Gruppo 13

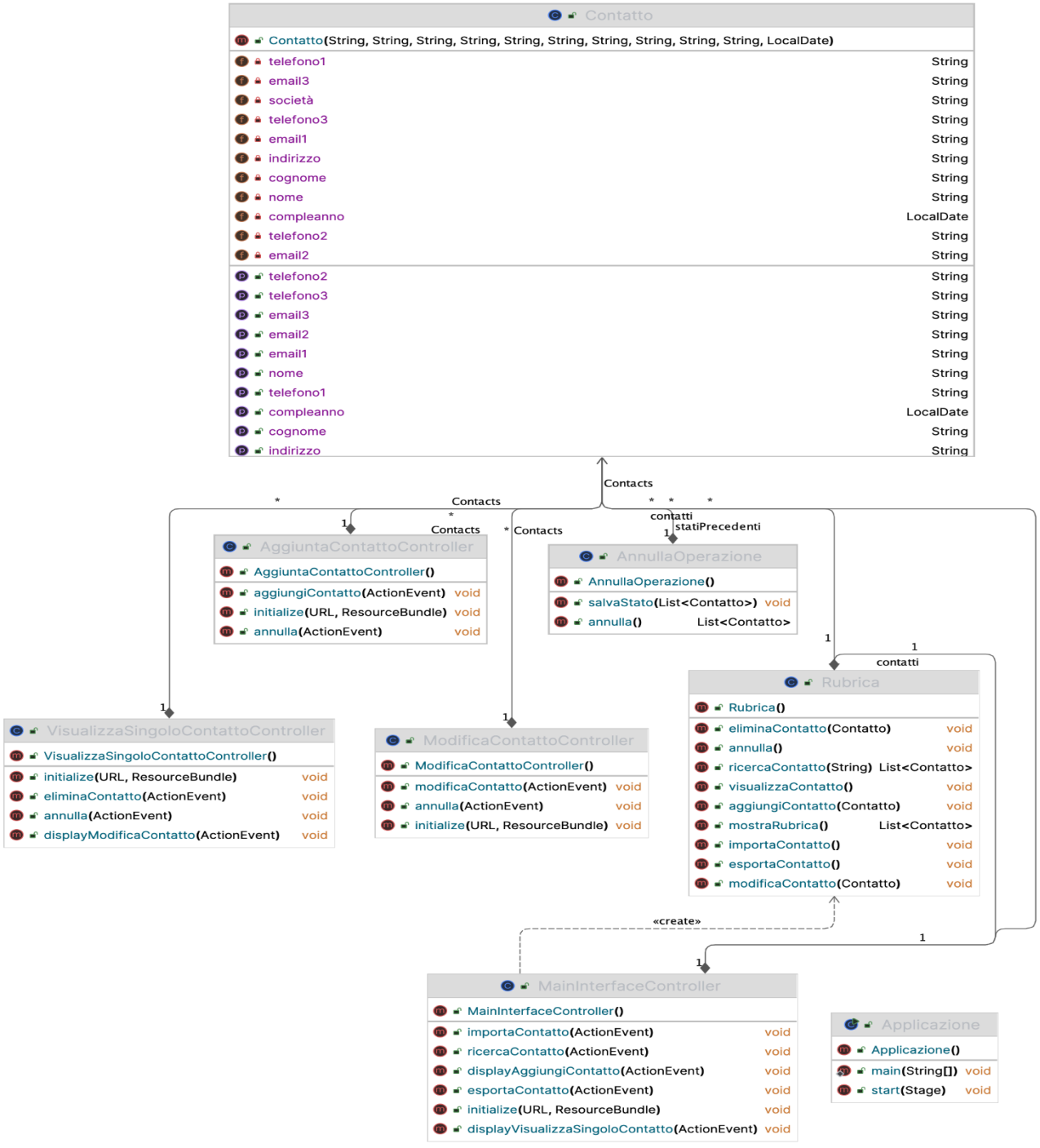
Catello Donnarumma **Luca Celentano** **Federico Cervo**
Francesco De Bonis

Project Design Report

2024/2025

Indice

1. Diagramma delle Classi



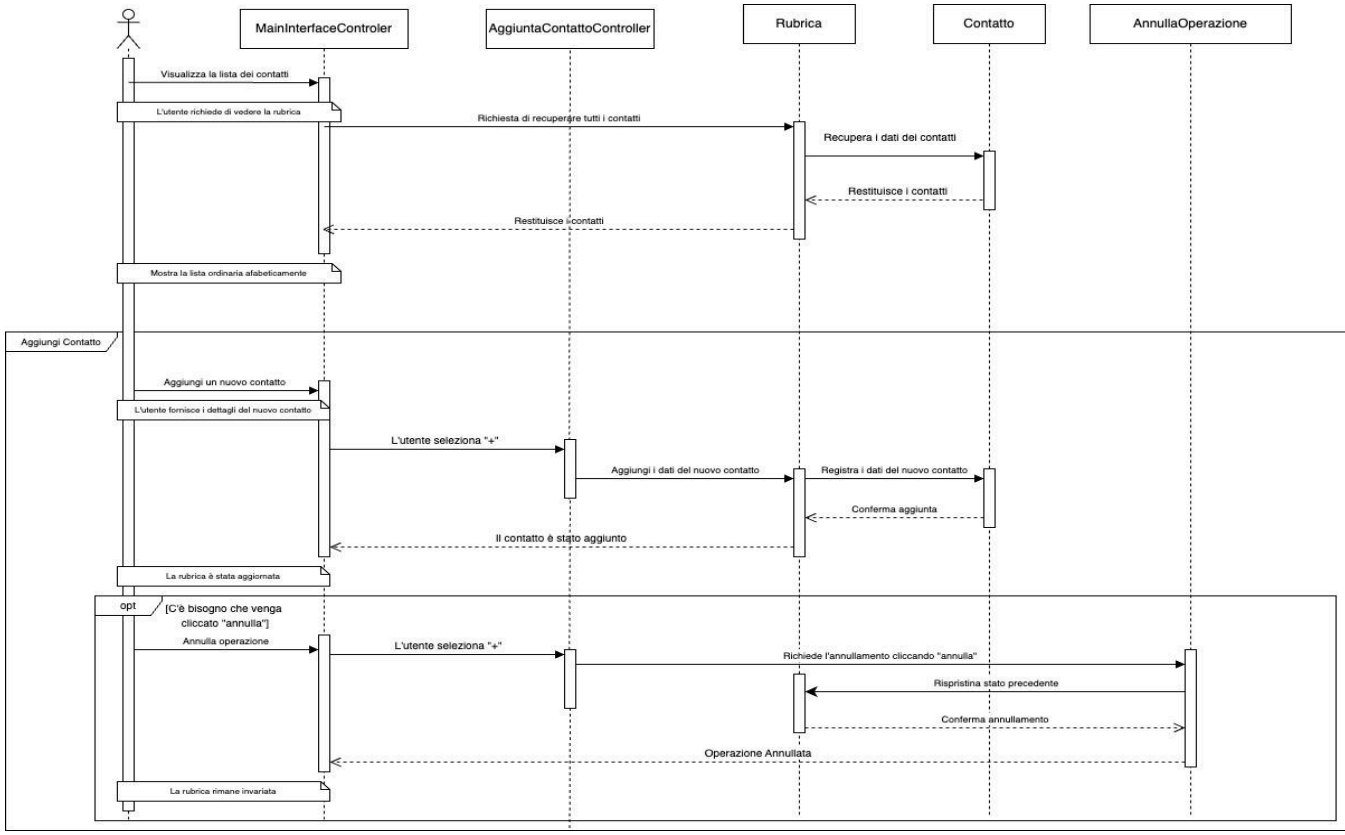
1.1 Commento al Diagramma Delle Classi

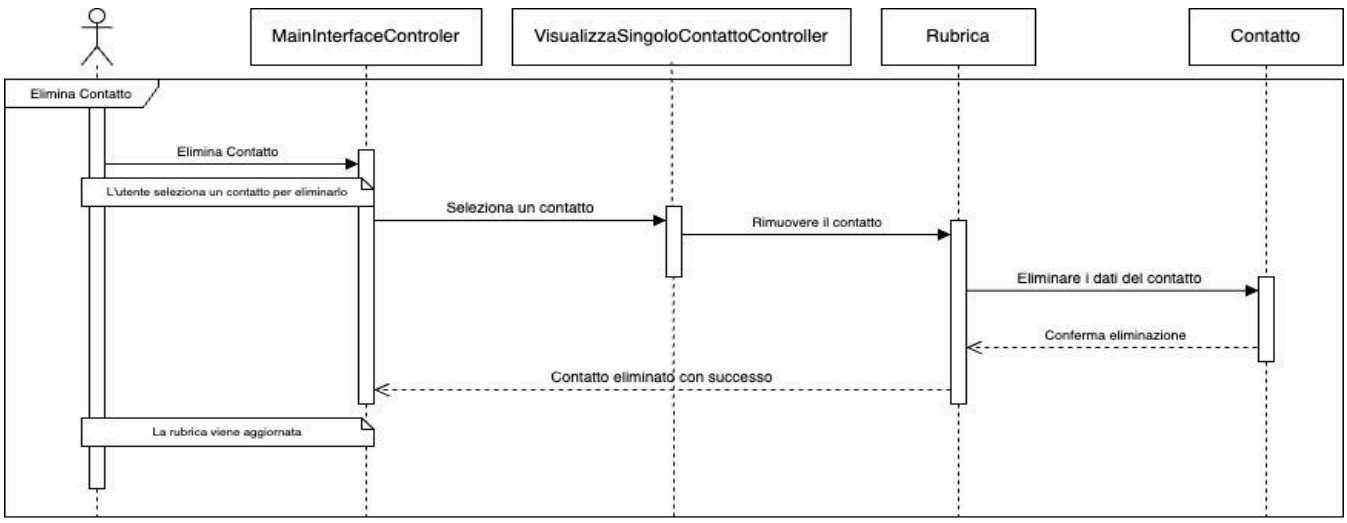
Il diagramma delle classi mostra un'implementazione che segue i principi SOLID. Ogni classe segue il principio della singola responsabilità (SRP): ad esempio, MainInterfaceController si occupa di gestire le operazioni dell'interfaccia principale, mentre Rubrica si occupa della gestione dei contatti.

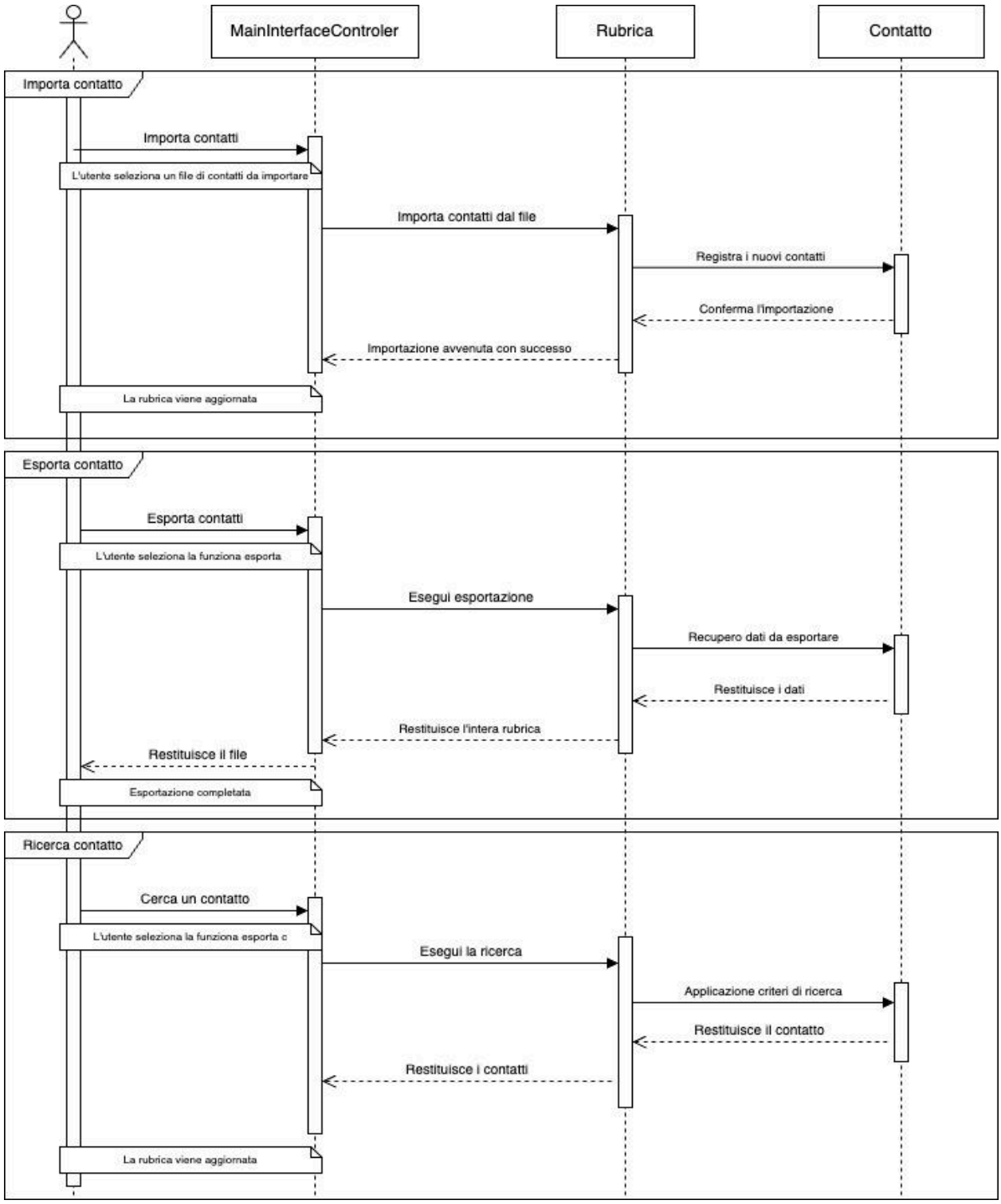
Le relazioni tra classi sono ben definite e seguono i principi SOLID:

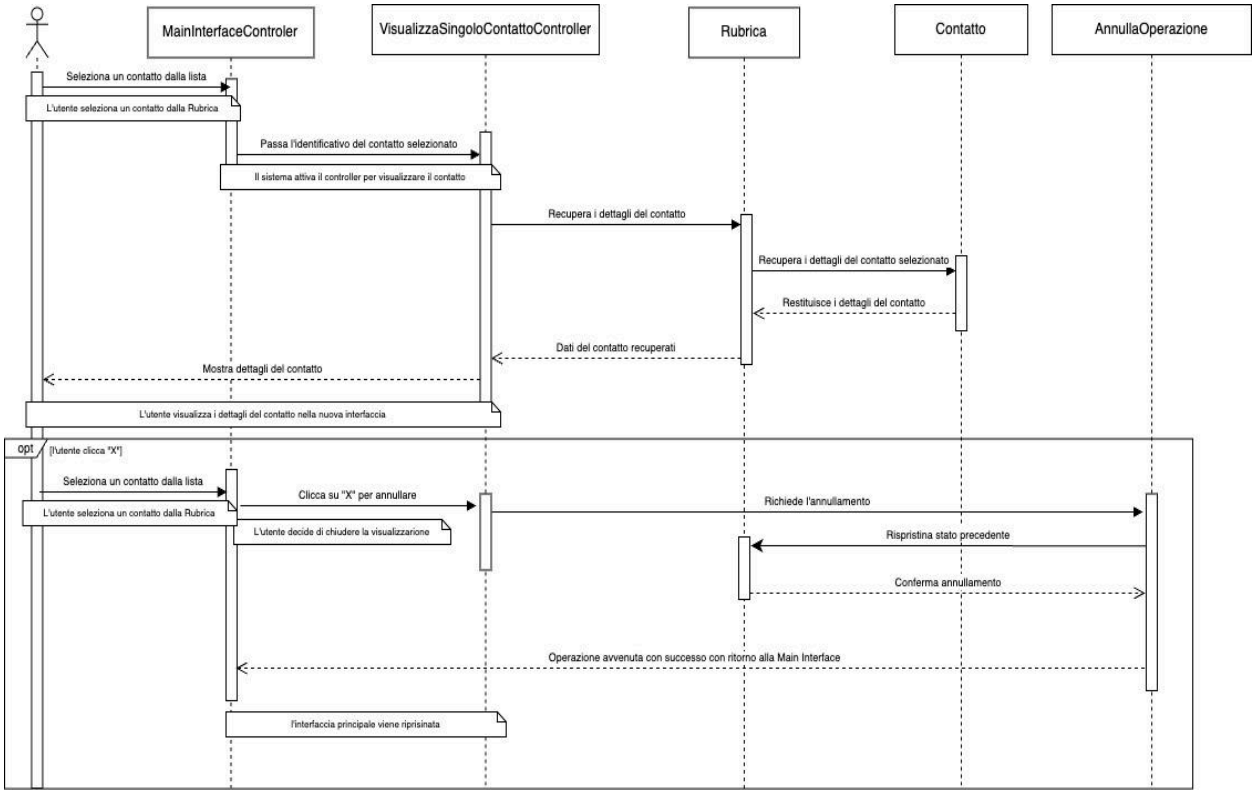
- SRP (Single Responsibility Principle): Ogni classe ha una responsabilità chiara, ad esempio MainInterfaceController gestisce l'interfaccia principale, mentre Rubrica si occupa della gestione dei contatti.
- OCP (Open-Closed Principle): Le classi sono progettate per essere estese senza necessità di modifiche dirette, favorendo la scalabilità.
- LSP (Liskov Substitution Principle): Le classi derivanti rispettano il contratto delle classi base, garantendo sostituibilità.
- ISP (Interface Segregation Principle): Le interfacce sono specifiche e ridotte, riducendo la dipendenza dai metodi non utilizzati.
- DIP (Dependency Inversion Principle): Le classi di alto livello dipendono da astrazioni, migliorando la modularità.

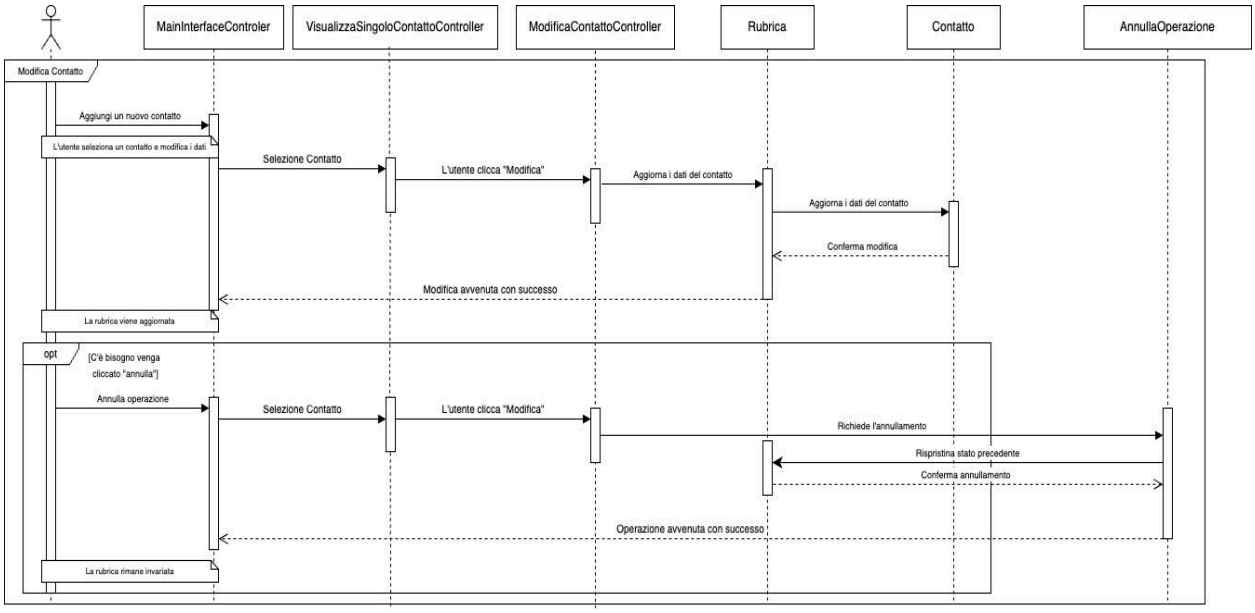
2. Diagrammi di Sequenza






MainInterfaceControler
Rubrica
Contatto
Importa contatto
Importa contatti
L'utente seleziona un file di contatti da importare
Importa contatti dal file
Registra i nuovi contatti
Conferma l'importazione
Importazione avvenuta con successo
La rubrica viene aggiornata
Esporta contatto
Esporta contatti
L'utente seleziona la funziona esporta
Esegui esportazione
Recupero dati da esportare
Restituisce i dati
Restituisce l'intera rubrica
Restituisce il file
Esportazione completata
Ricerca contatto
Cerca un contatto
L'utente seleziona la funziona esporta c
Esegui la ricerca
Applicazione criteri di ricerca
Restituisce il contatto
Restituisce i contatti
La rubrica viene aggiornata


MainInterfaceControler
VisualizzaSingoloContattoController
Rubrica
Contatto
AnnullaOperazione
Seleziona un contatto dalla lista
L'utente seleziona un contatto dalla Rubrica
Passa l'identificativo del contatto selezionato
Il sistema attiva il controller per visualizzare il contatto
Recupera i dettagli del contatto
Recupera i dettagli del contatto selezionato
Restituisce i dettagli del contatto
Dati del contatto recuperati
Mostra dettagli del contatto
L'utente visualizza i dettagli del contatto nella nuova interfaccia
opt
[l'utente clicca "X"]
Seleziona un contatto dalla lista
L'utente seleziona un contatto dalla Rubrica
Clicca su "X" per annullare
L'utente decide di chiudere la visualizzarione
Richiede l'annullamento
Rispristina stato precedente
Conferma annullamento
Operazione avvenuta con successo con ritorno alla Main Interface
l'interfaccia principale viene riprisinata


MainInterfaceControler
VisualizzaSingoloContattoController
ModificaContattoController
Rubrica
Contatto
AnnullaOperazione
Modifica Contatto
Aggiungi un nuovo contatto
L'utente seleziona un contatto e modifica i dati
Selezione Contatto
L'utente clicca "Modifica"
Aggiorna i dati del contatto
Aggiorna i dati del contatto
Conferma modifica
Modifica avvenuta con successo
La rubrica viene aggiornata
opt
[C'è bisogno venga cliccato "annulla"]
Annulla operazione
Selezione Contatto
L'utente clicca "Modifica"
Richiede l'annullamento
Rispristina stato precedente
Conferma annullamento
Operazione avvenuta con successo
La rubrica rimane invariata

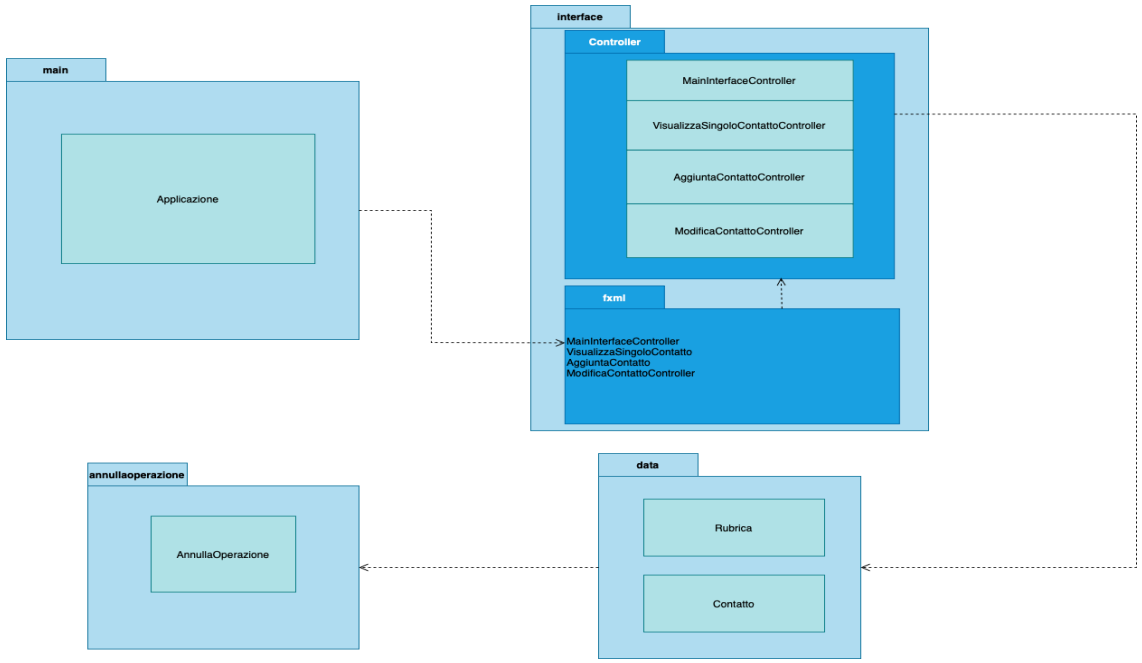
2.1 Commenti Diagrammi Di Sequenza

I diagrammi di sequenza descrivono in dettaglio i flussi principali: aggiunta, modifica, ricerca,eliminazione, importazione, esportazione e visualizzazione dei contatti.

I diagrammi di sequenza ci mostrano una chiara separazione dei ruoli tra le classi, seguendo i principi di:

- Ortogonalità: Ogni classe esegue solo le operazioni di sua competenza.
- Regola del Boy-Scout: Ogni interazione migliora la chiarezza e riduce il debito tecnico.
- KISS: Ogni flusso (aggiunta, modifica, eliminazione, importazione, esportazione,ricerca) è rappresentato con semplicità e linearità.

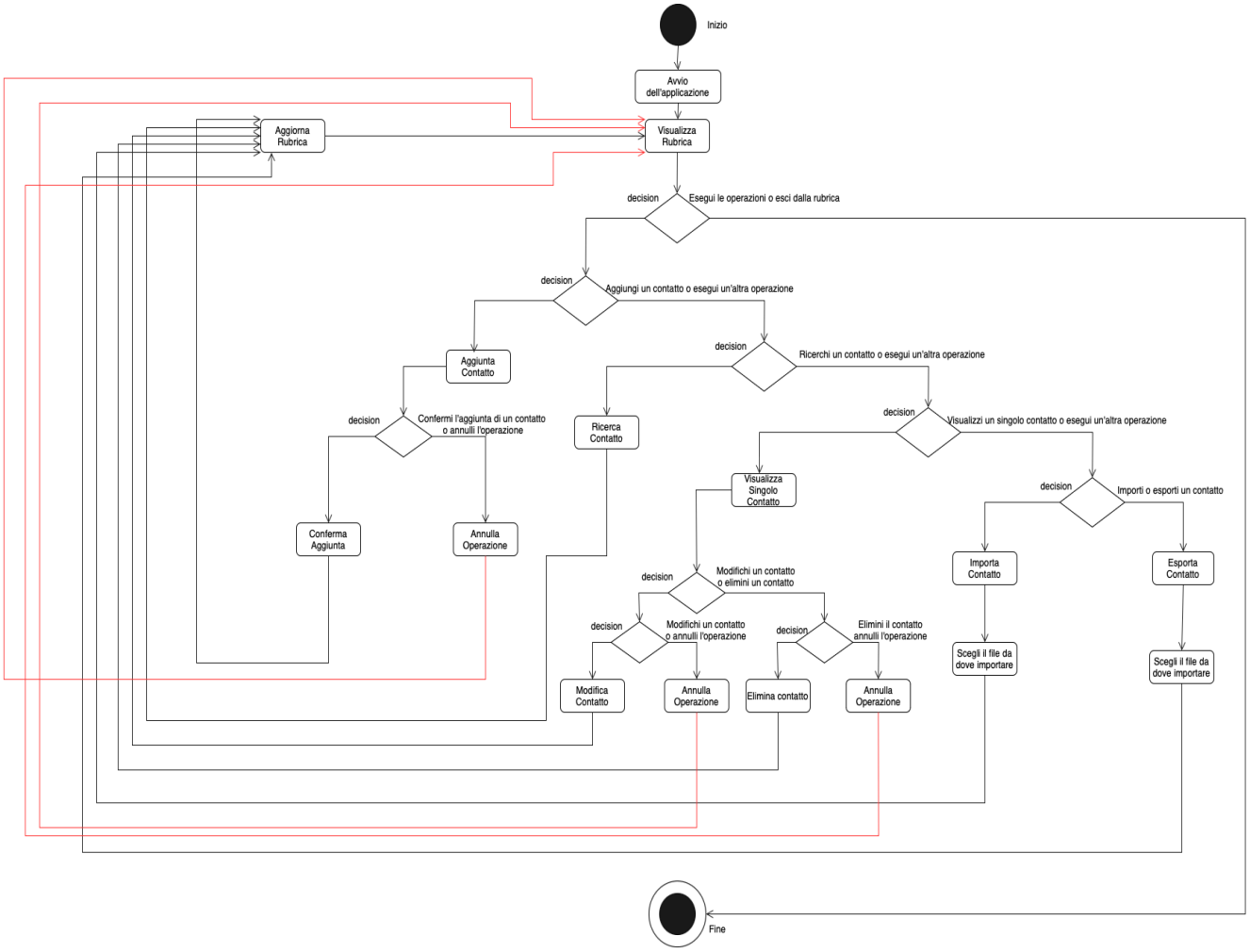
3. Diagramma dei Package



3.1 Commento al Diagramma Dei Package

Il diagramma dei package suddivide il sistema in moduli principali: ‘main’, ‘interface’, ‘data’ e ‘annullaoperazione’. Questa struttura mostra una buona separazione delle preoccupazioni (Separation of Concerns). Ogni package ha una responsabilità chiara: ‘main’ avvia l’applicazione, ‘interface’ gestisce i controller e le interfacce grafiche, ‘data’ contiene le entità principali come Rubrica e Contatto, e ‘annullaoperazione’ gestisce l’operazione di annullamento.

4. Diagramma delle attività



4.1 Commento al Diagramma delle Attività

Il diagramma delle attività rappresenta un flusso operativo robusto, conforme al:

- Principio di robustezza: Gestione chiara degli errori e percorsi definiti per prevenire stati inconsistenti.
- DRY: Le decisioni e attività sono rappresentate una sola volta, evitando duplicazioni.
- Evitare l'ottimizzazione precoce: Il focus è sulla funzionalità corretta prima di eventuali ottimizzazioni.

5. Decomposizione in Moduli

5.1 Modulo Main

Il modulo 'Main' contiene il punto di ingresso dell'applicazione, rappresentato dalla classe 'Applicazione'. Responsabilità principale: avviare il sistema e coordinare gli altri moduli.

Coesione: Funzionale – il modulo realizza un compito ben definito (avvio del sistema).

Accoppiamento: Per dati – comunica con altri moduli tramite parametri essenziali, riducendo le dipendenze.

5.2 Modulo Interface

Il modulo 'Interface' gestisce l'interazione utente ed è suddiviso in due sottosezioni: 'Controller' e 'FXML'. I controller gestiscono la logica, mentre la sezione FXML si occupa della rappresentazione grafica.

Coesione: Procedurale – i controller gestiscono operazioni frequentemente usate insieme.

Accoppiamento: Per controllo – dipende dalle specifiche dei moduli ‘Data’ per accedere ai dati.

5.3 Modulo Data

Il modulo ‘Data’ contiene le entità principali del sistema: ‘Rubrica’ e ‘Contatto’. La Rubrica fornisce operazioni come aggiunta, modifica, eliminazione e ricerca, mentre Contatto rappresenta ogni singolo elemento.

Coesione: Comunicazionale – tutte le operazioni lavorano sugli stessi dati (contatti e rubrica).

Accoppiamento: Per dati – espone solo i dati necessari tramite metodi specifici, minimizzando dipendenze dirette.

5.4 Modulo Annulla Operazione

Il modulo ‘Annulla Operazione’ include una classe progettata per gestire l'annullamento di operazioni, prevenendo stati inconsistenti e garantendo robustezza.

Coesione: Temporale – tutte le operazioni vengono eseguite durante una specifica fase del processo.

Accoppiamento: Nessuno – il modulo opera indipendentemente, interagendo solo quando richiesto dai controller.