**ANALISI DEI REQUISITI – RUBRICA – GRUPPO 02**

**INDICE**

* [**INTRODUZIONE**](#Introduzione)
* [**REQUISITI FUNZIONALI**](#Requisiti_funzionali)
  + [**FUNZIONALITÀ INDIVIDUALI**](#Funzioni_individuali)
  + [**BUSINESS FLOW**](#Business_flow)
  + [**ESIGENZE DI DATI E INFORMAZIONI**](#Esigenze_di_dati_e_informazioni)
  + [**INTERFACCIA UTENTE**](#Interfaccia_utente)
  + [**INTERFACCIA CON SISTEMI ESTERNI**](#Interfacce_con_sistemi_esterni)
* [**REQUISITI NON FUNZIONALI**](#Requisiti_funzionali)
  + [**VINCOLI DI PROGETTAZIONE**](#Vincoli_di_progettazione)
* [**ATTORI**](#Attori)
* [**CASI D’USO**](#Casi_d_uso)
* [**DESCRIZIONE CASI D’USO**](#Descrizione_casi_d_uso)
* [**TABELLA DELLE PRIORITÀ**](#Tabella_delle_priorità)

**INTRODUZIONE**

Il programma implementa una rubrica per gestire i contatti telefonici e/o e-mail. Il programma consente agli utenti registrati di inserire, eliminare o modificare dei contatti all’interno della rubrica e permette ai nuovi utenti di registrarsi. A ogni contatto sono associate almeno le seguenti informazioni:

·        Un nome e un cognome; una delle due informazioni può essere vuota (ma non entrambe)

·        Da zero a tre numeri di telefono

·        Da zero a tre indirizzi e-mail

**REQUISITI FUNZIONALI**

* **Funzionalità individuali**

1. Il sistema deve permettere agli utenti non registrati di registrarsi inserendo una e-mail e una password.
2. Il sistema deve permettere agli utenti registrati di accedere tramite e-mail e password.
3. Il sistema deve permettere agli utenti registrati di inserire un nuovo contatto.
4. Il sistema notifica con un messaggio se l’utente tenta di aggiungere un contatto duplicato. Due contatti sono uguali se hanno lo stesso nome e cognome.
5. Il sistema deve permettere agli utenti registrati di eliminare un contatto selezionato.
6. Il sistema deve permettere agli utenti di modificare i campi di un contatto selezionato.
7. Il sistema deve permettere di ricercare un contatto tramite una stringa inserita in una barra di ricerca.
8. Il sistema consente di aggiungere tag personalizzabili ai contatti.
9. Il sistema consente di filtrare i contatti secondo i tag scelti.
10. Il sistema deve permettere di ordinare le righe della tabella secondo il nome o cognome e con un ordinamento che può essere crescente o decrescente.
11. Una volta selezionato un criterio di ordinamento, questo viene memorizzato e viene applicato in automatico alle successive sessioni.
12. Il sistema deve permettere agli utenti registrati di importare i contatti da un file o esportare i contatti in un file.

* **Business flow**

Al momento essendo il sistema semplice non ce ne sono

* **Esigenze di Dati e informazioni**

1. Dati del profilo utente: e-mail e password.
2. Dati dei contatti: Ogni contatto deve essere salvato con nome, cognome (può anche esser presente o solo il nome o solo il cognome ma i due campi non possono essere contemporaneamente vuoti) da zero a tre numeri di telefono, da zero a tre indirizzi e-mail e zero o più tag. Un tag è un ulteriore attributo il cui valore può essere definito dall’utente durante l’aggiunta o la modifica di un contatto.

* **Interfaccia utente**

1. Pagina di login dove inserire l’e-mail e la password per il login
2. Pagina di Registrazione dove inserire l’e-mail e la password e conferma password per la registrazione
3. La rubrica viene visualizzata in forma tabellare. Ogni colonna della tabella corrisponde a un campo del generico contatto. Ogni riga della tabella corrisponde a un singolo contatto. I campi relativi al numero di telefono e all’indirizzo e-mail possono ospitare da zero a tre valori.
4. La selezione di un contatto avviene cliccando la corrispondente riga con il tasto sinistro del mouse.
5. Nella pagina contenente la rubrica sarà presente una barra di ricerca, un bottone che permetterà di selezionare il tipo di ordinamento (se ordinare per nome o cognome) e un bottone per indicare se l’ordinamento va fatto in ordine crescente o decrescente.
6. Messaggi di errore: in caso di errore nelle informazioni inserite dall’utente l’interfaccia deve indicare quale informazione è errata ed eventuali vincoli.
7. Quando l’utente proverà a inserire un contatto già esistente comparirà una notifica.

* **Interfacce con sistemi esterni**

1. Driver JDBC per PostgreSQL
2. PgAdmin per la gestione del DataBase

**Requisiti non funzionali**

1.  Sicurezza: Il sistema deve essere protetto da utenti non autorizzati. Solo gli utenti registrati possono accedere e visualizzare la propria rubrica.

2.  Scalabilità: Il sistema deve poter crescere con l’aumento del numero di utenti e contatti senza compromettere le prestazioni

3.  Usabilità: Il programma deve disporre di un’interfaccia grafica che permetta agli utenti di accedere facilmente alle diverse funzioni.

4. Manutenibilità: Il sistema deve organizzare i dati in un database che facilita l’implementazione di, future ed eventuali, nuove funzioni.

**VINCOLI DI PROGETTAZIONE**

1. Il sistema deve essere implementato tramite un’applicazione Java compatibile con Maven.
2. La scadenza finale è prevista il 15/12/2024.

**ATTORI**

·        Utente registrato: può effettuare il login; una volta effettuato il log-in, visualizza la pagina principale della rubrica contenente i contatti (e le relative informazioni) associati al proprio account.

·        Utente: viene guidato dall’interfaccia grafica nella procedura di registrazione accessibile dal tasto di sign-in

·        Database: gestisce il servizio di autenticazione e di memorizzazione dei contatti.

·        Interfaccia grafica (GUI): permette all’utente di accedere alle funzionalità del programma

**CASI D’USO**

1.      Registrazione (extend login)

2.      Login

3.      Visualizza rubrica (include login)

4.      Aggiungere contatto (extend visualizza rubrica)

5.      Rimuovere contatto (extend visualizza rubrica)

6.      Modificare informazioni per un contatto già esistente (extend visualizza rubrica)

7.      Ricerca contatti (extend Visualizza rubrica)

8.      Importare contatti (extend Aggiunge contatto)

9.      Esportare contatti (extend visualizza rubrica)

10.   Filtrare contatti (extend Visualizza rubrica)

11.   Ordinare contatti (extend Visualizza rubrica)

12. Contatto duplicato (extend Aggiungere contatto, extend modificare informazione per un contatto già esistente)

DESCRIZIONE CASI D’USO

**REGISTRAZIONE**

Attori partecipanti: utente, database, interfaccia grafica

Precondizioni: l’utente apre l’applicazione

Post-condizioni: l’utente ha un account e diventa un utente registrato

Flusso di eventi:

1.      L’utente inserisce e-mail e password (e conferma password)

2.      Verifica della correttezza dei dati inseriti

3.      I dati vengono inseriti nel database

4.      Registrazione completata

Flusso di eventi alternativo:

   2a.   L'indirizzo e-mail è già presente nel database, l’utente è già registrato. L’utente   viene reindirizzato alla pagina di login

 2b.   La password inserita non rispetta le condizioni di sicurezza consigliate; si richiede l’inserimento di una password differente.

2c. Le password non corrispondono

**LOGIN**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente deve essere registrato

Post-condizioni: l’utente ha accesso alla homepage della rubrica contenente i contatti associati all’account.

Flusso di eventi:

1.      Inserire e-mail e password

2.      verifica delle credenziali (nel database)

Flusso di eventi alternativo:

   1a. E-mail non presente nel database → Registrazione

   2a.   Credenziali errate

**VISUALIZZA RUBRICA**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente ha eseguito il login

Post-condizioni:

Flusso di eventi:

1.      l’utente visualizza la pagina principale della rubrica

Flusso di eventi alternativo:

   1a.   Aggiungere contatto

   1b.   Rimuovere contatto

   1c.   Modificare contatto

    1d.  Ricerca di uno o più contatti

     1e.  Importare/esportare contatti

    1f.  Filtrare contatti

    1g.  Ordinare contatti

**AGGIUNGERE CONTATTO**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente visualizza la rubrica

Post-condizioni: l’utente ha aggiunto un contatto alla propria rubrica

Flusso di eventi:

1.      L’utente chiama la funzione attraverso l’apposito pulsante

2.      L’utente aggiunge le informazioni da associare al nuovo contatto

Flusso di eventi alternativo:

   2a.  Se il campo nome e cognome sono entrambi vuoti il contatto non può essere inserito in rubrica

   2b.   Contatto duplicato.

**RIMUOVERE CONTATTO**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente visualizza la rubrica ed è stato inserito almeno un contatto

Post-condizioni: il contatto selezionato è stato eliminato dalla rubrica

Flusso di eventi:

1.     L’utente seleziona il contatto da eliminare

2.  L’utente chiama la funzione attraverso l’apposito pulsante

Flusso di eventi alternativo:

    1a.  Non è stato selezionato nessun contatto. Il tasto per la funzione di eliminazione è oscurato.

**MODIFICARE INFORMAZIONI PER UN CONTATTO GIÀ ESISTENTE**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente visualizza la rubrica ed è stato inserito almeno un contatto

Post-condizioni: le informazioni del contatto selezionato sono modificate

Flusso di eventi:

1.  L’utente seleziona il contatto da modificare

2. L’utente chiama la funzione attraverso l’apposito pulsante

3.  L’utente modifica i dati associati al contatto

4. Salva i cambiamenti.

Flusso di eventi alternativo:

4a. Non è possibile salvare la modifica effettuata perché sono stati eliminati sia il nome che il cognome.

4b. Contatto duplicato.

**RICERCA CONTATTI**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente visualizza la rubrica

Post-condizioni: vengono mostrati i contatti che contengono la stringa inserita

Flusso di eventi:

1.      L’utente inserisce nell’apposita barra di ricerca la stringa da ricercare tra le informazioni dei contatti presenti in rubrica.

2.      L’interfaccia mostra i soli contatti le cui informazioni combaciano con la stringa inserita

Flusso di eventi alternativo:

2a.   Nessun contatto contiene tra le proprie informazioni la stringa inserita. L’interfaccia mostra il messaggio “Nessuna corrispondenza trovata!”.

**IMPORTARE CONTATTI**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente ha effettuato il login

Post-condizioni: i contatti da importare sono aggiunti alla rubrica

Flusso di eventi:

1.      L’utente richiama la funzione attraverso l’apposito pulsante

2.      L’utente seleziona il file contenente i contatti da importare

1. Aggiungi contatto

Flusso di eventi alternativo:

2a.   In caso di errore di importazione l’interfaccia mostra il messaggio “Errore durante l’importazione.”

**ESPORTARE CONTATTI**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente visualizza la rubrica

Post-condizioni: i contatti presenti in rubrica vengono esportati in un apposito file

Flusso di eventi:

1.      L’utente richiama la funzione attraverso l’apposito pulsante

2.      L’utente seleziona la directory in cui desidera salvare il file.

Flusso di eventi alternativo:

2a.   In caso di errore di esportazione l’interfaccia mostra il messaggio “Errore durante l’esportazione.”

**ORDINARE CONTATTI**

Attori partecipanti: utente registrato, database, interfaccia grafica

Precondizioni: l’utente visualizza la rubrica

Post-condizioni: i contatti della rubrica sono visualizzati secondo il criterio scelto

Flusso di eventi:

1.  L’utente richiama la funzione attraverso l’apposito pulsante

2.  L’utente seleziona il criterio di ordinamento da eseguire (nome, cognome)

3. L’utente sceglie tra ordine crescente o decrescente

4. I contatti vengono ordinati.

Flusso di eventi alternativo:

4a. I contatti che non presentano il campo di ordinamento scelto vengono posizionati in fondo alla tabella e ordinati in base al valore dell’altro campo presente.

**FILTRARE CONTATTI**

Attori partecipanti: utente registrato, database, interfaccia

Precondizioni: l’utente visualizza la rubrica

Post-condizioni: vengono mostrati i soli contatti che rispettano il/i tag selezionato/i

Flusso di eventi:

1.      L’utente richiama la funzione attraverso l’apposito pulsante

2.      L’utente seleziona i tag

**CONTATTO DUPLICATO**

Attori partecipanti: utente registrato, database, interfaccia

Precondizioni: l’utente prova ad aggiungere o modificare un contatto

Post-condizioni: la rubrica viene aggiornata

Flusso di eventi:

1. compare un messaggio di scelta

Flusso di eventi alternativo:

1a. il contatto esistente viene modificato

1b. viene creato un contatto nuovo

1c. l’utente annulla l’operazione

**TABELLA DELLE PRIORITA’**

| NUMERO DEL REQUISITO | BREVE DESCRIZIONE DEL REQUISITO | PRIORITÀ DEL REQUISITO | STATO DEL REQUISITO | RISCHIO TECNICO |
| --- | --- | --- | --- | --- |
| 1 | GUI | Priorità 1 | Prima release | Basso |
| 2 | Registrazione utente | Priorità 1 | Prima release | Medio |
| 3 | Login utente | Priorità 1 | Prima release | Basso |
| 4 | Aggiunta di nuovi contatti | Priorità 1 | Prima release | Basso |
| 5 | Eliminazione dei contatti | Priorità 1 | Prima release | Basso |
| 6 | Modifica dei contatti | Priorità 1 | Prima release | Basso |
| 7 | Ricerca dei contatti | Priorità 1 | Prima release | Basso |
| 8 | Filtraggio dei contatti tramite il tag | Priorità 2 | Release successiva | Basso |
| 9 | Funzione di importazione ed esportazione dei contatti | Priorità 2 | Release successiva | Medio |
| 10 | Database | Priorità 2 | Release successiva | Alto |
| 11 | Gestione dei contatti duplicati | Priorità 1 | Prima release | Basso |