



Università degli Studi di Salerno

Dipartimento di Ingegneria dell'Informazione ed Elettrica e
Matematica Applicata (DIEM)

Documentazione del Progetto

Sviluppo di un applicazione Java per la gestione di una
rubrica

Corso di Ingegneria del Software - A.A. 2024/25

Studenti Gruppo 12:

Francesco Peluso
Matricola: 0612707469

Gerardo Selce
Matricola: 0612707692

Valerio Volzone
Matricola: 0612707693

Sharon Schiavano
Matricola: 0612708676

Docente:

Prof. Nicola Capuano

Last update: 4 dicembre 2024

Indice

| | | |
|----------|--|-----------|
| 1 | Elicitazione dei requisiti | 2 |
| 2 | Requisiti e vincoli di progetto | 3 |
| 2.1 | Requisiti Funzionali | 3 |
| 2.2 | Requisiti Non Funzionali | 3 |
| 3 | Categorizzazione di dettaglio | 4 |
| 3.1 | Funzionalità individuali | 4 |
| 3.2 | Business flow (Scenari) | 4 |
| 3.3 | Esigenze di dati e informazioni | 5 |
| 3.4 | Interfaccia utente | 5 |
| 3.5 | Altre interfacce con sistemi/piattaforme esterne | 6 |
| 3.6 | Ulteriori vincoli non funzionali | 6 |
| 3.7 | Schema di categorizzazione | 7 |
| 4 | Valutazione dei requisiti | 8 |
| 4.1 | Priorità dei requisiti | 8 |
| 4.2 | Rischio tecnico | 9 |
| 5 | Casi d'uso | 10 |
| 5.1 | Diagramma dei casi d'uso | 10 |
| 5.2 | Descrizione dei casi d'uso | 11 |
| 6 | Design architetturale | 13 |
| 6.1 | Attributi di qualità esterni | 13 |
| 6.2 | Attributi di qualità interni | 13 |
| 6.3 | Diagramma dei package | 13 |
| 7 | Design funzionale | 15 |
| 7.1 | Diagramma delle classi | 15 |
| 7.2 | Diagramma di sequenza | 15 |

Abstract

Il software da realizzare é una rubrica telefonica, con una interfaccia utente intuitiva e facilmente accessibile, che permette l'inserimento, la modifica e la cancellazione dei contatti. Le funzionalità richieste includono la visualizzazione in ordine alfabetico della lista dei contatti, la ricerca di un contatto attraverso parte del suo nome e/o cognome.

1 Elicitazione dei requisiti

- Importare una rubrica già esistente.
- Esportare una rubrica
- Creare un contatto.
- Salvare un contatto.
- Modificare un contatto.
- Eliminare un contatto.
- Ricercare un contatti.
- Visualizzare dei contatti.
- Interfaccia utente (GUI) semplice e accessibile.
- Ogni contatto deve avere almeno il nome o il cognome.
- Associare da 0 a 3 numeri di telefono per contatto.
- Inserimento di una categoria per ognuno dei numeri di telefono salvati.
- Associare da 0 a 3 indirizzi email per contatto.
- Inserimento di una categoria per ognuna delle email salvate.
- Creare una lista di contatti preferiti.
- Il sistema software dovrà poter gestire un massimo di 5000 contatti.
- Il sistema software dovrà mantenere un tempo di risposta inferiore a 2 secondi in operazioni di ricerca e salvataggio.
- Il file contenente i contatti, e le informazioni associate, sarà criptografato.

2 Requisiti e vincoli di progetto

Una prima categorizzazione viene effettuata tra requisiti funzionali e non funzionali.

2.1 Requisiti Funzionali

1. Consentire la creazione di un contatto con i seguenti campi:
 - Obbligatorie: nome o cognome.
 - Opzionali: fino a tre numeri di telefono (ognuno con una categoria opzionale) e fino a tre indirizzi email (ognuno con una categoria opzionale).
2. Salvare i contatti nel sistema.
3. Modificare un contatto esistente.
4. Eliminare un contatto.
5. Ricercare un contatto.
6. Visualizzare un elenco di contatti con opzioni di ordinamento.
7. Creare una lista di contatti preferiti.
8. Importare una rubrica già esistente.
9. Esportare una rubrica.

2.2 Requisiti Non Funzionali

1. Il sistema deve supportare un massimo di 5000 contatti, mantenendo un tempo di risposta inferiore a 2 secondi per operazioni di ricerca e salvataggio.
2. I dati devono essere criptografati utilizzando l'algoritmo AES-256.
3. L'interfaccia utente deve essere intuitiva e accessibile.
4. L'applicazione deve essere compatibile con sistemi operativi Windows, MacOS e Linux.

3 Categorizzazione di dettaglio

La categorizzazione di dettaglio può seguire le sei dimensioni dei requisiti, mostrati e descritti qui di seguito.

3.1 Funzionalità individuali

Vengono qui elencati i servizi individuali (ed essenziali) che il sistema dovrà offrire:

- **[IF-1.1] Gestione CRUD** dei contatti all'interno della rubrica.
 - **CRUD:** Create Read Update Delete;
- **[IF-1.2] Ricerca di un contatto** in base ad uno specifico attributo, quale può essere un numero di telefono, un indirizzo email, il nome con cui è stato salvato, una sottostringa, o altri attributi che il sistema memorizza per un contatto;
- **[IF-1.3] Ordinamento della vista dei contatti** in base a criteri come ordine alfabetico in base a nome, cognome o altri attributi;
- **[IF-1.4] Gestione dei contatti** tramite lista preferiti o associazione ad una categoria (*Es. Personale, Lavoro, Scuola, ecc...*);
- **[IF-2] Gestione di una rubrica** che l'utente ha creato/importato;
- **[IF-3.1] Salvataggio automatico della rubrica** su un file salvato in una directory di sistema (*Es. "AppData" di Windows*);
- **[IF-3.2] Import/Export** di un'intera rubrica o di un singolo contatto (quest'ultimo tramite standard vCard, utilizzato anche da Android, iOS, ...);

3.2 Business flow (Scenari)

Vengono qui elencati le interazioni tra uno o più utenti e il sistema, per la realizzazione di uno specifico processo:

- **[BF-1] Registrazione di un contatto**
 - L'utente vuole registrare un nuovo contatto all'interno della propria rubrica. Vengono richiesti dunque dal sistema, obbligatoriamente almeno il nome o il cognome del contatto. La registrazione non può avvenire se non vengono compilati i campi obbligatori, il resto è facoltativo. A registrazione avvenuta, l'utente potrà vedere che il nuovo contatto è stato registrato, poiché sarà visibile fin da subito nella lista di tutti i suoi contatti.
- **[BF-2] Modifica di un contatto**
 - L'utente vuole effettuare una modifica ad un contatto che è presente nella propria rubrica. Può effettuare qualunque modifica a qualunque campo presente, affinché vengano rispettati i requisiti precedentemente richiesti (ovvero la presenza obbligatoria di almeno il nome o il cognome).

- **[BF-3] Eliminazione di un contatto**

- L'utente vuole eliminare un contatto presente nella propria rubrica. Se questa azione verrà effettuata nella lista generale, allora il contatto verrà eliminato da qualsiasi lista categorizzata presente all'interno della rubrica. Se l'azione verrà invece effettuata all'interno di una lista specifica (es. preferiti), allora l'eliminazione avrà effetto solo su quella lista, e non su tutte quelle presenti.

- **[BF-4] Salvataggio/Lettura di una rubrica su/da file**

- Di default, il software effettuerà in automatico operazioni di salvataggio e lettura da un file che verrà salvato in una directory di sistema. Questo viene fatto per garantire che all'apertura del software, l'utente potrà subito interagire con gli elementi che ha inserito, e che alla chiusura, tutte le modifiche effettuate vengono salvate. Per prevenire la perdita di modifiche non ancora salvate (per cause esterne e non controllabili dal programma, come crash, blackout, ecc...) viene fatto anche un salvataggio periodico in background durante il runtime del programma.

3.3 Esigenze di dati e informazioni

Vengono qui elencati e descritti i dati che il sistema deve gestire:

- **[DF-1] Dati dei contatti**

- Dati anagrafici generali dell'utente (nome, cognome) e informazioni di contatto (indirizzo email, numero di telefono).

- **[DF-2] Sistema operativo**

- Famiglia di OS attualmente in uso dall'utente (Windows, Linux o macOS) per effettuare il salvataggio nell'apposita directory di sistema per le applications data.

- **[DF-3] Chiave di crittografia**

- Chiave usata per codificare/decodificare il file di salvataggio automatico della rubrica dell'utente.

3.4 Interfaccia utente

Vengono qui elencati le caratteristiche e gli elementi che si concentrano sulla progettazione e usabilità dell'interfaccia utente da realizzare:

- **[UI-1.1] Dashboard**

- Deve visualizzare tutti i contatti fino ad ora registrati, oltre ad una barra di menù per effettuare operazioni di massa sulla lista, delle tab per poter cambiare lista di visualizzazione (es. Generale, Preferiti, Lavoro, ecc...), il numero di contatti inseriti.

- **[UI-1.2] Scheda contatto**

- Riepilogo di tutti i dati inseriti per il contatto che è stato selezionato. Devono essere messi a disposizione anche dei bottoni per poter effettuare modifiche sul singolo contatto o per poterlo eliminare.

- **[UI-2] Design reattivo**

- Il software deve essere accessibile, intuitivo e di facile utilizzo. Per questa applicazione è previsto l'uso esclusivamente da desktop o PC tablet, dunque la finestra e i layout interni all'applicazione devono essere reattivi e potersi adattare in base alla dimensione della finestra.

- **[UI-3] Messaggi di errore**

- In caso di errori da parte dell'utente, l'applicazione deve indicare chiaramente dove e perchè si è verificato l'errore, e una possibile soluzione. Il tutto deve essere indicato in maniera chiara e trasparente, in modo da poter rendere utilizzabile l'applicazione anche agli utenti meno esperti.

3.5 Altre interfacce con sistemi/piattaforme esterne

Vengono qui elencati i modi in cui il sistema interagisce con altri sistemi esterni:

- *Il sistema attualmente non prevede l'utilizzo di interfacce con sistemi esterni.*

3.6 Ulteriori vincoli non funzionali

- **[FC-1] Prestazioni**

- Il sistema deve supportare un massimo di 5000 contatti, mantenendo un tempo di risposta inferiore a 2 secondi per qualsiasi operazione che viene fatta sulla lista di contatti;

- **[FC-2] Sicurezza**

- I dati che vengono salvati e letti dal file presente nella directory di sistema sono crittografati utilizzando l'algoritmo di cifratura AES-256. Opzionalmente, è possibile anche crittografare con una propria password l'export della rubrica;

- **[FC-3] Accessibilità**

- L'interfaccia utente (GUI) deve avere una UI e una UX abbastanza curata, dunque di facile utilizzo, ma soprattutto accessibile.

- **[FC-4] Compatibilità**

- L'applicazione deve essere compatibile con i sistemi operativi Windows, macOS e le varie distribuzioni Linux attualmente disponibili.

3.7 Schema di categorizzazione

| Area dei requisiti | Prefisso | Numerazione delle dichiarazioni dei requisiti |
|---------------------------|----------|--|
| Funzionalità individuali | IF | IF-1.1, IF-1.2, IF-1.3, IF-1.4, IF-2, IF-3.1, IF-3.2 |
| Business flow | BF | BF-1, BF-2, BF-3, BF-4 |
| Dati e formato dei dati | DF | DF-1, DF-2, DF-3 |
| Interfaccia utente | UI | UI-1.1, UI-1.2, UI-3, UI-4 |
| Interfaccia con i sistemi | IS | - |
| Ulteriori vincoli | FC | FC-1, FC-2, FC-3, FC-4 |

Tabella 1: La Tabella 1 mostra chiaramente ogni categoria ed i requisiti ad essa associata. E' esplicitato il prefisso con il quale riconoscere la categoria ed il numero del requisito, richiamando la descrizione fatta nei punti precedenti.



4 Valutazione dei requisiti

4.1 Priorità dei requisiti

Questo elenco aiuta a definire le priorità di ciascun requisito. Si è scelto di valutare i requisiti in base al criterio del business value - dunque ad ogni requisito verrà assegnato un livello di priorità tra i tre disponibili:

- **Alto (Must have)** - se il requisito non è soddisfatto, il sistema non è utilizzabile;
- **Medio (Should have)** - importante, ma non comporta un fallimento se omesso;
- **Basso (Nice to have)** - da soddisfare solo se non richiede grandi sforzi;

| Livello di priorità | Requisito |
|---------------------|--|
| Alto | <ul style="list-style-type: none">• Gestione CRUD dei contatti;• Ricerca di un contatto;• Ordinamento dei contatti;• Salvataggio automatico della rubrica;• Interfaccia utente; |
| Medio | <ul style="list-style-type: none">• Gestione della lista dei contatti preferiti;• Accessibilità dell'interfaccia;• Sicurezza tramite crittografia sui file di salvataggio;• Efficienza del sistema; |
| Basso | <ul style="list-style-type: none">• Messaggi di errore;• Import/Export di una rubrica da/su file; |

Tabella 2: La tabella mostra il livello di priorità associato ad ogni requisito, ottenuto valutando il grado di necessità di ognuno per il corretto funzionamento del sistema.

4.2 Rischio tecnico

Questo elenco aiuta a definire il rischio tecnico di ciascun requisito. Ad ogni requisito verrà assegnato un grado di rischio tra i tre disponibili:

- **Alto** - è difficile prevedere se si riuscirà a realizzare il requisito (in assoluto, o semplicemente nei tempi e nei costi previsti);
- **Medio** - si ritiene di poter realizzare il requisito, ma ci sono alcuni fattori di rischio che non possono essere controllati;
- **Basso** - si ha piena fiducia nella propria capacità di realizzarlo;

| Grado di rischio | Requisito |
|------------------|---|
| Alto | <ul style="list-style-type: none">• Accessibilità dell'interfaccia;• Sicurezza tramite crittografia sui file di salvataggio;• Efficienza del sistema; |
| Medio | <ul style="list-style-type: none">• Salvataggio automatico della rubrica;• Ordinamento dei contatti;• Gestione della lista dei contatti preferiti;• Import/Export di una rubrica da/su file; |
| Basso | <ul style="list-style-type: none">• Interfaccia utente;• Ricerca;• Gestione CRUD dei contatti;• Messaggi di errore; |

Tabella 3: La tabella mostra il grado di rischio associato ad ogni requisito, ottenuto valutando la difficoltà di implementazione di ognuno nel rispetto delle scadenze concordate.

5 Casi d'uso

5.1 Diagramma dei casi d'uso

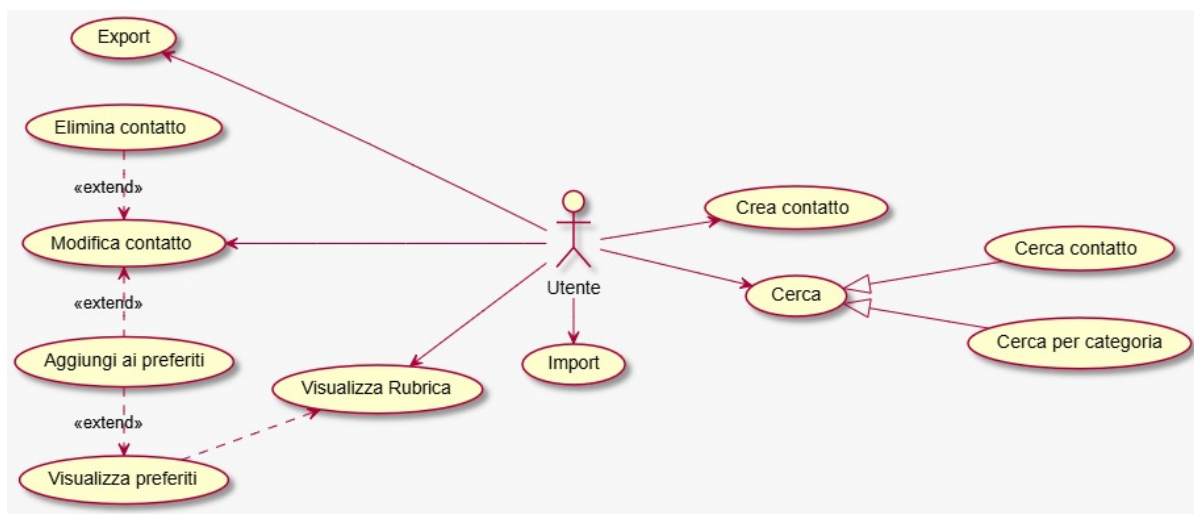


Figura 1: Diagramma dei casi d'uso



5.2 Descrizione dei casi d'uso

| | |
|-----------------------------|--|
| Nome: | <u>Creazione di un contatto</u> |
| Attore partecipante: | Utente |
| Precondizioni: | Nessuna |
| Postcondizioni: | Salvataggio del contatto sulla rubrica. |
| Flusso di eventi: | <ol style="list-style-type: none"><u>1.</u> L'utente visualizza una schermata con diversi campi<u>2.</u> Il cliente compila i campi obbligatori (nome o cognome) ed eventualmente i campi opzionali (da 0 a 3 numeri di telefono e da 0 a 3 email con la possibilità di categorizzare ognuno di essi).<u>3.</u> Il sistema salva il nuovo contatto. |
| Nome: | <u>Modifica/Eliminazione di un contatto</u> |
| Attore partecipante: | Utente |
| Precondizioni: | Esistenza in rubrica del contatto da modificare |
| Postcondizioni: | Modifica richiesta apportata al contatto in rubrica |
| Flusso di eventi: | <ol style="list-style-type: none"><u>1.</u> L'utente sceglie il contatto a cui apportare una modifica<u>2.</u> L'utente sceglie l'opzione "Modifica contatto"<u>3.</u> Si apre una schermata con le informazioni del contatto selezionato<u>4.</u> L'utente modifica le informazioni<u>5.</u> La modifica è apportata al contatto nell'elenco |
| Flusso alternativo: | <ol style="list-style-type: none"><u>a.</u> Eliminazione del contatto<u>2a.</u> L'utente sceglie l'opzione "Elimina contatto"<u>3a.</u> Il contatto viene eliminato dalla rubrica<u>b.</u> Aggiungi ai preferiti<u>2b.</u> L'utente sceglie l'opzione "Aggiungi ai preferiti"<u>3b.</u> Il contatto è aggiunto alla lista dei preferiti |
| Nome: | <u>Ricerca di un contatto</u> |
| Attore partecipante: | Utente |
| Precondizioni: | Nessuna |
| Postcondizioni: | Risultati coerenti con la ricerca effettuata |
| Flusso di eventi: | <ol style="list-style-type: none"><u>1.</u> L'utente seleziona la ricerca nell'elenco<u>2.</u> L'utente ricerca per nome o per cognome<u>3.</u> Il sistema reagisce dinamicamente durante l'inserimento dei dati, mostrando i risultati coerenti de presenti |
| Flusso alternativo: | <ol style="list-style-type: none"><u>a.</u> Ricerca di una categoria<u>2a.</u> L'utente seleziona la ricerca per Categoria<u>3a.</u> Il sistema mostra le Categorie presenti<u>4a.</u> L'utente seleziona la Categoria nella lista<u>5a.</u> Il sistema mostra gli elementi che rispondono alla ricerca.<u>b.</u> Nessun risultato coerente<u>3b.</u> Il sistema mostra una schermata vuota nel momento in cui i criteri non corrispondono a nessun contatto, cambiare i dati in qualsiasi momento aggiorna la ricerca dinamicamente |

| | |
|-----------------------------|---|
| Nome: | <u>Visualizzazione contatti</u> |
| Attore partecipante: | Utente |
| Precondizioni: | Nessuna |
| Postcondizioni: | Elenco visualizzato su schermo |
| Flusso di eventi: | <ol style="list-style-type: none"> 1. L'utente esegue l'applicativo 2. Il sistema mostra l'elenco dei contatti in ordine alfabetico per cognome |
| Flusso alternativo: | <ol style="list-style-type: none"> a. Visualizzazione lista dei preferiti 3a. L'utente seleziona "lista dei preferiti" 4a. Il sistema mostra l'elenco dei contatti preferiti ab. Ampliamento della lista dei preferiti 5ab. L'utente seleziona l'opzione di aggiunta dei contatti alla lista preferiti 6ab. Il sistema mostra l'elenco 7ab. L'utente seleziona nell'elenco i contatti da aggiungere alla lista dei preferiti 8ab. I contatti selezionati sono aggiunti alla lista dei preferiti |
| Nome: | <u>Salvataggio dei contatti in un file</u> |
| Attore partecipante: | Utente |
| Precondizioni: | Esistenza del file da cui prendere i contatti |
| Postcondizioni: | Prodotto un file contenente i contatti della rubrica |
| Flusso di eventi: | <ol style="list-style-type: none"> 1a. Il cliente seleziona l'opzione "Esporta" 2a. Il sistema salva il contenuto della rubrica in un file |
| Nome: | <u>Importazione dei contatti da un file</u> |
| Attore partecipante: | Utente |
| Precondizioni: | Esistenza del file da cui prendere i contatti |
| Postcondizioni: | I contatti contenuti nel file sono presenti in rubrica |
| Flusso di eventi: | <ol style="list-style-type: none"> 1. Il cliente seleziona l'opzione "Importa" 2. Il sistema inserisce i contatti presenti nel file all'interno della rubrica |
| Flusso alternativo: | <ol style="list-style-type: none"> a. Formato del file non coerente 3a. Il sistema mostra una schermata di errore, torna al punto 1 |

6 Design architetturale

Gli attributi di qualità che il nostro sistema, a parere del team, dovrebbe rispettare, sono i seguenti:

6.1 Attributi di qualità esterni

- **Disponibilità:** Il sistema deve essere sempre pronto all'uso.
- **Efficienza:** Il sistema deve rispettare i vincoli di risposta.
- **Security:** Il sistema deve proteggere i dati dell'utente.
- **Usabilità:** Il sistema deve fornire un'interfaccia chiara e un alto livello di user experience.
- **Installabilità:** Il sistema deve essere di facile installazione.
- **Adattabilità:** Il sistema deve fornire un'interfaccia grafica adattabile alle capacità grafiche del dispositivo utilizzato dall'utente.

6.2 Attributi di qualità interni

- **Manutenibilità:** Il sistema deve essere progettato con un alto livello di coesione e un basso livello di accoppiamento per facilitare la manutenibilità.
- **Modularità:** Il sistema deve distribuire le responsabilità su più moduli funzionali.
- **Riusabilità:** Il sistema deve essere composto da moduli funzionali riutilizzabili in altri progetti.
- **Portabilità:** Il sistema deve essere progettato per poter essere eseguito su dispositivi eterogenei.
- **Testabilità:** Il sistema deve essere composto da moduli funzionali facilmente testabili.

6.3 Diagramma dei package

La progettazione del diagramma dei package di un'applicazione per una rubrica telefonica in Java utilizzando Maven, JavaFX e SceneBuilder dovrebbe seguire i principi di modularità e separazione delle responsabilità.

7 Design funzionale

7.1 Diagramma delle classi

7.2 Diagramma di sequenza

