pagamento.
   13. RF 13: Il sistema deve fornire la possibilità scegliere il metodo di pagamento.
   14. RF 14: Il sistema deve fornire la possibilità di finalizzare acquisto tramite bottone specifico.
3. Utente guest:
   1. RF 15: Il sistema deve fornire la possibilità di connettersi alla piattaforma.
   2. RF 16: il sistema deve fornire la possibilità di compilare un form per la registrazione.
   3. RF 17: Il sistema deve fornire la possibilità di effettuare il login tramite la compilazione di un form dedicato.
   4. RF 18: Il sistema deve fornire la possibilità di effettuare il log out dal proprio profilo tramite un bottone dedicato, che farà scadere la sessione attuale.
   5. RF 19: Il sistema deve fornire la possibilità di visualizzare le categorie di tutti i prodotti disponibili sul sito.
   6. RF 20: Il sistema deve fornire la possibilità di visualizzare i prodotti nelle specifiche pagine di ogni categoria.
   7. RF 21: Il sistema deve fornire la possibilità di ricercare un prodotto in modo veloce, attraverso una barra di ricerca.
   8. RF 22: Il sistema deve fornire la possibilità di inserire prodotti nel proprio carrello.
   9. RF 23: Il sistema deve fornire la possibilità di eliminare prodotti dal proprio carrello.
   10. RF 24: Il sistema deve fornire la possibilità di visualizzare i prodotti nel carrello col rispettivo totale da pagare.
   11. RF 25: il sistema deve fornire la possibilità di compilare un form per la registrazione.
4. Admin:
   1. RF 25: Il sistema deve fornire la possibilità di connettersi alla piattaforma.
   2. RF 26: Il sistema deve fornire la possibilità di effettuare il login tramite la compilazione di un form dedicato.
   3. RF 27: Il sistema deve fornire la possibilità di effettuare il log out dal proprio profilo tramite un bottone dedicato, che farà scadere la sessione attuale.
   4. RF 28: Il sistema deve fornire la possibilità di visualizzare gli ordini effettuati dai clienti attraverso una tabella.
   5. RF 29: Il sistema deve fornire la possibilità di inserire un nuovo prodotto all’interno del catalogo, attraverso la compilazione di un form dedicato.
   6. RF 30: Il sistema deve fornire la possibilità di eliminare un nuovo prodotto all’interno del catalogo, attraverso la compilazione di un form dedicato.
   7. RF 31: Il sistema deve fornire la possibilità di gestire gli utenti all’interno della web app.
   8. RF 32: Il sistema deve fornire supporto agli utenti per qualsiasi evenienza.
5. TABELLA DELLE PRIORITA’

|  |  |
| --- | --- |
| RF1, RF1.1, RF1.1.1 | ALTA |
| RF2, RF2.1, RF2.1.1 | ALTA |
| RF3, RF3.1, RF3.1.1 | ALTA |
| RF4 | MEDIA |
| RF5 | MEDIA |
| RF6, RF6.1 | ALTA |
| RF7, RF7.1 | ALTA |
| RF8, RF8.1 | MEDIA |
| RF9, RF9.1 | MEDIA |
| RF10, RF10.1 | MEDIA |
| RF11 | ALTA |
| RF12 | ALTA |
| RF13 | ALTA |
| RF14 | ALTA |
| RF15 | MEDIA |
| RF16 | MEDIA |
| RF17 | ALTA |
| RF18 | ALTA |

1. REQUISITI NON FUNZIONALI
2. Usability:

* Il sistema è progetto per rendere il tuo utilizzo più facile possibile. È semplice, con una grafica invitate e intuitiva. Ogni funzione sarà interpreta al meglio con bottoni, immagini e scritte. Esso è responsive, ovvero si adatta ad ogni tipo di dispositivo (tablet, smartphone, pc) su qual si utilizzi.
* L’utente registrato e no, avrà da subito la possibilità di consultare il catalogo presente. Per la prima categoria non avrà problemi a finalizzare l’ordine, mentre per l’altro, ci sarà l’obbligo di effettuare l’accesso o registrarsi nel caso in cui non lo si fosse ancora.

1. Realibility:

* Il sistema ha la capacità di rispondere a qualsiasi tipo di input, corretto o meno che sia.
* Soddisfare le richieste dell’utente.
* Sarà garantita la persistenza dei dati anche in caso di guasto. Inoltre, sarà garantito un minor *mean time to failure* possibile.

1. Performance:
   * Il tempo di risposta del sistema sarà immediato anche con un numero elevato di accessi.
   * Almeno 80% del sistema dovrà superare i test in fase di verifica delle performance attraverso collaudi e test.
2. Supportability:
   * Estrema facilità con cui il sistema potrà mutare dopo il deployement.
   * Ci sarà la possibilità di subire cambiamenti o aggiornamenti, in quanto sviluppato tramite approccio Object-Oriented.
3. Security:
   * Tutti i dati personali saranno criptati per garantire la sicurezza degli utenti.
4. Legal:

* Il sistema renderà obbligatorio acconsentire al trattamento dei dati personali in fase di caricamento dei dati anagrafici.

1. CASI D’USO

**UC1 Creazione profilo**

* **Descrizione**

Utente accede per la prima volta alla piattaforma e crea il proprio profilo

* **Attori**

Visitatore web app

* **Entry condition**

L’utente non deve essere registrato

* **Flussi eventi**

1. L’utente clicca sul pulsante Registrati presente nell’header;
2. Viene reindirizzato ad una nuova pagina, dove è presente il form da compilare per la registrazione;
3. Utente compila i campi del form inserendo:
   1. Nome
   2. Cognome
   3. Data di nascita
   4. Recapito telefonico
   5. Indirizzo
   6. Città
   7. Username
   8. Password
4. Utente conferma il modulo, tramite bottone dedicato
5. Utente viene reindirizzato alla pagina di login.

* **Eccezioni/flussi alternativi**

Ai punti 3.c e 3.h, se l’utente inserisce dati non idonei 🡪 UC1.1

Se username già esistente 🡪 UC 1.2

Se password non conforme 🡪 UC1.3

Se l’utente conferma senza aver inserito tutti i dati (campo vuoto)🡪 UC 1.4

* **Exit condition**

Tutti i campi sono stati inseriti correttamente e l’utente viene registrato al sistema

**UC 1.1 dato non idoneo**

* **Descrizione**

Utente inserisce un dato non ideneo

* **Attori**

Visitatore/utente guest/ utente registrato

* **Entry condition**

L’utente ha inserito un dato che non è conforme agli standard decisi in un form

* **Flussi eventi**

**1.** Al campo non idoneo viene aggiunto un bordo rosso

2. Utente corregge l’errore seguendo le linee guida

3. Utente clicca sul pulsante registrati

4. Utente viene reindirizzato alla pagina di login.

* **Eccezioni/flussi alternativi**

Se campo non idoneo 🡪UC1.1

* **Exit condition**

Tutti i campi sono stati inseriti correttamente e l’utente viene registrato al sistema

**UC 1.2 username esistente**

* **Descrizione**

Utente inserisce username esistente

* **Attori**

Visitatore/utente guest/ utente registrato

* **Entry condition**

L’utente ha inserito una username già esistente durante la registrazione

* **Flussi eventi**

1. Il sistema analizza il DB e capisce la username scelta dall’utente è stata già utilizzata
2. Il bordo del campo diventa rosso per avvisare dell’errore
3. L’utente decide una nuova username
4. Utente clicca pulsate “registrati”
5. Utente viene reindirizzato alla pagina di login

* **Eccezioni/flussi alternativi**

Se campo non idoneo 🡪UC1.1

Se username già utilizzata 🡪 UC1.2

* **Exit condition**

Username corretta e l’utente viene registrato

**UC 1.3 Password non conforme**

* **Descrizione**

Utente inserisce una password non conforme

* **Attori**

Visitatore/utente guest/ utente registrato

* **Entry condition**

L’utente ha inserito una password non conforme

* **Flussi eventi**

1. Utente nuova password
2. Il sistema analizza la password ed attraverso il colore del bordo ci fa capire se è corretta o meno: rosso (non conforme), gialla (poco sicura), verde (molto sicura)
3. Utente sceglie password giusta
4. Utente clicca “registrati”
5. Utente viene reindirizzato alla pagina di login

* **Eccezioni/flussi alternativi**

Se campo non idoneo 🡪UC1.1

Se password non idonea 🡪UC1.3

* **Exit condition**

password conforme e l’utente viene registrato

**UC 1.4 Campo vuoto**

* **Descrizione**

Utente non compila tutti i campi

* **Attori**

Visitatore/utente guest/ utente registrato

* **Entry condition**

L’utente ha confermato i form senza compilare tutti i campi

* **Flussi eventi**

1. Ai campi vuoti viene inserito un bordo rosso
2. Il sistema restituisce un messaggio di errore
3. Utente riempie campi vuoti
4. Utente clicca il pulsante “registrati”
5. Utente viene reindirizzato alla pagina di login

* **Eccezioni/flussi alternativi**

Campo vuoto🡪UC1.4

* **Exit condition**

Password conforme e l’utente viene registrato

**CU2. Login utente**

* **Descrizione**

L’utente registrato vuole effettuare il login al sito web.

* **Attori**

Utente registrato

* **Entry condition**

L’utente è in qualsiasi pagina del sito web/form login.

* **Flusso di eventi**

1. L’utente clicca il tasto LOGIN dalla homepage, presenta nell’header
2. Viene visualizzato il form del login
3. L’utente inserisce le sue credenziali: username e password
4. L’utente clicca sul pulsante “login”
5. L’utente viene indirizzato nella Home page

* **Eccezioni/flussi alternativi**

Al punto 4 vengono inseriti dati non corretti 🡪UC2.1

* **Exit condition**

Viene effettuato il login al sito web e reindirizzati alla home page

**UC2.1. Dati login errati**

* **Descrizione**

L’utente inserisce dati non corretti nel form di login.

* **Attori**

Utente registrato.

* **Entry condition**

L’utente inserisce username o password errati nel form del login

* **Flusso di eventi**

1. L’utente non viene reindirizzato alla pagina Home
2. L’utente inserisce nuovamente userame e password;
3. L’email clicca il pulsante “login”;
4. L’utente viene indirizzato alla pagina Home.

* **Eccezioni/flussi alternativi**

Nessuna.

* **Exit condition**

Viene effettuato il login al sito web e si viene reindirizzati alla home page

**UC2.2 Logout utente**

* **Descrizione**

l’utente vuole effettuare il logout al sito web.

* **Attori**

Utente registrato.

* **Entry condition**

L’utente si trova in qualsiasi pagina del sito web.

* **Flusso di eventi**

1. L’utente clicca sul pulsante “esci” presente nell’header
2. Il sito web indirizza l’utente alla home page del sito web.

* **Eccezioni/flussi alternativi**

Nessuna.

* **Exit condition**

Viene effettuato il logout al sito web e si viene reindirizzati alla home page.