

Ingegneria del Software

DIEM - Università di Salerno

Progetto gruppo 07:

- Emanuele Tocci
- Alessio Leo
- Claudia Montefusco
- Rossella Pale



Introduzione

Una semplice rubrica telefonica realizzata in Java, che consente la gestione di contatti dotati di numeri telefonici ed indirizzi email, attraverso un'interfaccia grafica realizzata in JavaFX.

- Requisiti e casi d'uso
- Diagrammi dei package e delle classi
- Refactoring
- Demo





Requisiti e casi d'uso

Requirements Engineering



1. Creare un contatto
2. Modificare un contatto
3. Eliminare un contatto
4. Visualizzare un contatto
5. Cercare un contatto
6. Ordinare alfabeticamente i contatti
7. Catalogare i contatti in "preferiti" ed "emergenza"
8. Ogni contatto deve avere almeno nome o cognome
9. Ogni contatto deve avere da 0 a 3 numeri di telefono
10. Ogni contatto deve avere da 0 a 3 indirizzi email
11. Ogni contatto può avere informazioni aggiuntive opzionali (eg. compleanno, indirizzo di residenza, social links, note, foto profilo...)
12. Interfaccia grafica (GUI) minimale
13. Importare rubrica esistente (vCard)
14. Esportare rubrica esistente (vCard)
15. Applicazione cross-platform: Windows, macOS, Linux

Categorizzazione dei requisiti

Requisiti funzionali:

1. Gestione contatti: creazione, lettura, modifica ed eliminazione
2. Ricercare contatto
3. Ordinare alfabeticamente i contatti
4. Catalogare i contatti in "preferiti" ed "emergenza"
5. Ogni contatto può avere informazioni aggiuntive opzionali
6. Interfaccia grafica (GUI)
7. Import/Export

Requisiti non funzionali (vincoli)

1. Ogni contatto deve avere da 0 a 3 numeri di telefono
2. Ogni contatto deve avere da 0 a 3 indirizzi email
3. Ogni contatto deve avere almeno nome o cognome
4. Standard vCard per Import/Export
5. Da consegnare entro il 15 dicembre
6. Applicazione cross-platform: Windows, macOS, Linux

Categorizzazione di dettaglio

Funzionalità individuali - servizi individuali che deve offrire il sistema

- **IF1.1: Catalogazione dei contatti in “preferiti” e/o “emergenza”:**
 - Il sistema offre la possibilità di marcare i contatti come “preferiti” e/o “di emergenza”.
- **IF2.1: Ricerca contatti:**
 - Il sistema deve supportare la ricerca di un contatto tramite la sottostringa (parte) del nome o cognome.
 - Il sistema deve evidenziare i risultati nella listaUI e mostrare il relativo contatto all'utente
- **IF3.1: Ordinamento alfabetico:**
 - Il sistema offre la possibilità di ordinare i contatti per nome o cognome (opzione toggle).

Categorizzazione di dettaglio

Business flow - interazioni tra uno o più utenti e il sistema per la realizzazione di uno specifico processo aziendale

•BF1.1 : Creazione contatto

- Il sistema consente la creazione di un nuovo contatto e verifica automaticamente che i requisiti necessari alla creazione vengano rispettati

•BF1.2: Modifica contatto

- Il sistema consente di modificare un contatto già presente in rubrica;

•BF 1.3: Eliminazione contatto

- Il sistema consente di eliminare un contatto già presente in rubrica;

•BF 1.4: Visualizzazione/Lettura contatto

- Il sistema consente di visualizzare i contatti presenti in rubrica

•BF2.1: Import/Export (vCard)

- Il sistema offre all'utente la possibilità di esportare manualmente la rubrica in formato .vcf
- Il sistema offre all'utente la possibilità di importare manualmente una rubrica esistente in formato .vcf

Categorizzazione di dettaglio

Esigenze di dati e informazioni - dati che il sistema deve gestire

- **DF1.1: Informazioni essenziali sul contatto:**
 - Nome, cognome, numeri telefonici, email
 - Il sistema deve controllare automaticamente i campi (es. email valida) e restituire un errore nel caso il campo non sia valido
- **DF1.2: Informazioni opzionali sul contatto:**
 - Il sistema offre la possibilità di aggiungere dei dettagli aggiuntivi ai contatti come, per esempio: data di nascita, indirizzo fisico, foto profilo, link social, note, e relazioni (eventuali verifiche dei formati di dati)



Categorizzazione di dettaglio

Interfacce utente - progettazione e sull'usabilità delle interfacce utente del sistema

- **UI1.1: GUI (Interfaccia Grafica):**
 - Il sistema deve avere una GUI intuitiva e user-friendly.
 - La GUI deve consentire l'accesso rapido alle principali funzionalità del sistema (es. ricerca, aggiunta, modifica, eliminazione)



Categorizzazione di dettaglio

Ulteriori vincoli

- **FC1.1: Numero min/max di numeri telefonici ed email registrabili per ogni contatto**
 - Ogni contatto può avere da 0 a 3 numeri di telefono e da 0 a 3 indirizzi email
- **FC1.2: Vincoli sulla creazione di un contatto**
 - Il sistema consente la creazione del contatto solamente se l'utente inserisce almeno nome o cognome (una delle due informazioni può essere vuota)
- **FC2.1: Standard vCard**
 - Standard vCard per Import/Export
- **FC3.1: Deadline**
 - Da consegnare entro il 15/12/24
- **FC4.1: Cross-Platform**
 - Applicazione cross-platform compatibile con Windows, macOS e Linux

Casi d'uso

Nome: Creazione di un contatto

Attore: Utente.

Pre-condizioni: l'utente è in possesso di Nome, Cognome, indirizzo e-mail e numero telefonico valido.

Post-condizioni: il contatto viene creato e salvato in rubrica.

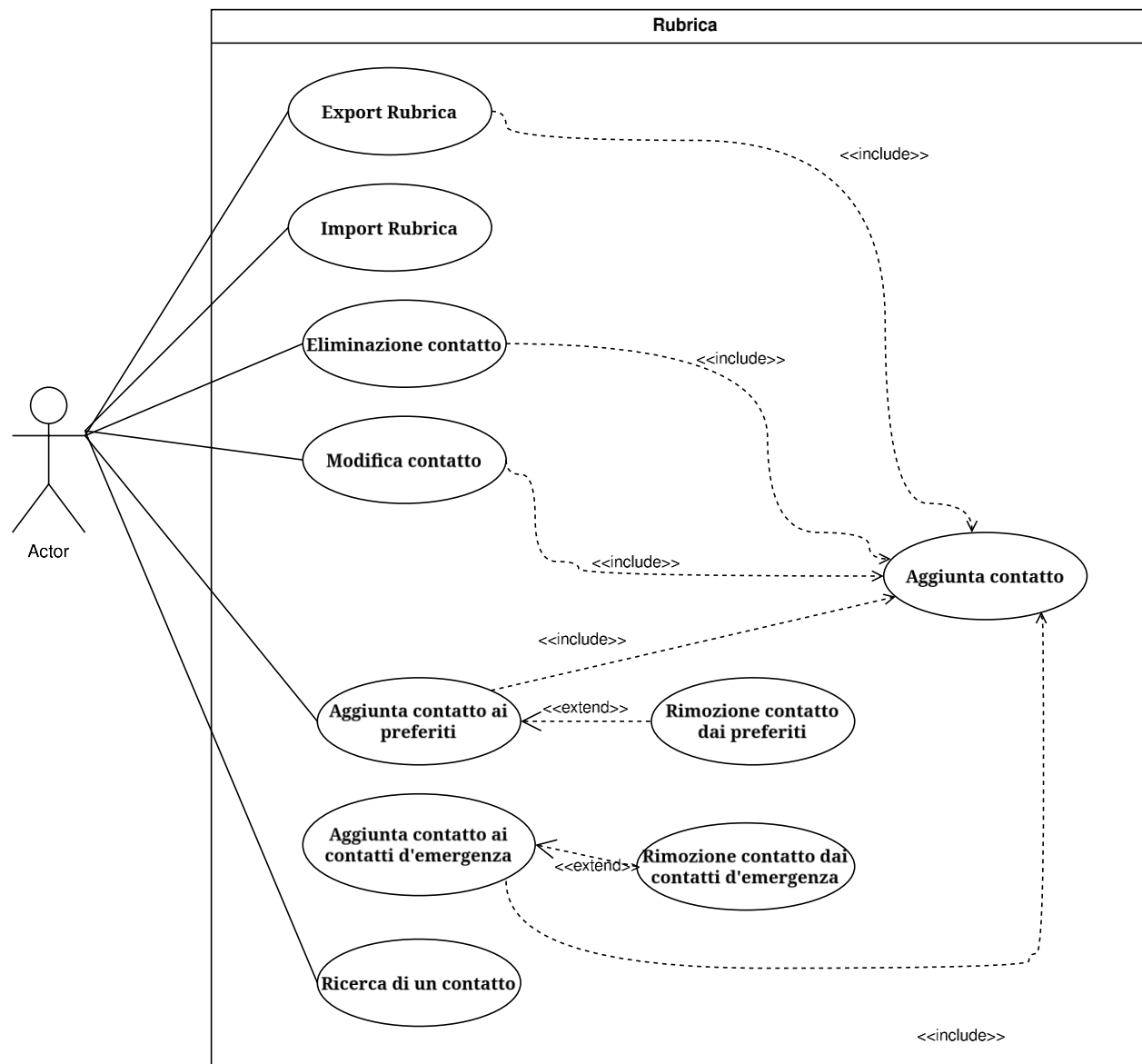
Flusso di eventi normale:

- L'utente inserisce i dati negli appositi campi di testo.
- Il sistema controlla la validità dei campi inseriti
- Il sistema salva e aggiunge il contatto in rubrica secondo l'ordine alfabetico.

Flussi alternativi:

- **a.1.** L'utente inserisce i dati negli appositi campi di testo.
- **a.2.** Il contatto già esistente, viene mostrato messaggio di errore.
- **b.1** L'utente non ha inserito nè nome nè cognome negli opportuni campi di testo.
- **b.2** il sistema richiede all'utente di inserire almeno uno tra le due informazioni.

UML





Diagrammi dei Package e delle Classi

Diagramma dei package

1.0 Diagramma dei package

Struttura organizzativa del codice in package differenti, al fine di separare al meglio funzionalità e responsabilità. La suddivisione segue il pattern MVC:

1.1 `gestoreIO`: si occupa della gestione dell'input/output e del formato vCard.

1.2 `models`: contiene le classi che rappresentano le entità principali del dominio applicativo.

1.3 `controllers`: package dedicato alla logica applicativa e all'interazione tra modelli e viste.

Diagramma dei package

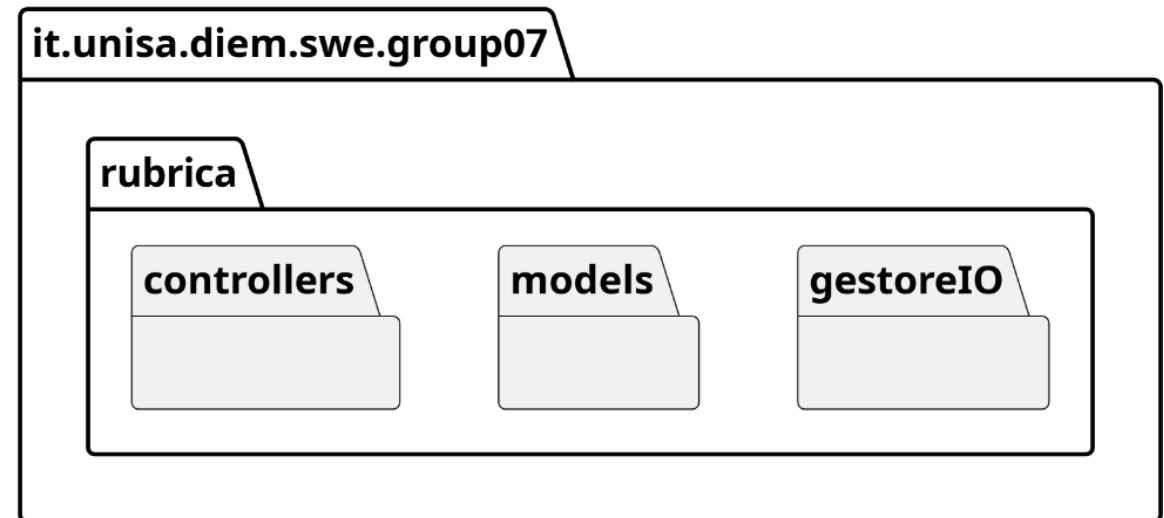


Diagramma delle classi

2.0 Diagramma delle classi

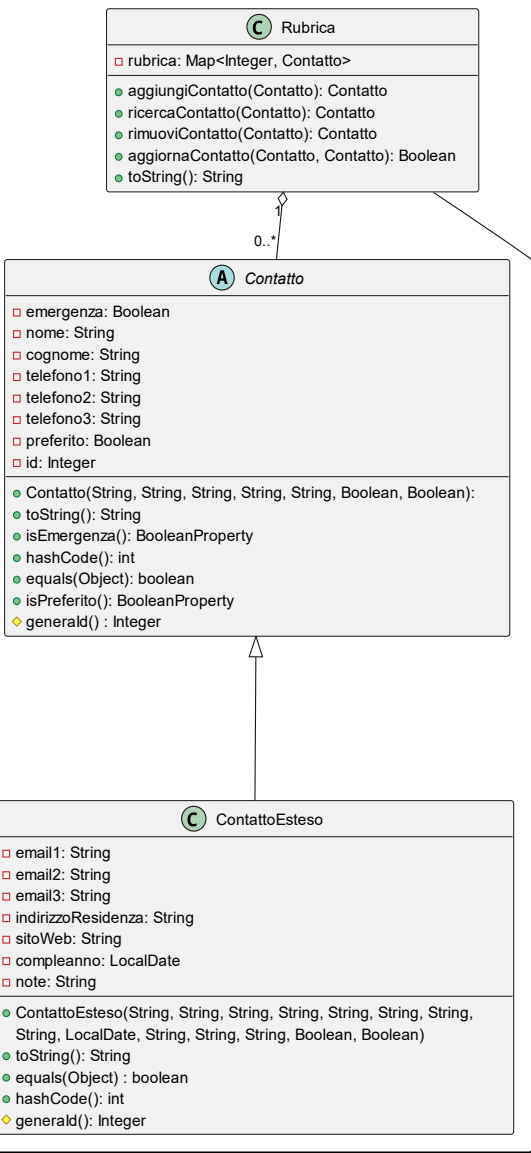
Per quanto riguarda la struttura delle classi, si è deciso di presentare due implementazioni differenti dei diagrammi delle classi:

2.1 Diagramma delle classi compatto

2.2 Diagramma delle classi esteso

La differenza fondamentale sta nel fatto che nel 1.1 non compaiono elementi FXML ed i metodi getter/setter; al contrario, l'1.2 presenta una versione più completa.

models



controllers



gestoreIO

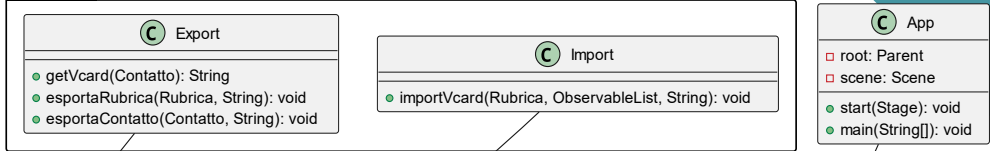


Diagramma delle classi: descrizione delle classi e delle relazioni

Classi:

- **Rubrica:** si occupa della gestione della rubrica.
- **Contatto:** rappresenta un contatto telefonico dotato di un set ristretto di attributi.
- **Contatto Esteso:** oggetto instanziabile aggiungibile all'interno della rubrica.
- **Generic Controller:** astrae il concetto di controller generico responsabile dell'interazione della vista con i modelli.
- **Rubrica Controller:** gestisce le interazioni dell'utente con l'interfaccia grafica principale.
- **Crea Contatto Controller:** gestisce l'interazione dell'utente con l'interfaccia grafica secondaria responsabile della creazione di un nuovo contatto.
- **Import:** si occupa della logica d'importazione da file .vcf.
- **Export:** si occupa della logica di esportazione su file .vcf.

Relazioni:

- Rubrica "1" o-- "0..*" Contatto (aggregazione)
- Contatto <|-- ContattoEsteso (ereditarietà)
- GenericController <|-- RubricaController (ereditarietà)
- GenericController <|-- CreaContattoController (ereditarietà)
- Rubrica – GenericController (associazione)
- Import – RubricaController (associazione)
- Export – RubricaController (associazione)



Refactoring

Durante la fase di sviluppo sono state apportate diverse modifiche rispetto al design iniziale. Le modifiche hanno riguardato principalmente l'interfaccia grafica dell'applicazione e il soddisfacimento di alcuni requisiti opzionali, che sono stati poi modificati.

Refactoring: ricerca tramite informazioni aggiuntive sul contatto

L'idea originale era di consentire la ricerca di un contatto mediante **qualunque informazione** associata ad esso come per esempio un numero telefonico, un indirizzo email o la data di nascita.

Durante la fase di design abbiamo tuttavia deciso di limitare la ricerca al **solo nome o cognome**, reputando la ricerca mediante gli altri attributi poco interessante per l'utente che usa la rubrica.

Abbiamo di conseguenza dovuto aggiornare i requisiti e i casi d'uso correlati.

The screenshot shows a desktop application window titled 'Rubrica Telefonica'. It has a menu bar with 'File', 'Modifica', 'Finestra', and '?'. Below the menu bar, there are two tabs: 'Contatti' and 'Nuovo Contatto'. The 'Contatti' tab is active, displaying a table with two columns: 'Nome' and 'Cognome'. The first row contains 'Emanuele' and 'Tocci'. To the right of the table, there is a 'Nuovo Contatto' button and a search bar containing 'emanuele'. The 'Dettagli' tab is also visible, showing a profile for 'Emanuele Tocci' with a profile picture and various fields for personal information: Name (Emanuele), Surname (Tocci), Phone 1 (+3933333333), Phone 2 (0890890890), Phone 3 (+39), Email 1 (emanueletozzi@gmail.com), Email 2 (emanueletozzi@cloud.com), Email 3 (emanueletozzi@unisa.it), Birthdate (dd/mm/aa), Website (https://github.com/emanueletr), Address (Via Giovanni Paolo II, 132, 84084), and a 'Note' field with a link to 'Più informazioni'. At the bottom of the window, it says 'Rubrica Gruppo 07'.

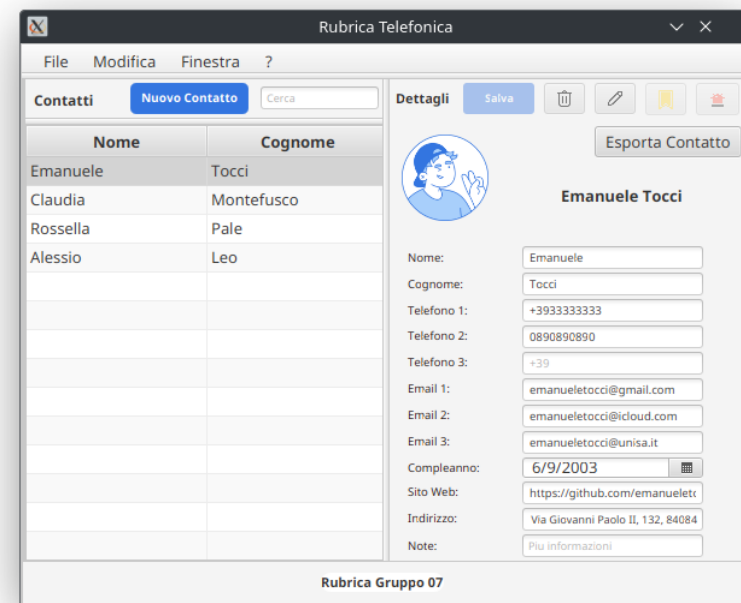
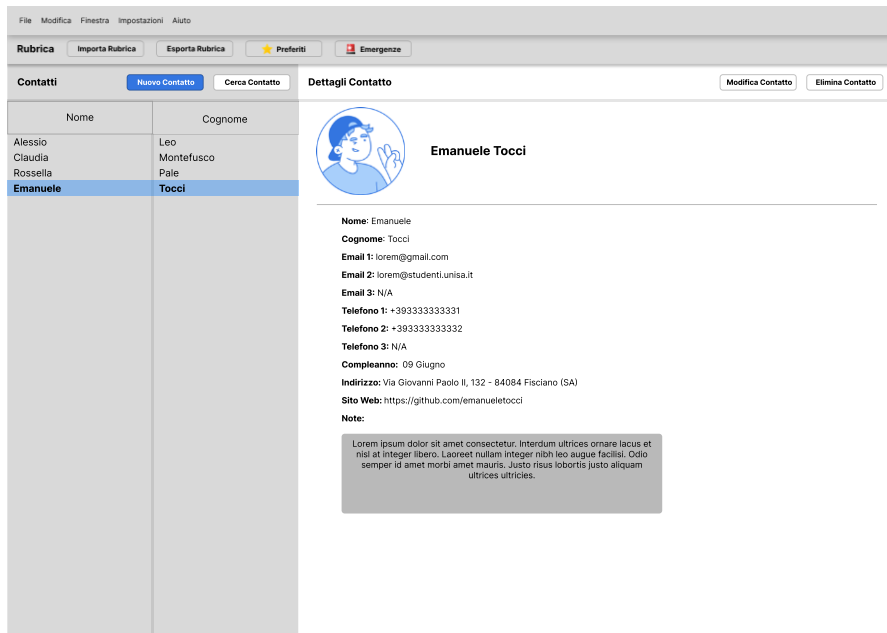
Refactoring: categorizzazione dei numeri telefonici ed indirizzi email

Tra i requisiti inizialmente definiti vi era la **categorizzazione** dei numeri telefonici e degli indirizzi email con delle etichette personalizzabili associate ad ogni numero/email. L'utente avrebbe avuto la possibilità di selezionare un'etichetta già presente nel sistema o crearne una nuova.

Abbiamo deciso di rimandare l'implementazione ad una eventuale release successiva e pertanto il relativo requisito (o caso d'uso) è stato rimosso dalla documentazione.

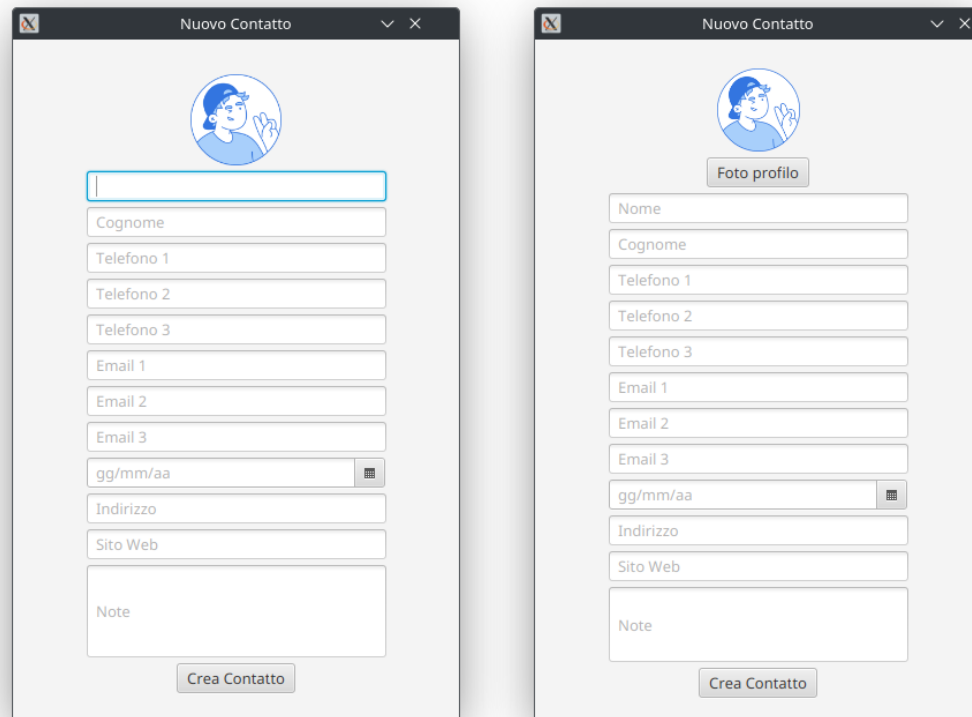
Refactoring: modifiche all'interfaccia grafica e caricamento avatar utente

L'interfaccia grafica è stata soggetta a varie rivisitazioni durante lo sviluppo dell'applicazione. Il mockup realizzato infatti presenta svariate differenze con la GUI effettivamente implementata in javaFX.



Refactoring: modifiche all'interfaccia grafica e caricamento avatar utente

Inizialmente avevamo intenzione di implementare un meccanismo che consentisse all'utente di sostituire l'avatar predefinito con un'immagine caricata.



The image displays two side-by-side screenshots of a web application window titled "Nuovo Contatto". Both windows contain a form for creating a new contact. The form includes a profile picture placeholder at the top, followed by input fields for "Cognome", "Telefono 1", "Telefono 2", "Telefono 3", "Email 1", "Email 2", "Email 3", a date field "gg/mm/aa", "Indirizzo", "Sito Web", and a "Note" text area. A "Crea Contatto" button is at the bottom. In the left screenshot, the profile picture is a default cartoon character, and there is a large empty input field below it. In the right screenshot, the profile picture has been replaced by a user-uploaded image, and the label "Foto profilo" is visible below it.

Per una serie di problemi con il corretto **ridimensionamento** delle immagini caricate dall'utente, abbiamo deciso di non implementare momentaneamente questa funzionalità.

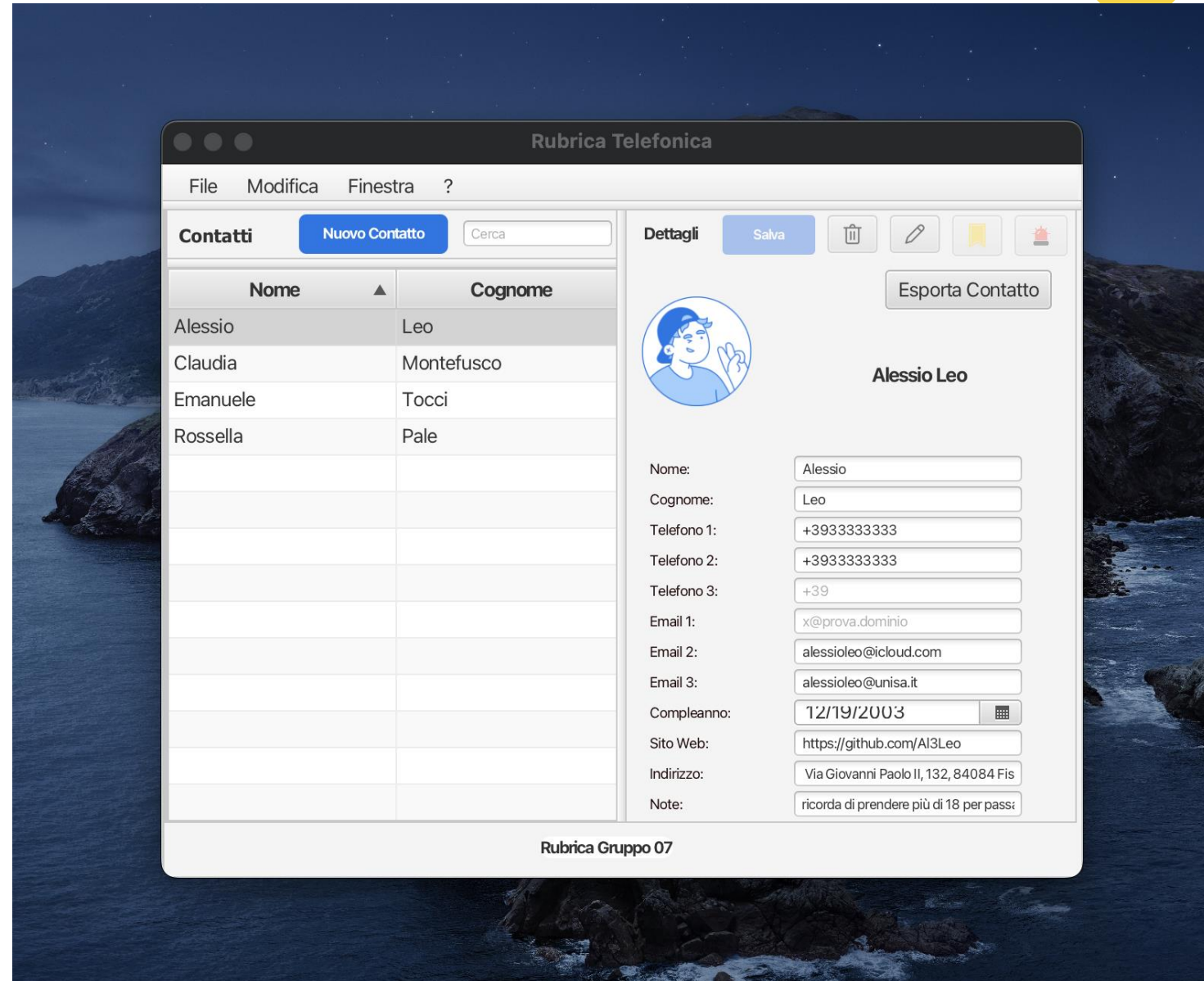


Demo

Demo

Gruppo07

- Home e top bar
- Import/export
- Ordinamento
- Ricerca di un contatto
- Visualizza un contatto
- Modifica di un contatto
- Preferiti/Emergenza
- Rimozione di un contatto



Demo

- **Home:** semplice ed user-friendly
- **Import:** accetta contatti in standard [rfc 6350](#) (Vcard) singoli e di intere rubriche
- **Export:** contatti singoli o del intera rubrica in formato vcard e quindi compatibile con la maggior parte di rubriche software in circolazione (Apple o Google contacts)
- **Ordinamento:** alfabetico per cognome o nome
- **Ricerca:** tramite nome o cognome e sottostringhe

Demo

- **Visualizzazione:** informazioni del contatto selezionato
- **Modifica di un contatto:** è possibile modificare tutte le informazioni di un contatto presente in rubrica
- **Preferiti/Emergenza:** è presente la possibilità di aggiungere dei contatti in delle sottorubriche "Preferiti" e "Contatti d'emergenza". Questa aggiunta migliora l'usabilità del sistema, consentendo una gestione più efficiente dei contatti in situazioni particolari, utile soprattutto per rubriche di grandi dimensioni.
- **Rimozione:** la rubrica permette la rimozione dei contatti