Documentazione progetto Ingegneria del Software

Kuldeep Kaur VR 458207

Claudia Cretti VR 489163

Angel Ricardo Aguirre Vasquez VR 495299

Febbraio 2025

INDICE

Introduzione
3
Iterazioni utente-sistema 4
Scenari d’uso da parte del personale medico: 5
Scenari d’uso da parte del personale infermieristico: 9
Scenari d’uso da parte del personale di segreteria: 12
Scenari d’uso da parte dei pazienti:
24
Diagramma di attività e di classe: 24
Sviluppo e progettazione 26
Assunzioni e scelte progettuali 26
Processo di sviluppo 27
Progettazione e Pattern Architetturali 28
Implementazione e Design Pattern utilizzati 30
31
Test e validazione 35
Validazione e Controlli sui Dati 35
Test degli sviluppatori 36
Test degli utenti **37**

Introduzione

Il sistema software sviluppato è stato realizzato per ottimizzare la gestione dei servizi offerti dalla Casa della Salute, come le visite mediche, i prelievi e le medicazioni, al fine di migliorare l'efficienza e la qualità dell'assistenza sanitaria. Esso permette di coordinare le attività quotidiane della struttura, semplificando le operazioni e favorendo un rapido accesso alle informazioni mediche da parte di pazienti e operatori sanitari.

Il software è pensato per essere facilmente accessibile sia dai pazienti che dal personale della Casa della Salute. L'interfaccia utente è stata progettata per essere intuitiva e facile da navigare, affinché tutte le operazioni siano fruibili in modo rapido e sicuro.

Il rispetto della protezione dei dati e l'accesso controllato alle informazioni sono garantiti dal personale della Segreteria, che ha la possibilità di modificare le credenziali in qualsiasi momento, evitando così eventuali inconvenienti. Queste credenziali consentiranno agli utenti di accedere esclusivamente alle funzionalità a loro riservate, garantendo così la privacy e la sicurezza delle informazioni sensibili.

In sintesi, questo sistema software rappresenta una soluzione completa e affidabile per la gestione dei servizi sanitari all’interno della Casa della Salute, con un forte focus sulla facilità d'uso, la sicurezza dei dati e l'efficienza delle operazioni quotidiane.

Iterazioni utente-sistema

Scenari d'uso:
Per ciascun scenario d'uso, si considera che l'utente abbia già completato il processo di autenticazione, garantendo così l'accesso al sistema in modo sicuro. Inoltre, è previsto che ogni scenario d'uso possa essere interrotto in qualsiasi momento, sia mediante l'azione di logout, che consente all'utente di uscire in modo sicuro dal sistema, sia avviando un altro scenario d'uso, che interrompe l'attuale sessione per proseguire con un'altra operazione. I pazienti e il personale della Casa Salute (medico ed infermieristico) non possono visualizzare e né aggiornare i propri dati personali, poiché tale responsabilità è esclusivamente affidata alla Segreteria. Ad esempio l'indirizzo e-mail, essendo un dato fondamentale per la comunicazione, non può essere modificato autonomamente.

Scenari d'uso da parte del personale medico:

Il personale medico accede al sistema tramite le proprie credenziali di autenticazione e ha la possibilità di consultare lo storico delle visite effettuate, nonché visualizzare l'elenco dei pazienti convenzionati e dei pazienti dei medici che sostituisce. Il medico può visualizzare le prenotazioni effettuate dai pazienti, inserire gli esiti delle visite mediche nel sistema e ha i privilegi per pianificare e gestire le proprie assenze tramite le funzionalità dedicate.

Scenari d'uso: Visualizza storico visite
Attore: Medico
Precondizione: La visita deve essere presente nel database del sistema
Sequenza degli eventi: 1. Clicca su "Visualizza storico visite" 2. Filtra per l'anno desiderato 3. Clicca sul riquadro della visita interessata di cui si vuole visualizzare il pdf

Scenari d'uso: Inserisci esito della visita
Attore: Medico
Precondizione: La prenotazione e il paziente deve essere presente nel database del sistema
Sequenza degli eventi: 1. Clicca su "Visualizza storico visite" 2. Clicca su "Inserisci Esito"

3. Clicca sulla visita prenotata di cui si vuole inserire l’esito 4. Compila i campi: urgenza, sintomi, diagnosi. Terapia prescritta e note aggiuntive 5. Clicca su “Salva” 6. Generazione del pdf automatico
Postcondizione: La visita viene salvata nel sistema

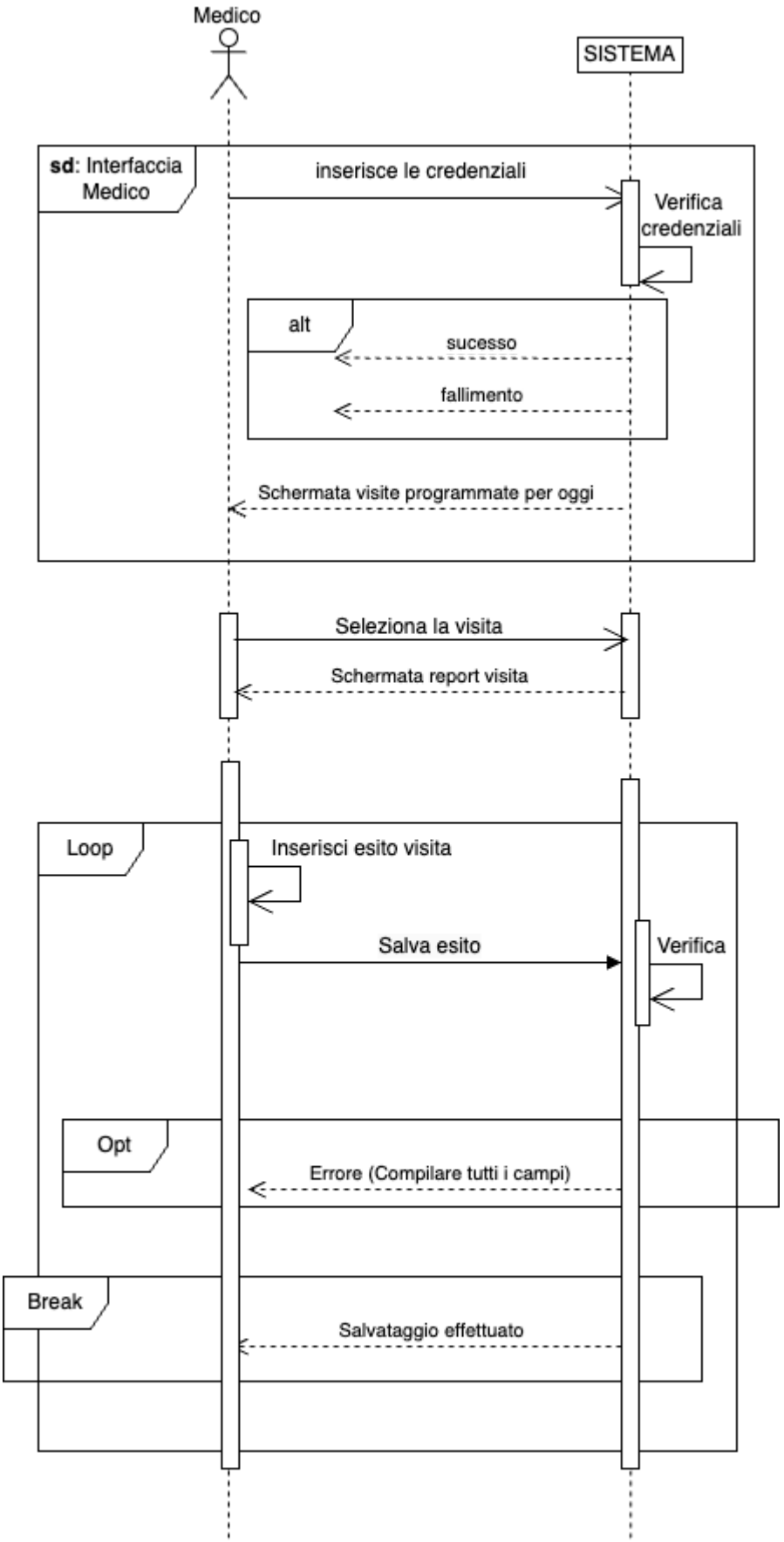


Diagramma di sequenza per lo scenario d'uso:
Inserisci esito della visita

Scenari d’uso: Visualizza storico visite
Attore: Medico
Precondizione: La visita e il paziente devono essere presente nel database del sistema
Sequenza degli eventi: 1. Clicca su “Pazienti convenzionati” 2. Clicca sulla riga del paziente di cui si desidera visionare le visite 3. Filtra per l’anno desiderato 4. Clicca sul riquadro della visita interessata di cui si vuole visualizzare il pdf

Scenari d’uso: Inserisci / Modifica le assenze pianificate
Attore: Medico
Sequenza degli eventi: 1. Clicca su “Pianifica Assenze” 2. Filtra per l’anno desiderato 3. Clicca sul giorno per cui si desidera pianificare l’assenza 4. Clicca su “Salva”
Postcondizione: L’assenza viene salvata nel sistema

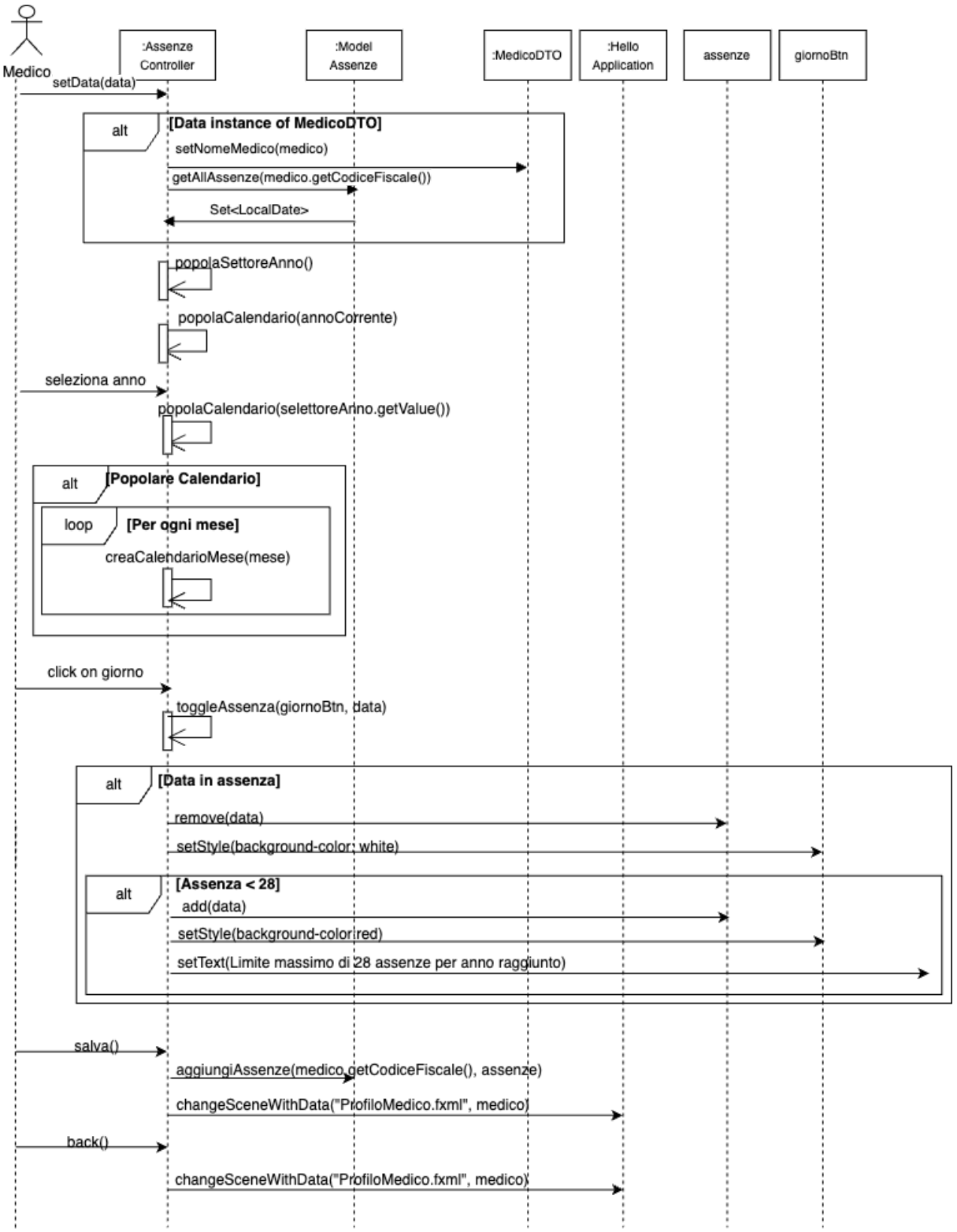
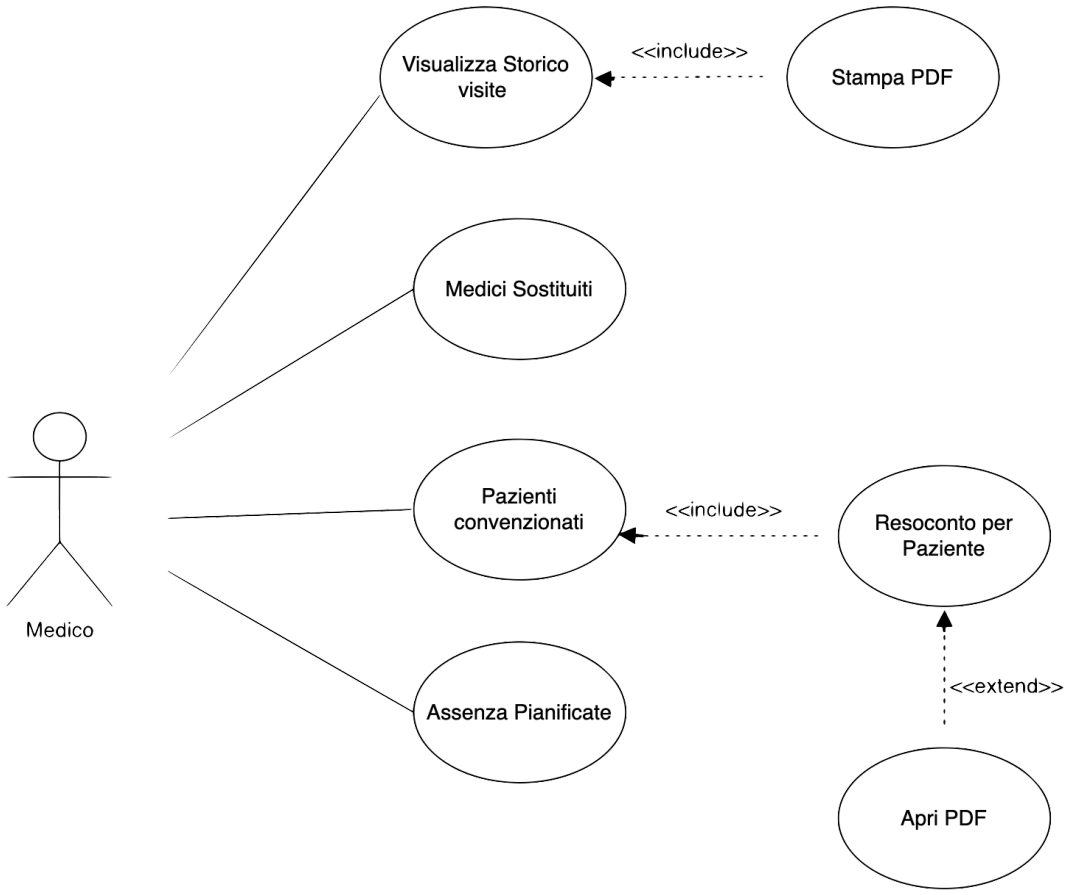


Diagramma di sequenza che gestisce le assenze pianificate del medico

Scenari d’uso: Visite programmate per oggi
Attore: Medico

Precondizione: La prenotazione deve essere presente nel database del sistema
Sequenza degli eventi: 1. Accedi al sistema 2. Visualizza il riquadro a destra “Visite programmate per oggi ” 3. Clicca sulla riga della visita di cui si vuole inserire l’esito 4. Compila i campi: urgenza, sintomi, diagnosi. Terapia prescritta e note aggiuntive 5. Clicca su “Salva” 6. Generazione del pdf automatico
Postcondizione: La visita viene salvata nel sistema

Scenari d’uso: Pazienti convenzionati ai medici sostituti
Attore: Medico
Precondizione: I medici sostituiti e i relativi pazienti devono essere presente nel database del sistema
Sequenza degli eventi: 1. Accedi al sistema 2. Clicca sul riquadro del medico che sostituisce 3. Visualizza i pazienti a lui convenzionati



Scenario d’uso dei medici

Scenari d’uso da parte del personale infermieristico:

Gli infermieri accedono al sistema utilizzando le credenziali (e-mail, password) e hanno la possibilità di visualizzare lo storico delle prenotazioni, nonché i giorni di servizio e riposo a loro assegnati dalla segreteria. Possono consultare le prestazioni (prelievi/medicazioni) programmate per la giornata corrente e inserire gli esiti delle prestazioni effettuate nel sistema.

Scenari d’uso: Visualizza storico prestazioni
Attore: Infermiere
Precondizione: La prestazione deve essere presente nel sistema
Sequenza di eventi: 1. Clicca su “Visualizza storico prestazioni “ 2. Filtre per l’anno desiderato

3. Clicca sul riguardo della prestazione di cui si desidera visualizzare il pdf

Scenari d'uso: Inserisci prestazione

Attore: Infermiere

Precondizione: La prestazione deve essere presente nel sistema

Sequenza di eventi:

1. Clicca su "Visualizza storico prestazioni "
2. Clicca su "Aggiungi prestazione"
3. Clicca sulla riga della prenotazione di cui si desidera inserire l'esito
4. Compila i campi: esito e note aggiuntive
5. Clicca su "Salva"
6. Generazione del pdf automatica

Postcondizione: L'esito della prestazione viene salvata nel sistema

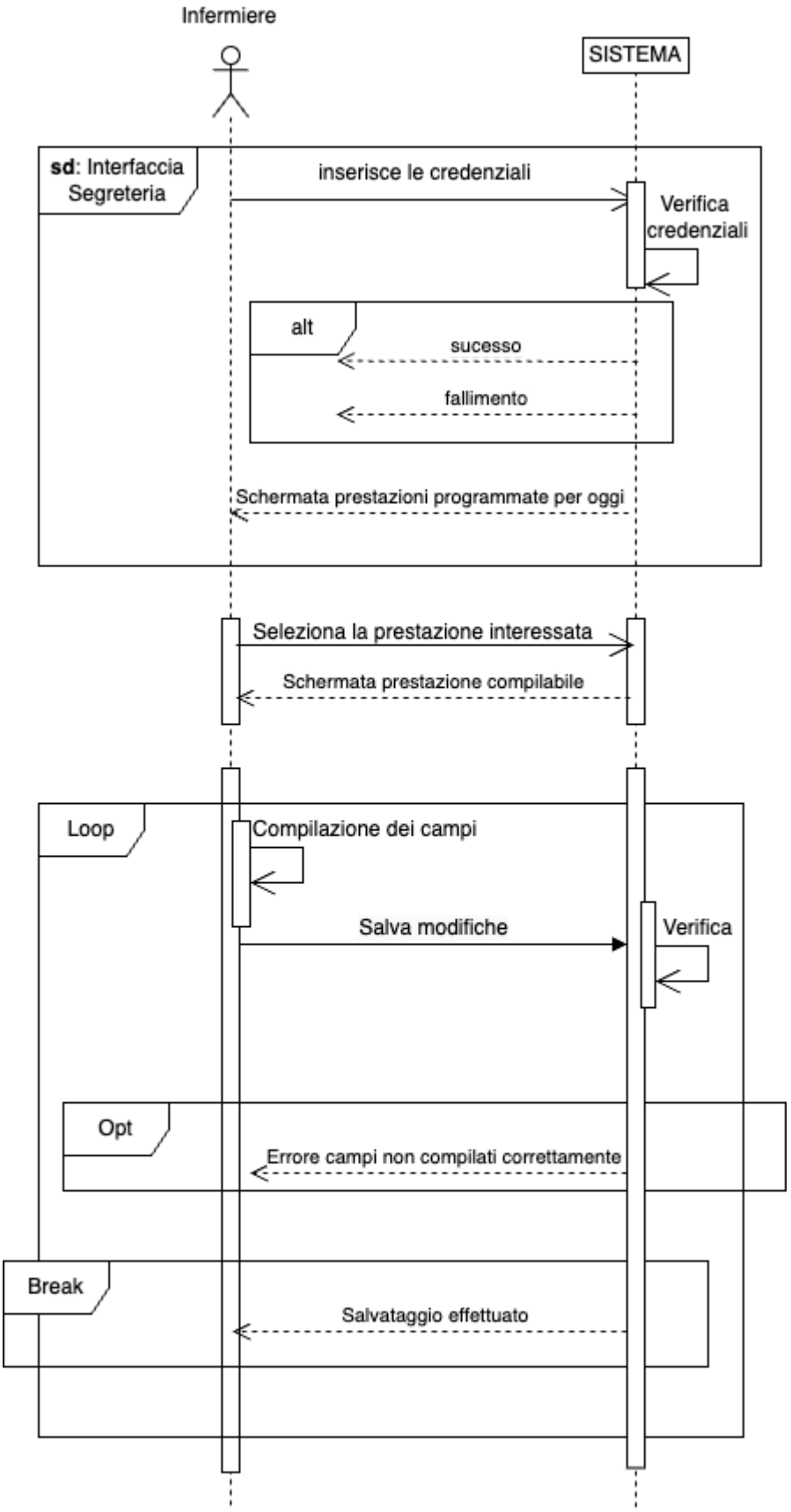
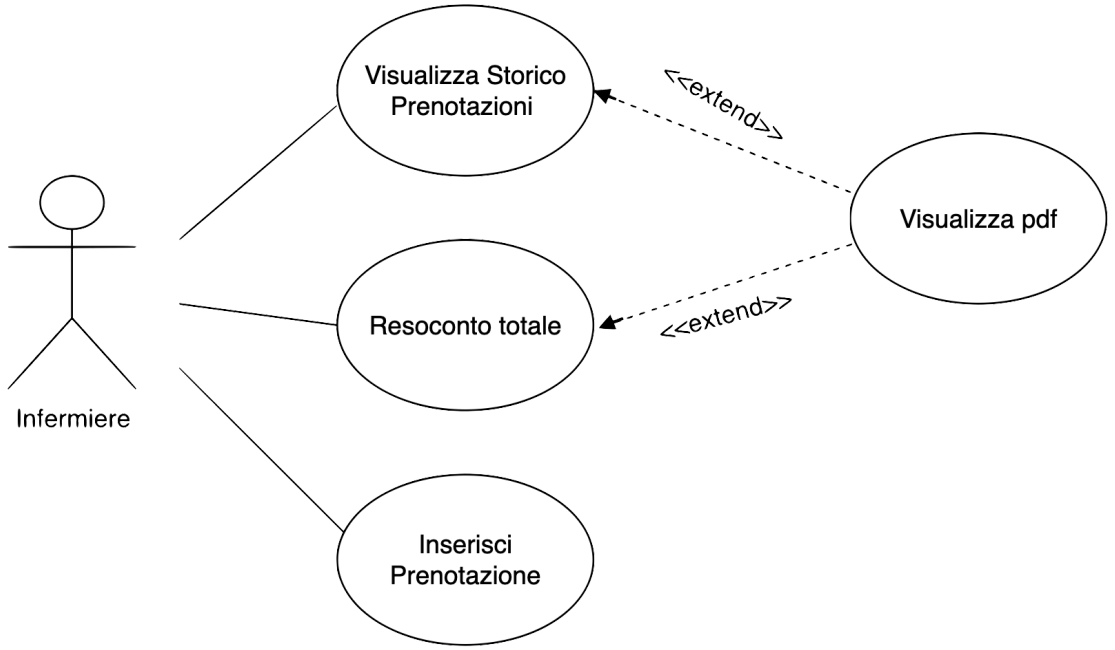


Diagramma di sequenza per lo scenario d'uso:
Inserisci prestazione

Scenari d'uso: Resoconto Pazienti
Attore: Infermiere
Precondizione: La prestazione deve essere presente nel sistema
Sequenza di eventi: 1. Clicca su "Visualizza storico prestazioni " 2. Filtra per l'anno desiderato 3. Clicca sul riguardo della prestazione di cui si desidera visualizzare il pdf

Scenari d'uso: Prestazioni programmate per oggi
Attore: Infermiere
Precondizione: La prestazione deve essere presente nel sistema
Sequenza di eventi: 1. Accedi al sistema 2. Visualizza il riquadro a destra 3. Clicca sulla riga della prenotazione di cui si desidera inserire l'esito 4. Compila i campi: esito e note aggiuntive 5. Clicca su "Salva"
Postcondizione: L'esito della prestazione viene salvata nel sistema



Scenario d’uso degli infermieri

Scenari d’uso da parte del personale di segreteria:

Il personale di segreteria è autorizzato ad aggiornare i dati relativi ai pazienti, ai medici e agli infermieri nel sistema. Inoltre, dispone dei permessi per accedere e visualizzare il resoconto dettagliato per ogni paziente e lo storico dei referti associati. Il sistema consente al personale di segreteria anche di registrare nuovi medici e infermieri, ampliando così il database degli utenti con specifiche credenziali di accesso.

Scenari d’uso: Modifica dati medico
Attore: Segreteria
Precondizione: Il medico deve essere presente nel database del sistema
Sequenza degli eventi: 1. Clicca su “Dati medici” 2. Clicca sulla riga del medico a cui sei interessato: Modifica i campi: email, password, specialità e i medici che lo sostituiscono Clicca sul pulsante “Salva”

Postcondizione: Salva le modifiche nel sistema

Scenari d’uso: Aggiungi medico
Attore: Segreteria
Sequenza degli eventi: 1. Clicca su “Dati medici” 2. Clicca su “Aggiungi Medico” 3. Compila i campi: nome, cognome, codice fiscale, specialità, email, password e i medici che lo sostituiscono. 4. Clicca sul pulsante “Salva” 5.
Postcondizione: Salva le modifiche nel sistema

Scenari d’uso: Modifica dati pazienti
Attore: Segreteria
Precondizione: Il paziente deve essere presente nel database del sistema
Sequenza degli eventi: 1. Clicca su “Dati Pazienti” 2. Clicca sulla riga del paziente a cui vuoi apporre modifiche 3. Modifica i campi: e-mail, password, e-mail del tutore e il medico curante. 4. Clicca sul pulsante “Salva”
Postcondizione: Salva le modifiche nel sistema

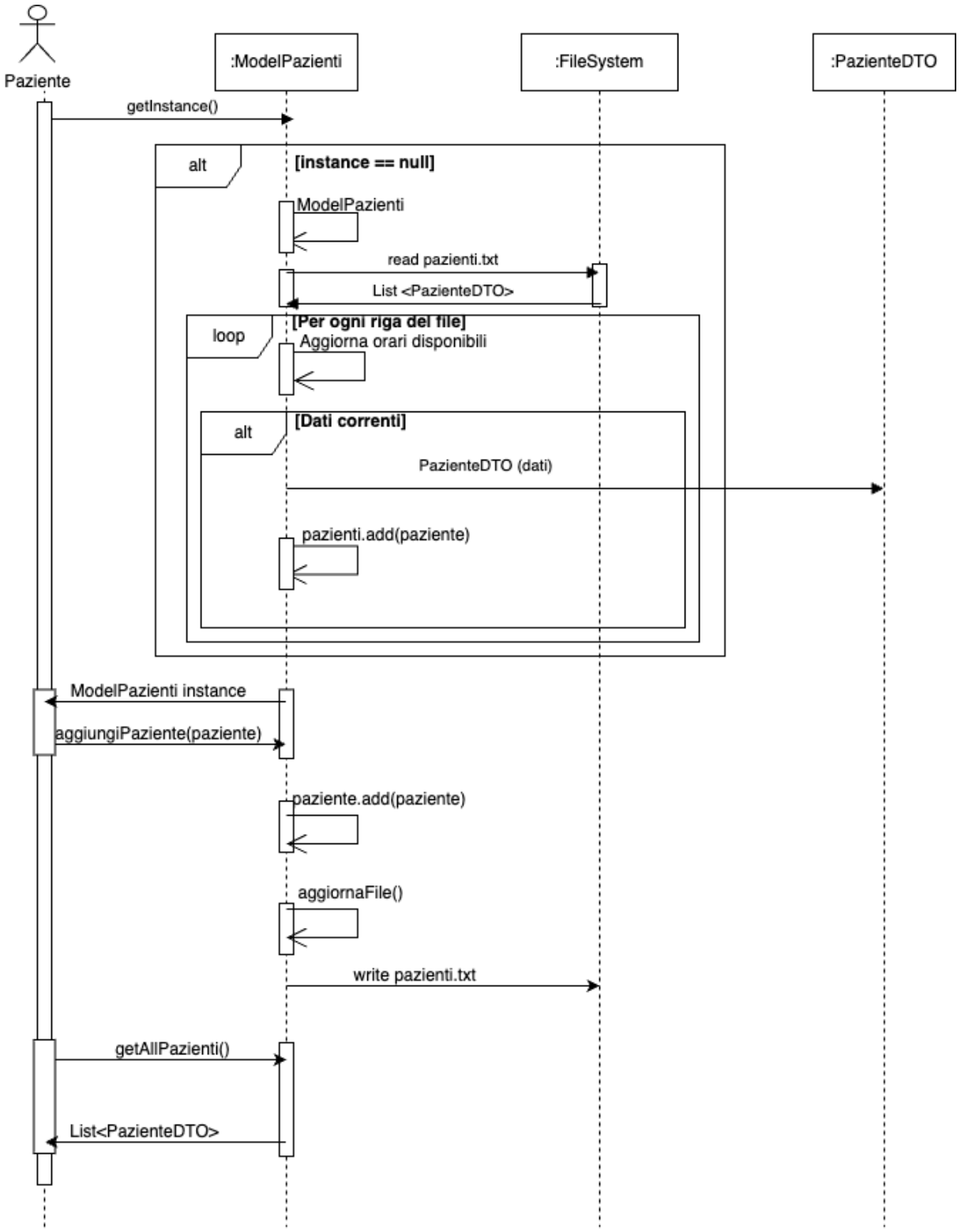


Diagramma di sequenza per la gestione dei file txt in cui vengono salvati e modificati i dati dei pazienti

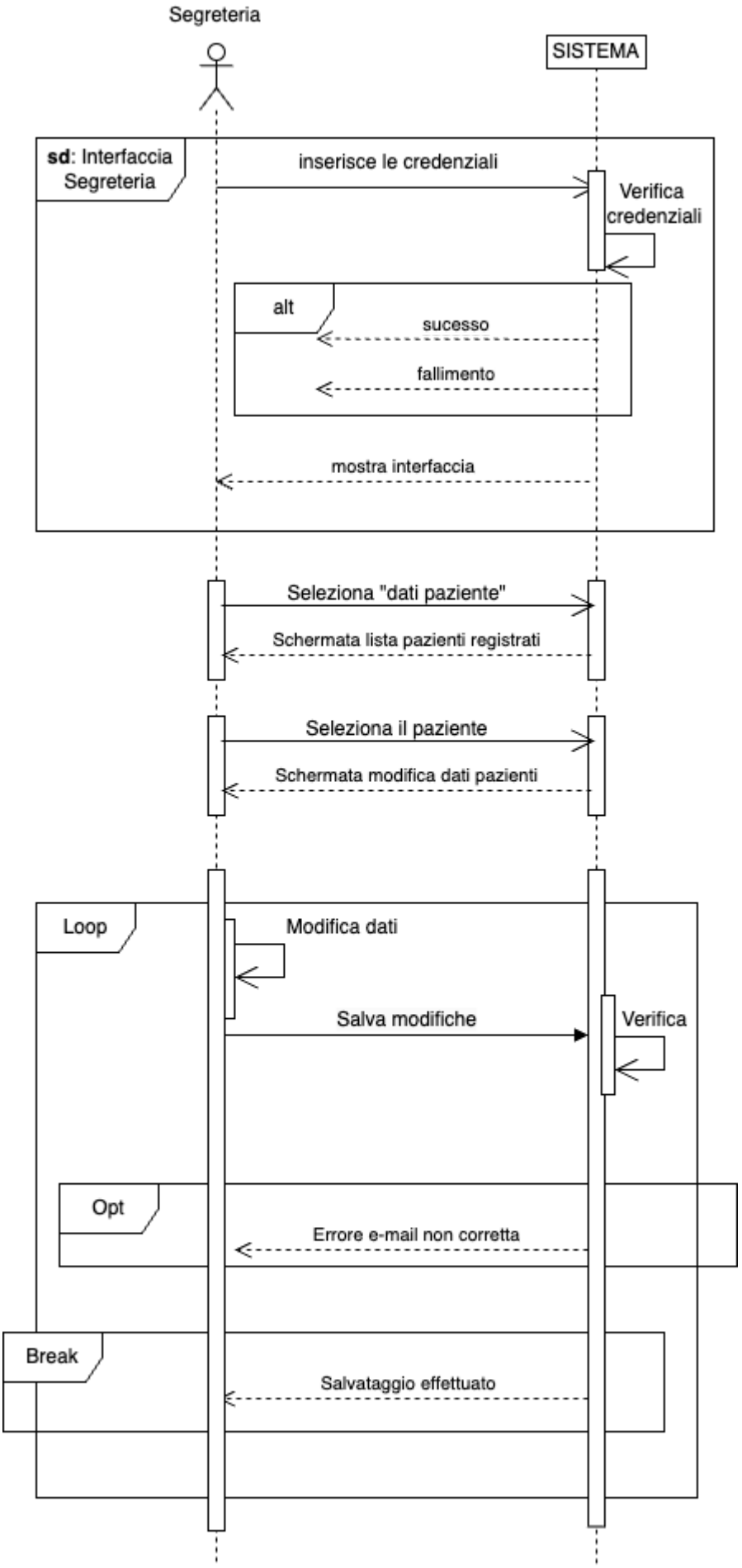


Diagramma di sequenza per lo scenario d'uso:
Modifica dati paziente/infermiere/medico

Scenari d’uso: Modifica dati infermieri
Attore: Segreteria
Precondizione: Il paziente deve essere presente nel database del sistema
Sequenza degli eventi: 1. Clicca su “Dati infermieri” 2. Clicca sulla riga dell’infermiere a cui vuoi apporre modifiche 3. Modifica i campi: e-mail, password e stabilisci i giorni della settimana in cui l’infermiere svolgerà la mansione di prelievo o medicazione 4. Rispetta i vincoli lasciando un solo giorno di riposo libero 5. Clicca su “Salva”
Postcondizione: Salva le modifiche nel sistema

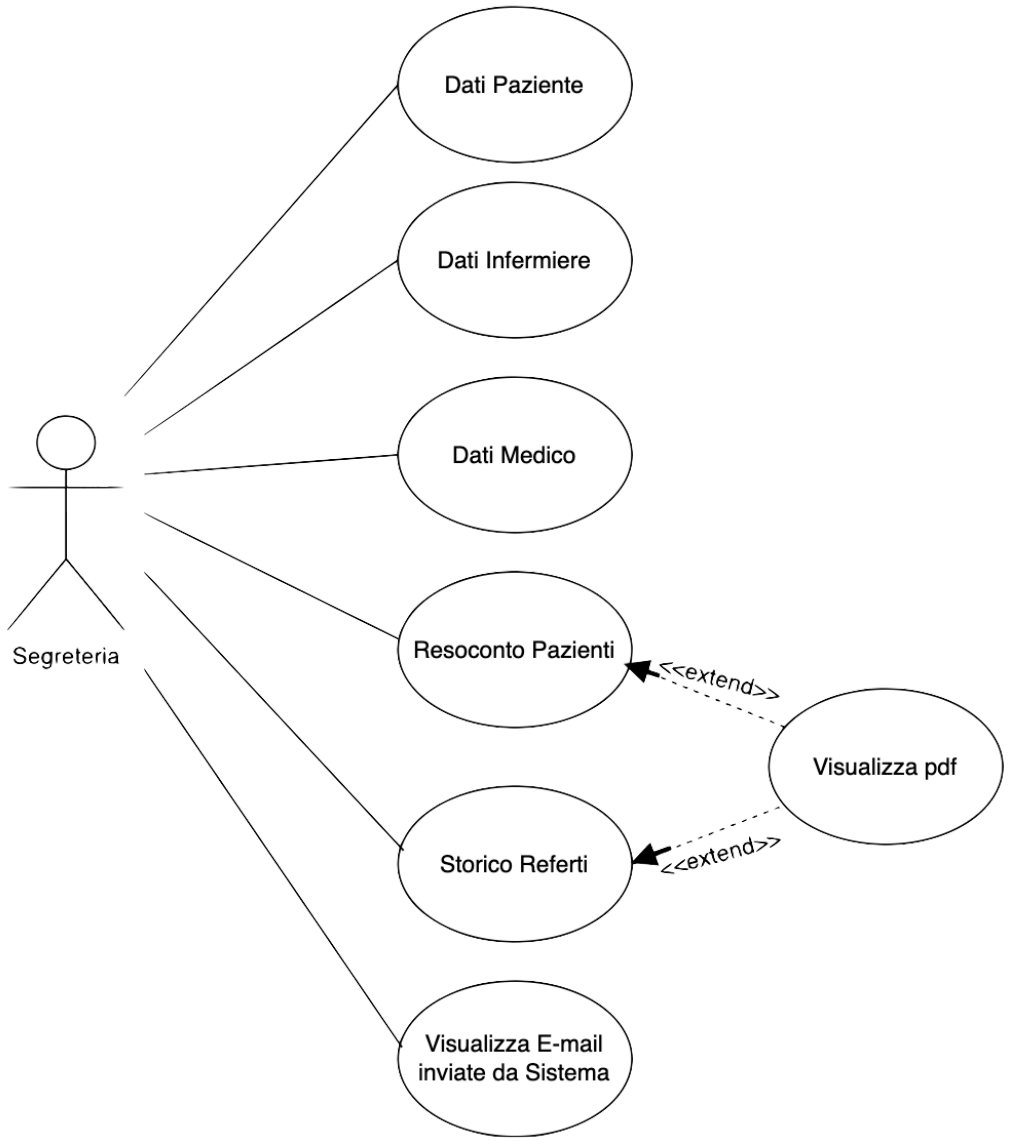
Scenari d’uso: Aggiungi infermiere
Attore: Segreteria
Sequenza degli eventi: 1. Clicca su “Dati infermiere” 2. Clicca su “Aggiungi infermiere” 3. Compila i campi: nome, cognome, codice fiscale, email e password. 4. Clicca sul pulsante “Salva”
Postcondizione: Salva le modifiche nel sistema

Scenari d’uso: Visualizza resoconto dei pazienti
Attore: Segreteria
Precondizione: Le visite dei relativi pazienti devono essere presenti nel database del sistema
Sequenza degli eventi:

1. Clicca su “Resoconto dei pazienti”
2. Clicca sulla riga del paziente di cui si vuole visualizzare il resoconto
3. Filtra per l’anno desiderato
4. Clicca sul riquadro della visita per aprire il formato pdf della visita.
5. Clicca su “Torna indietro”

Scenari d’uso: Visualizza Storico Referti
Attore: Segreteria
Precondizione: Le visite devono essere presenti nel database del sistema
Sequenza degli eventi: 1. Clicca su “Storico referti” 2. Filtra per l’anno desiderato 3. Clicca sul riquadro per aprire il formato pdf della visita 4. Clicca su “Torna indietro”

Scenari d’uso: Visualizza e-mail inviate da sistema
Attore: Segreteria, Paziente
Precondizione: L’esito della visita deve essere memorizzato nel sistema
Sequenza degli eventi: 1. Accedi al sistema 2. Nel riquadro a destra visualizza le e-mail arrivate 3. Clicca sulla mail interessata per visualizzarne il contenuto



Scenario d’uso da parte della segreteria

Scenari d’uso da parte dei pazienti:

Gli utenti accedono al sistema utilizzando le credenziali (e-mail, password) e hanno la possibilità di effettuare prenotazioni per visite mediche, prelievi o medicazioni. Possono consultare lo storico delle prestazioni erogate, scaricare gli esiti delle visite in formato PDF e gestire la ricezione e l'invio di notifiche via e-mail. Gli utenti maggiorenni hanno la facoltà di associare pazienti minori di 14 anni al proprio profilo, per la gestione e prenotazione delle relative prestazioni.

Scenari d'uso: Visualizza prestazioni
Attore: Paziente
Precondizione: Le prestazioni in programma sono salvate nel sistema
Sequenza di eventi: 1. Clicca su "Prenotazione Visita" e controlla quanto prenotato

Scenari d'uso: Registrazione paziente
Attore: Paziente
Sequenza di eventi: 1. Clicca sul link per la registrazione 2. Se il paziente > 18 anni: Compilare i campi, clicca su "registrati" 3. Se il paziente >= 14 e < 18: Compilare i campi, clicca su "registrati", inserisci l'Email del tutore
Postcondizione: Le credenziali vengono memorizzate nel sistema

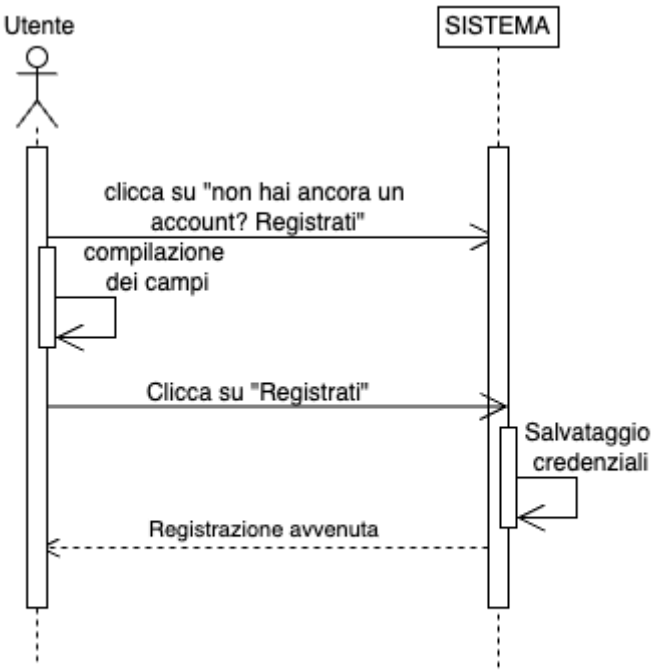


Diagramma di sequenza per lo scenario d'uso:
Login Utente >= 18

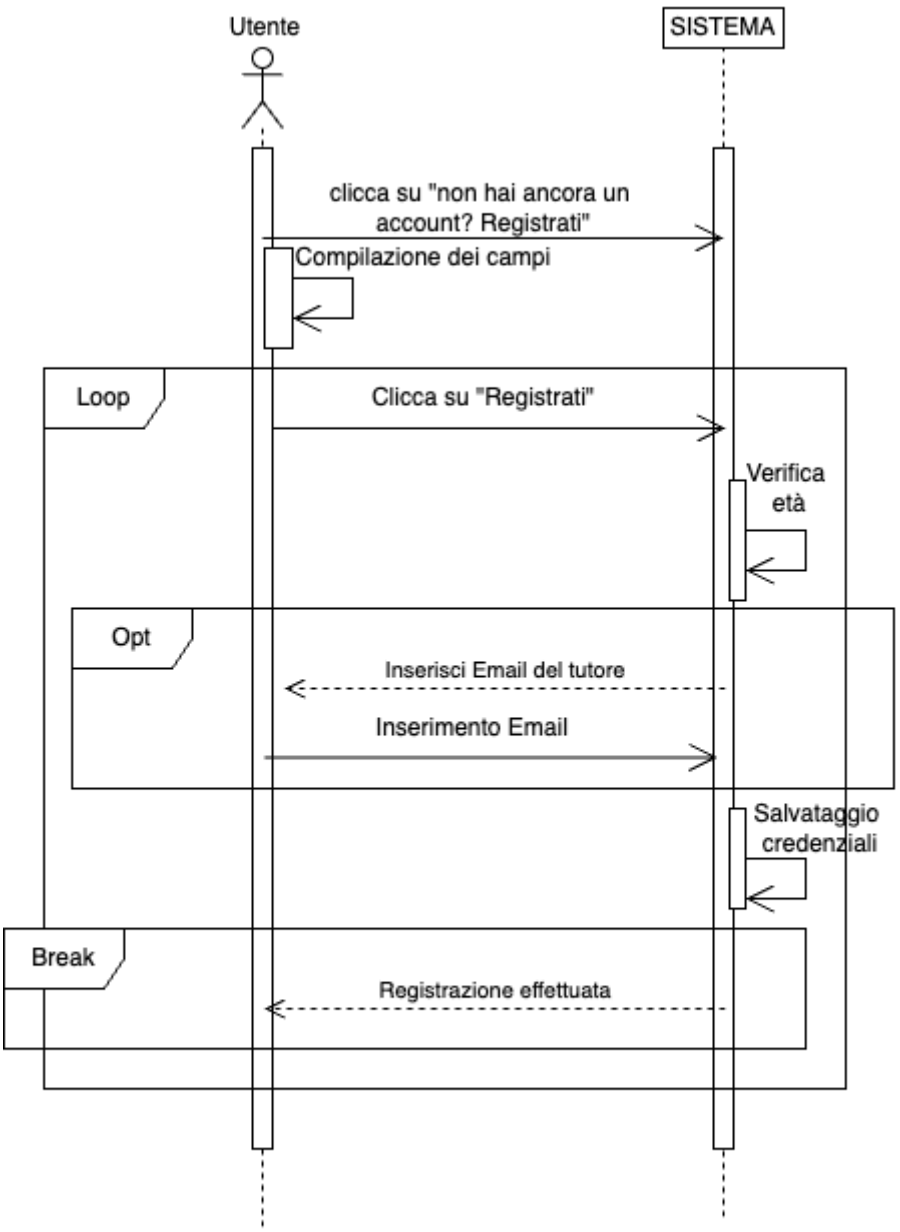


Diagramma di sequenza per lo scenario d'uso:
Login Utente per età compresa tra 14 e 18 escluso

Scenari d'uso: Effettua Prenotazione Visita
Attore: Paziente
Sequenza di eventi: 1. Clicca su "Prenota Visita" 2. Seleziona la data della visita 3. Seleziona l'orario, se il medico non è presente, selezionare il medico sostitutivo disponibile

4. Clicca su “Conferma Prenotazione
Postcondizione: La prenotazione è memorizzata nel sistema.

Scenari d’uso: Effettua Prenotazione Prelievi/Medicazioni
Attore: Paziente
Sequenza di eventi: 1. Clicca su “Prenota prestazione” 2. Seleziona la data della visita 3. Seleziona il tipo di prestazione tra prelievo o medicazione 4. Seleziona l’orario della prenotazione 5. Clicca su “Conferma prenotazione”
Postcondizione: La prenotazione è memorizzata nel sistema

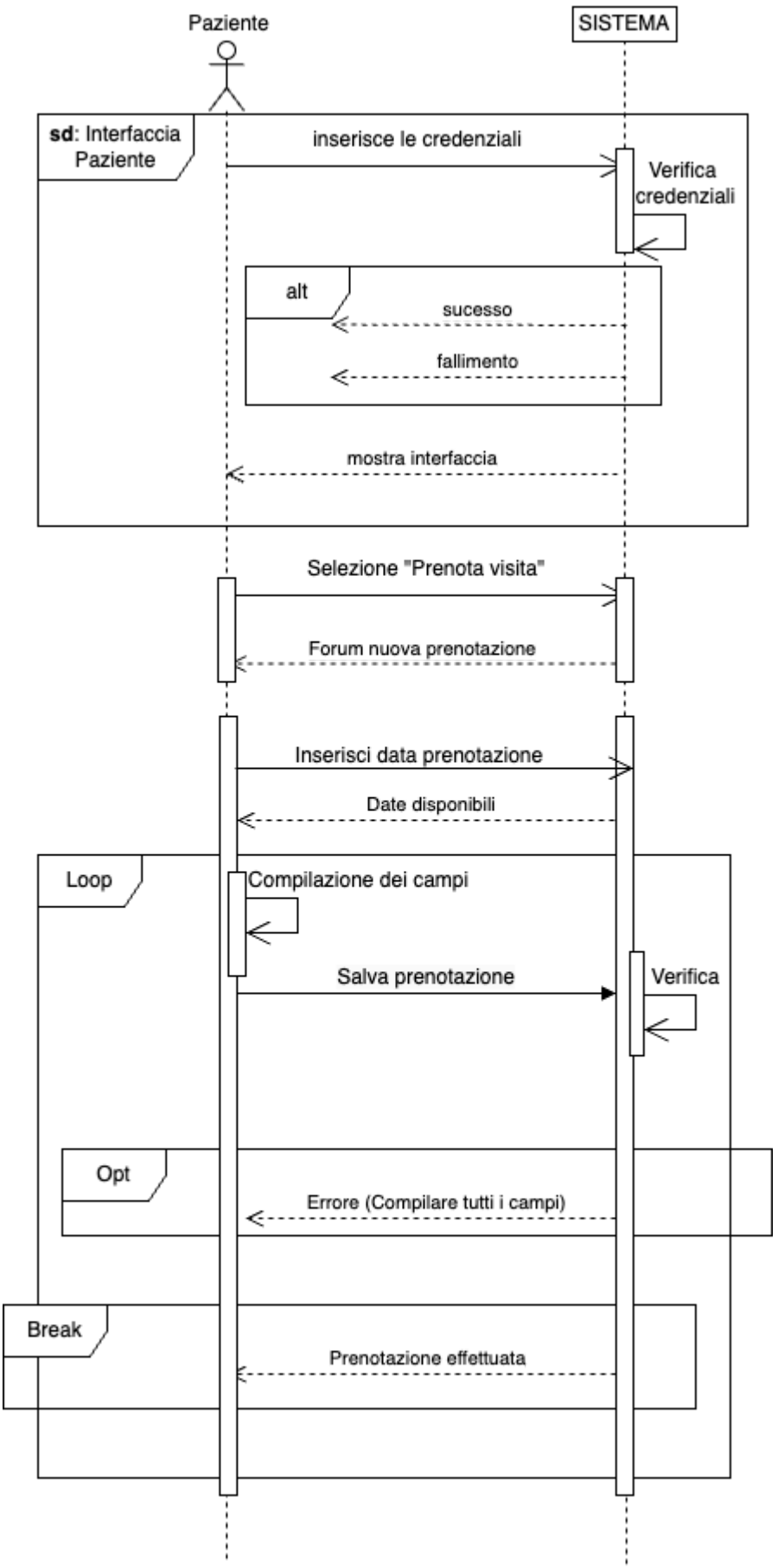


Diagramma di sequenza per lo scenario d'uso:
Effettua Prenotazione Visita e Prenotazione Prelievi o Medicazioni

Scenari d'uso: Stampa Esito Visita
Attore: Paziente
Precondizione: Le visite mediche sono memorizzate nel sistema
Sequenza di eventi: 1. Nel riquadro a destra, filtra per l'anno desiderato 2. Clicca sulla visita 3. File "Stampa"

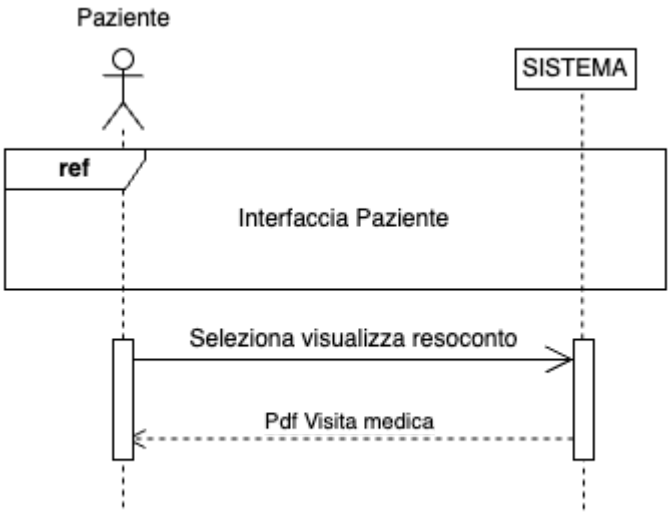


Diagramma di sequenza per lo scenario d'uso: stampa esito visita

Scenari d'uso: Aggiungi figli a carico
Attore: Paziente

Sequenza di eventi:
 	
1. Clicca su "Nucleo familiare"
2. Inserisci:
  - il nome,
  - cognome,
  - data di nascita
  - il luogo di nascita
3. Clicca su "Salva"

Postcondizione: Viene memorizzato nel sistema il paziente a carico dell'utente

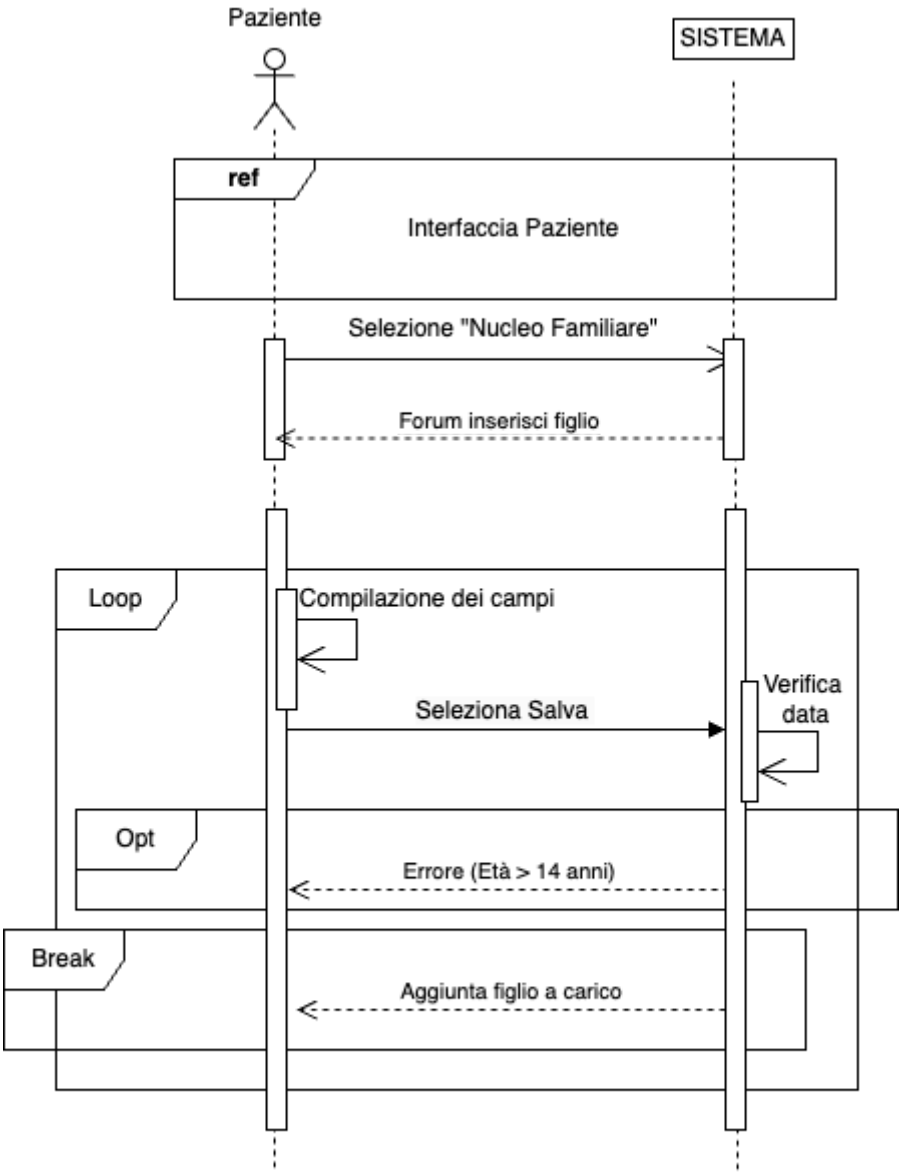
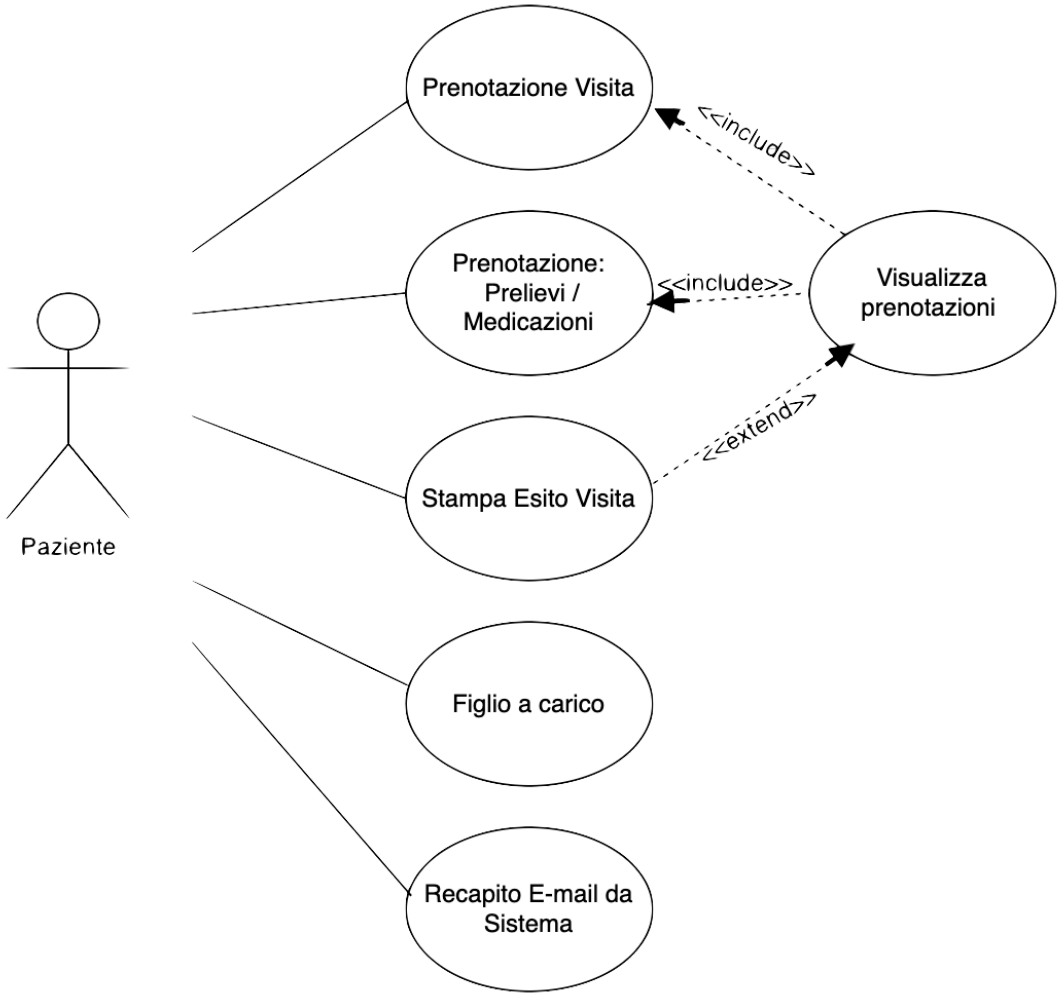


Diagramma di sequenza per lo scenario d'uso: Aggiungi figlio

Scenari d'uso: Visualizza E-mail inviate da sistema
Attore: Paziente, Segreteria
Precondizione: L'esito delle visite deve essere memorizzato nel sistema
Sequenza di eventi: 1. Accedi al sistema 2. Nel riquadro a destra visualizza le e-mail arrivate 3. Clicca sulla mail interessata per visualizzarne il contenuto



Scenario d’uso da parte dei pazienti

Diagramma di attività:

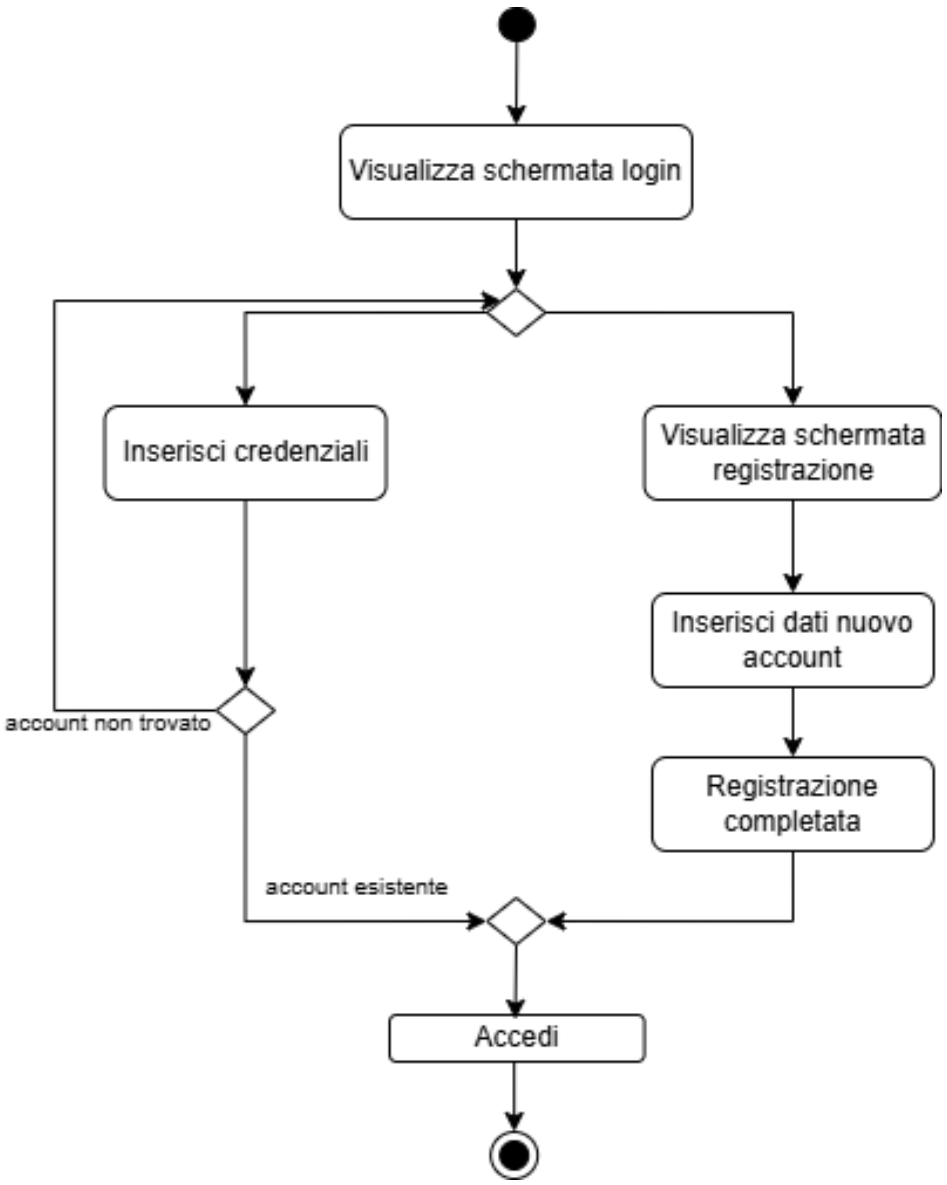


Diagramma delle attività dell'accesso iniziale con eventuale registrazione

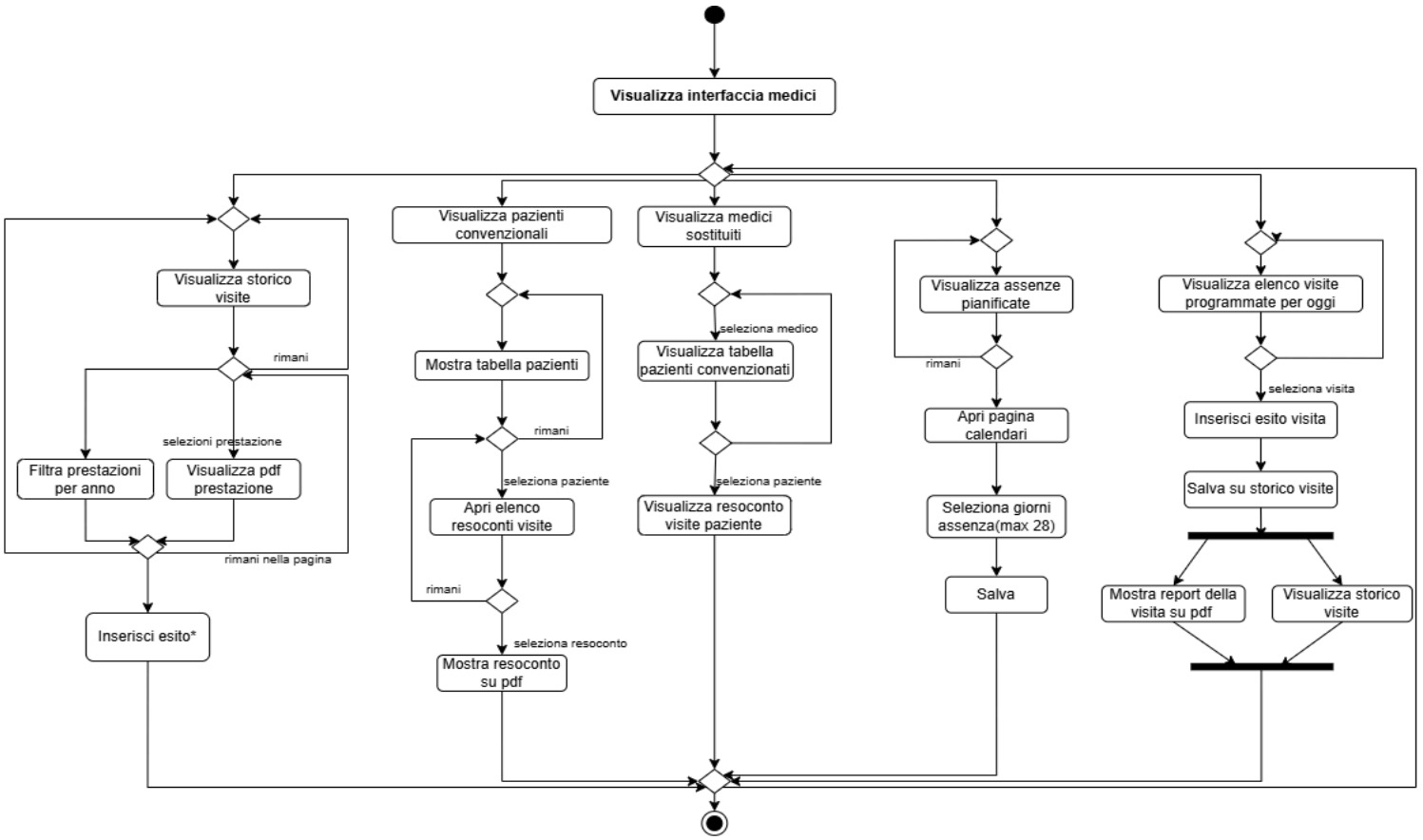


Diagramma delle attività dell'intera interfaccia Medici

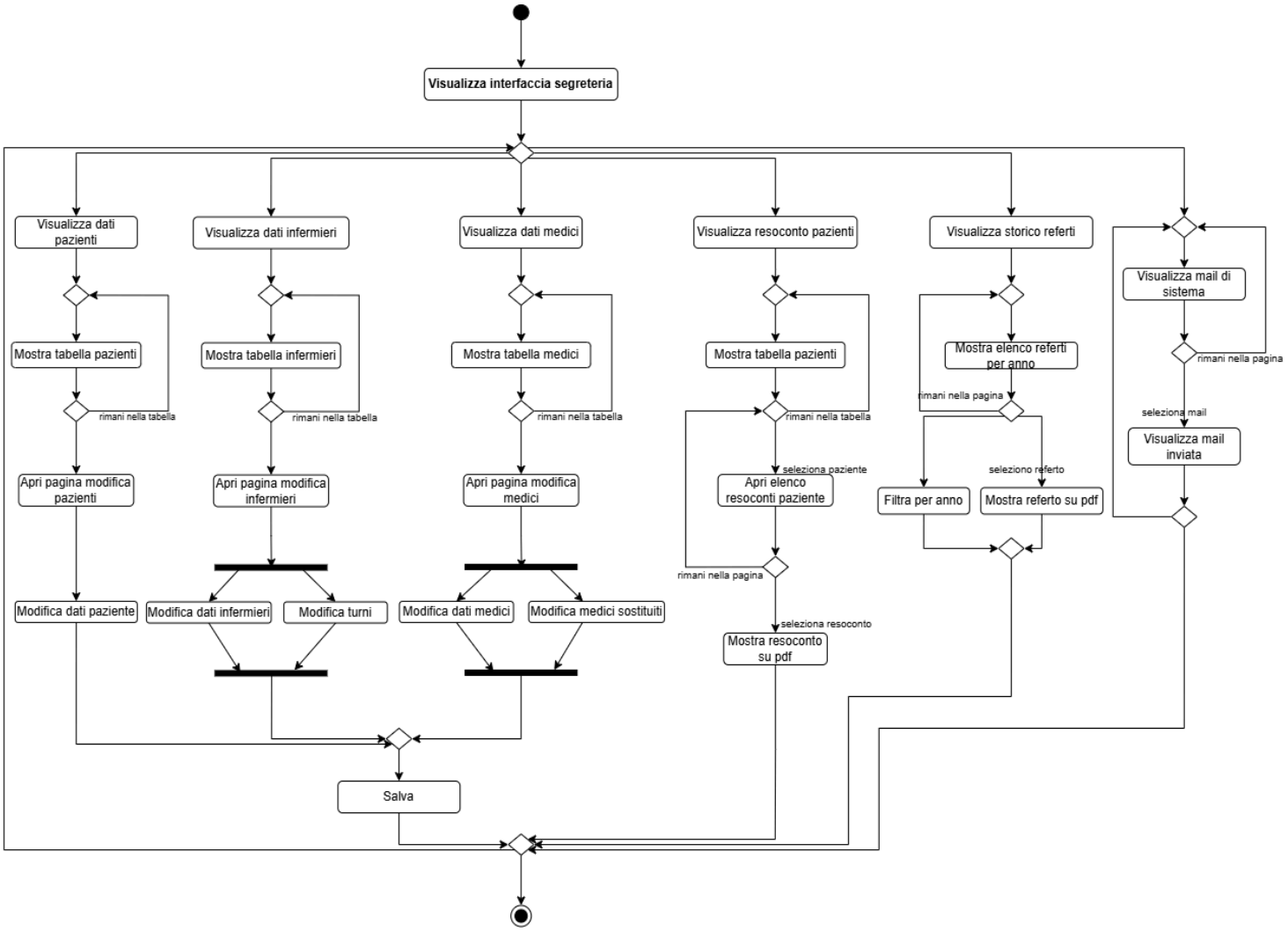


Diagramma delle attività dell’intera interfaccia segreteria

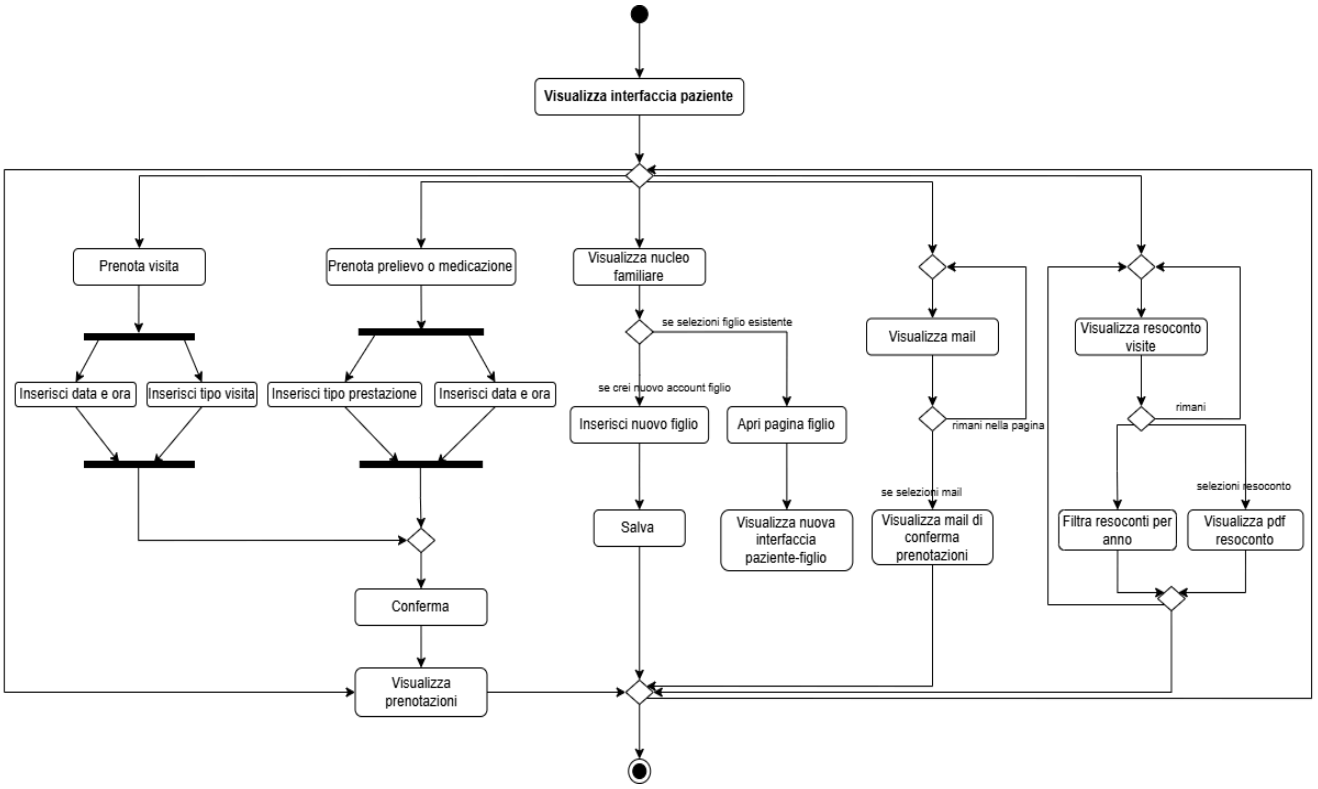


Diagramma delle attività dell'intera interfaccia Paziente

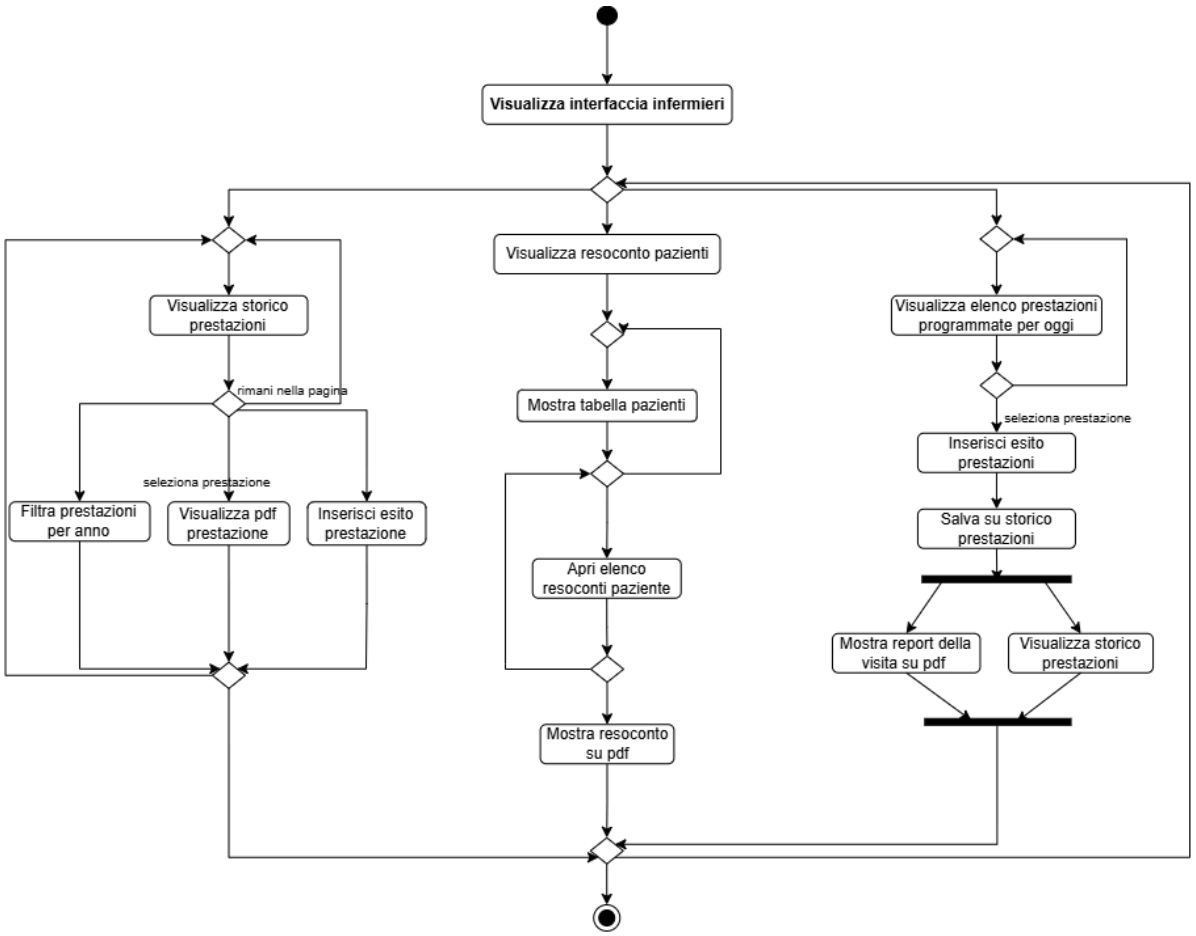
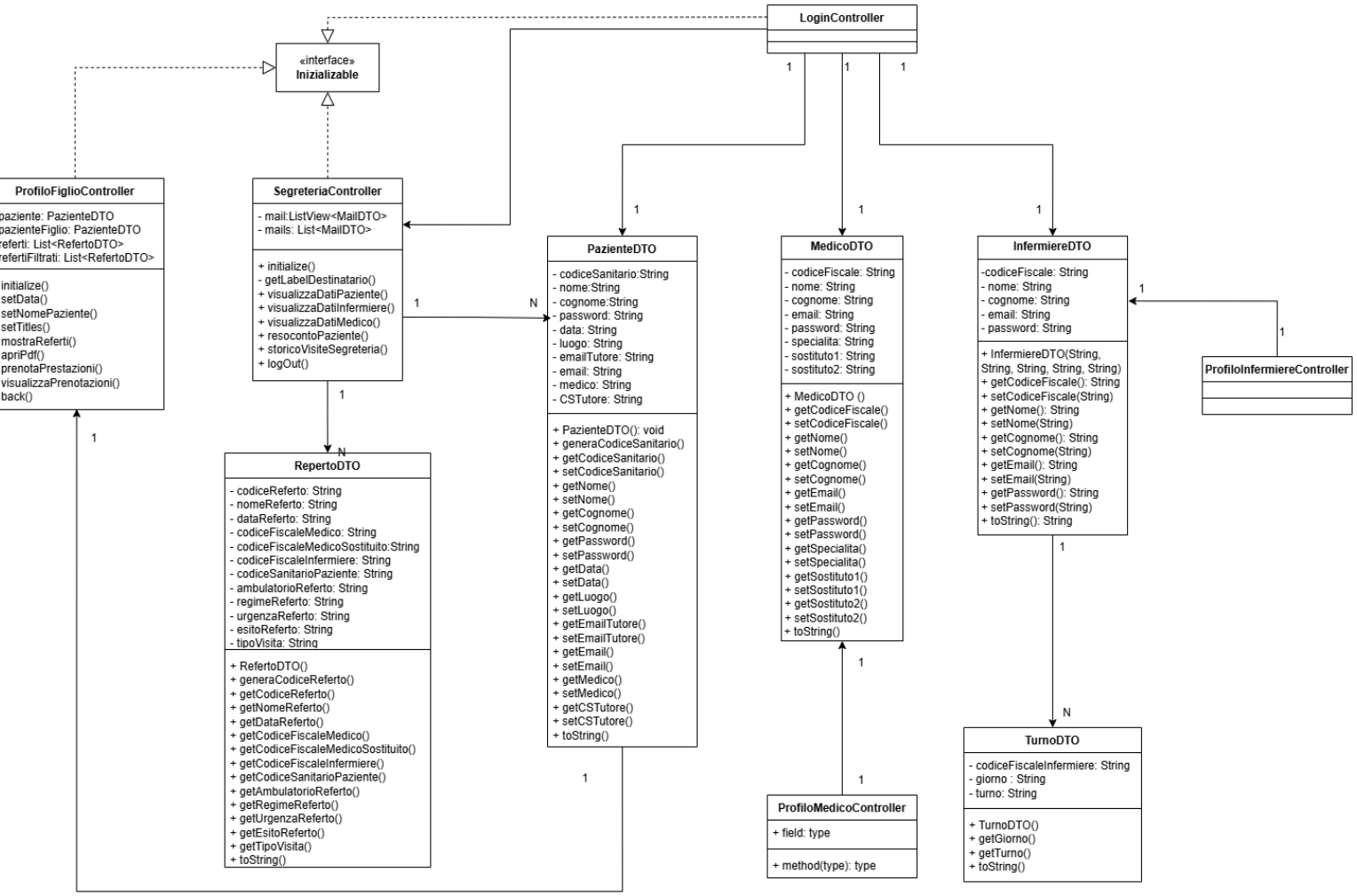


Diagramma delle attività dell'intera interfaccia Infermieri

Diagramma delle classi:



Sviluppo e progettazione

Assunzioni e scelte progettuali

Si assume che l'inserimento di un paziente minorenne con età inferiore ai 14 anni avvenga esclusivamente tramite il portale del genitore, che ne effettua la registrazione all'interno del nucleo familiare. Questa procedura garantisce l'assegnazione automatica dello stesso medico di base sia al genitore che al figlio, con un aggiornamento dinamico in caso di modifica del medico di base del tutore.

Per i pazienti di età compresa tra 14 e 17 anni, il sistema richiede l'indirizzo e-mail del tutore per completare la registrazione. La gestione delle assenze consente ai medici di base di pianificare fino a 28 giorni di indisponibilità all'anno, durante i quali i pazienti possono comunque prenotare visite con un medico sostituto. Per il personale infermieristico è garantito almeno un giorno di riposo obbligatorio a settimana.

Il sistema prevede inoltre l'invio automatico di notifiche via e-mail alla segreteria e ai pazienti non appena gli esiti delle visite diventano disponibili. Per garantire sicurezza, integrità dei dati e conformità, la gestione e l'inserimento degli esiti delle prestazioni sanitarie sono riservati esclusivamente al personale medico e infermieristico.

Processo di sviluppo

Il processo di sviluppo del software è stato articolato in diverse fasi, a partire dall'analisi dei requisiti. La prima fase ha riguardato l'individuazione dei fattori chiave per la realizzazione del progetto, come gli attori coinvolti e le azioni che ciascun attore può compiere. È stato fondamentale comprendere cosa ogni utente fosse in grado di fare all'interno del sistema, definendo così le necessità specifiche per ciascun tipo di utilizzo. Successivamente, sono stati analizzati i comportamenti necessari per realizzare queste azioni, ovvero come ciascun processo dovesse essere eseguito per soddisfare i requisiti.

Nella fase di progettazione, si è passati alla definizione delle strutture di base e, parallelamente, all'implementazione dei vari componenti. Durante la fase di sviluppo, sono stati eseguiti numerosi test per garantire che ogni parte del sistema funzionasse correttamente, individuando eventuali errori. Il controllo continuo durante il ciclo di sviluppo ha permesso di migliorare progressivamente la stabilità del sistema.

Requisiti

Durante la fase di analisi, sono stati identificati i requisiti funzionali e non funzionali necessari per il progetto, assicurandosi che tutte le necessità degli utenti fossero soddisfatte. È stato essenziale definire con precisione le caratteristiche del sistema, garantendo che rispondesse sia agli obiettivi operativi che alle necessità di usabilità.

Progettazione

- **Front-end:** La costruzione dell'interfaccia utente è stata realizzata utilizzando JavaFX, creando un'interfaccia grafica intuitiva e funzionale per l'utente. La progettazione visiva ha privilegiato l'ergonomia, garantendo che le informazioni fossero facilmente accessibili e chiare.
- **Back-end:** Le operazioni sui dati sono state gestite con Java, mentre la persistenza dei dati è stata garantita attraverso l'uso di file di testo, che permettono di memorizzare e recuperare informazioni in modo semplice ed efficace.

Implementazione

L'implementazione ha riguardato diversi aspetti chiave del sistema:

- **Gestione degli utenti:** È stata sviluppata una funzionalità per la gestione dei vari tipi di utenti e delle loro autorizzazioni, in modo che ciascun utente potesse accedere solo alle informazioni pertinenti al proprio ruolo.
- **Gestione delle operazioni:** Ogni operazione è stata progettata per essere eseguita in modo efficiente, garantendo che il sistema risponda correttamente alle richieste in tempi rapidi, minimizzando i rischi di errori.
- **Gestione dei dati:** È stata creata una logica per l'inserimento, l'aggiornamento e la visualizzazione dei dati in modo sicuro e organizzato, rispettando le normative di privacy e sicurezza.
- **Gestione della parte grafica:** Il design e la disposizione degli elementi grafici sono stati curati per offrire un'esperienza utente fluida e semplice da navigare, con particolare attenzione alla reattività e alla fruibilità su dispositivi diversi.

Documentazione

Al termine del progetto, è stata redatta la documentazione finale, che include i diagrammi necessari per comprendere l'evoluzione del sistema e ripercorrere i passi progettuali seguiti. La documentazione offre una panoramica dettagliata dei vari componenti e dei processi, facilitando la comprensione del funzionamento del sistema da parte di eventuali sviluppatori o utilizzatori futuri. È stato inoltre incluso un manuale d'uso per gli utenti finali, che spiega in modo chiaro come interagire con l'interfaccia e sfruttare al meglio le funzionalità del sistema.

Progettazione e Pattern Architetturali

Il sistema è stato progettato adottando metodologie di **modellazione orientata agli oggetti (OOP)**, garantendo una strutturazione modulare e riutilizzabile del codice. A livello architetturale, è stato implementato il **pattern MVC (Model-View-Controller)**, scelta motivata dalla sua naturale integrazione con **FXML**, impiegato per la definizione dell'interfaccia grafica. Questa soluzione è stata selezionata per la sua capacità di separare in modo chiaro e strutturato le tre componenti principali, semplificando lo sviluppo, la manutenzione e l'espansione del sistema.

- **Model**: componente dedicata alla gestione dei dati e alla definizione della loro struttura. Incapsula la logica di accesso e manipolazione delle informazioni, fornendo un'interfaccia coerente per le altre componenti.
- **View**: responsabile della rappresentazione grafica del Model e dell'interazione con l'utente. Supporta aggiornamenti dinamici in base ai cambiamenti dei dati. È stata implementata una View distinta per ogni tipologia di dato da gestire.
- **Controller**: gestisce la logica applicativa e il flusso di controllo, intercettando le azioni dell'utente e aggiornando il Model e la View di conseguenza. Ogni schermata dell'applicazione dispone di un controller dedicato per mantenere un'architettura modulare e scalabile.

Il sistema reagisce agli input dell'utente aggiornando il Model, il cui stato viene poi riflesso nella View per garantire una sincronizzazione costante. Non sono state introdotte ulteriori suddivisioni architetturali, mantenendo Model, View e Controller come blocchi monolitici, data l'efficacia operativa delle loro componenti a questo livello.

Model

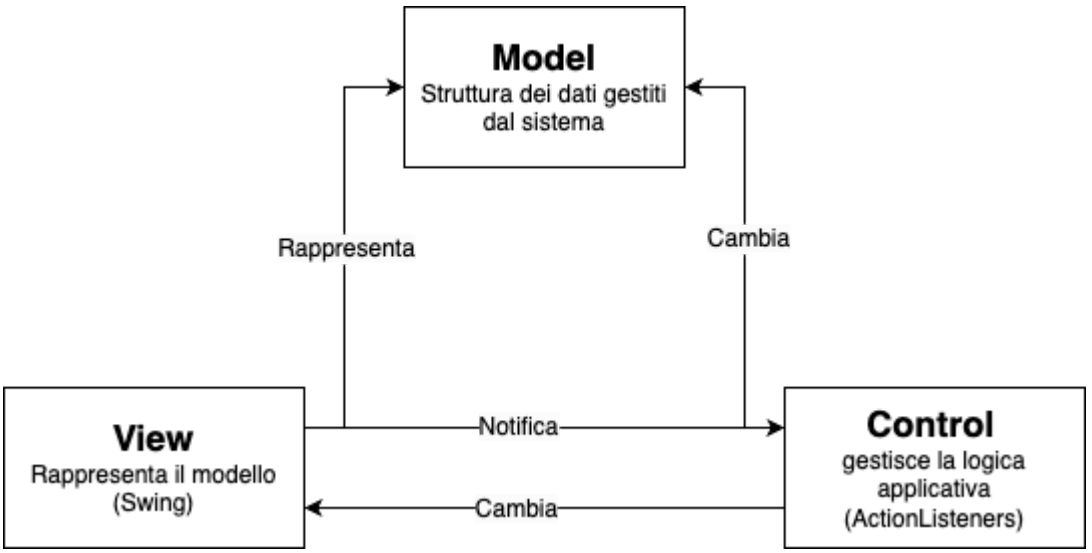
Il **Model** rappresenta il livello logico dell'applicazione e si occupa della persistenza e della coerenza dei dati. Per ogni entità gestita, è stato adottato il **pattern Singleton**, assicurando un'unica istanza per garantire un accesso centralizzato e coerente ai dati.

View

La **View** costituisce l'interfaccia grafica con cui l'utente interagisce. È responsabile della visualizzazione dei dati e della gestione dell'input utente, adattandosi dinamicamente ai cambiamenti del Model.

Controller

Il **Controller** gestisce la logica di interazione dell'applicazione, elaborando gli input dell'utente e inviandoli ai componenti appropriati (Model e View).

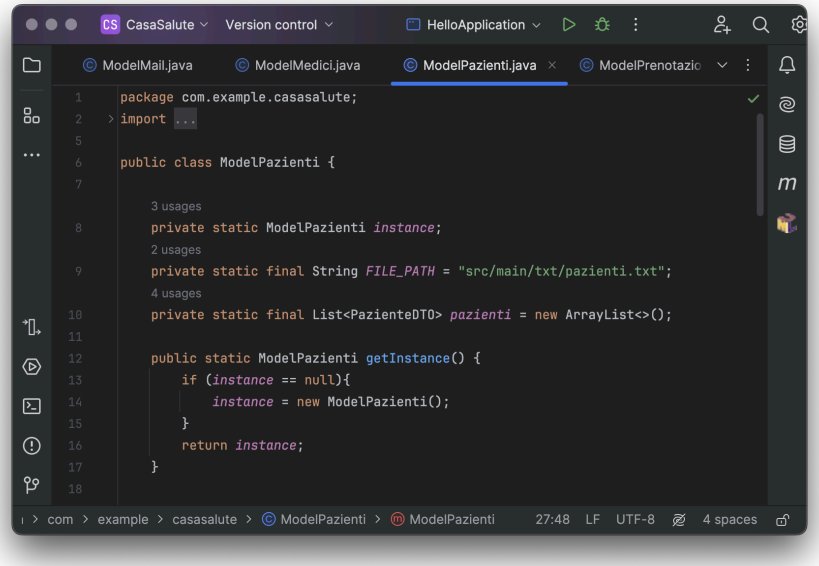


Implementazione e Design Pattern utilizzati

Singleton

Il Singleton Pattern è un design pattern creazionale che garantisce che esista una sola istanza di una classe in tutto il programma e fornisce un metodo globale per accedervi.

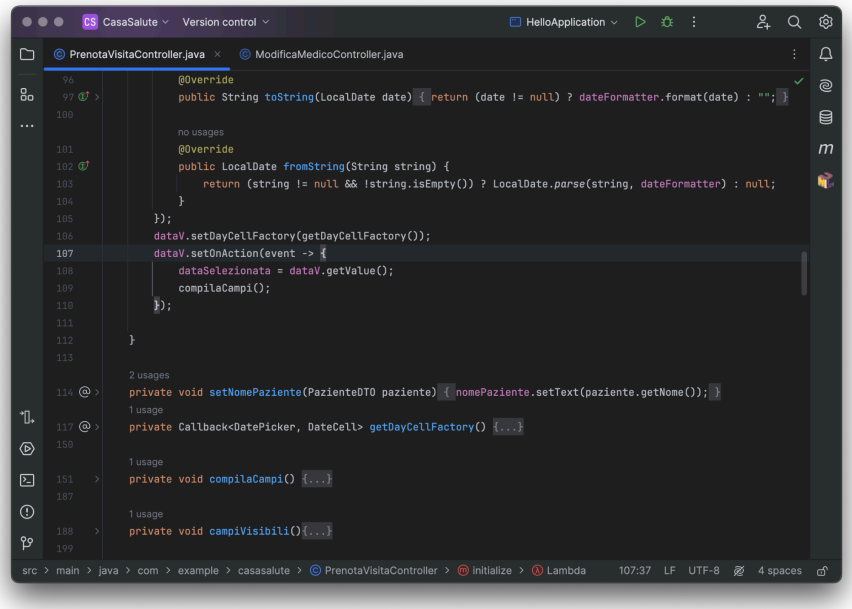
Esempio di utilizzo:
Nel nostro codice, il Singleton viene utilizzato nei model, come “ModelMedici”, “ModelPazienti”, “ModelInfermieri”, ecc. Queste classi gestiscono i dati condivisi all'interno dell'applicazione e il loro stato deve rimanere consistente.



Observer

L'Observer è un pattern che consente a un oggetto principale di notificare automaticamente altri oggetti quando il suo stato cambia. Questo elimina la necessità di controlli continui, permettendo agli elementi collegati di ricevere aggiornamenti in tempo reale e aggiornarsi autonomamente.

Esempio di utilizzo:
Nel nostro codice, l'Observer viene utilizzato in diversi Controller. Un esempio è il "PrenotaVisitaController.java" che gestisce la modifica degli orari disponibili del medico curante in base alla data selezionata.



Iterator

L'Iterator è un pattern comportamentale che permette di scorrere una collezione di oggetti senza esporne la struttura interna.

Esempio di utilizzo:
In questo esempio il codice utilizza un for-each per iterare attraverso "refertiFiltrati" e popolare l'interfaccia con i dati dei referti.

```java
1 usage
private void creaVboxMedico() {
    for (RefertoDTO x : refertiFiltrati) {
        Label labelNomeReferto = new Label(x.getNomeReferto());
        labelNomeReferto.setStyle("-fx-font-family: Arial; -fx-font-size: 14px; -fx-font-weight: bold;");
        Label labelDataReferto = new Label( s: " Data: " + x.getDataReferto());
        labelDataReferto.setStyle("-fx-text-fill: #555;");
        for (PazienteDTO y: pazienti){
            if (y.getCodiceSanitario().equals(x.getCodiceSanitarioPaziente()))
                paziente = y;
        }
        Label labelCSPaziente = new Label( s: " Paziente: " + paziente.getNome() + " " + paziente.getCognome());
        labelCSPaziente.setStyle("-fx-text-fill: #555;");
        VBox refertoBox = new VBox(labelNomeReferto, labelDataReferto, labelCSPaziente);
        refertoBox.setStyle("-fx-background-color: #f9f9f9; -fx-padding: 10; -fx-border-color: #ccc; -fx-border-radiu
        refertoBox.setSpacing(5);
        refertoBox.prefWidthProperty().bind(referti.widthProperty());
        referti.getChildren().add(refertoBox);
        refertoBox.setOnMouseClicked(event -> apriPdf(x.getCodiceReferto()));
    }
}
```

DTO (Data Transfer Object)

Il DTO (Data Transfer Object) è un pattern utilizzato per trasferire dati tra differenti parti dell'applicazione, riducendo il numero di chiamate ai metodi e migliorando l'organizzazione del codice.

Esempio di utilizzo:

Nel nostro progetto, il pattern DTO viene adottato per rappresentare le entità principali come Paziente, Medico, Infermiere, Prenotazione e Referto

```java
public PazienteDTO(String nome, String cognome, String data, String luogo, String email, String password, String
    this.codiceSanitario = generaCodiceSanitario(nome, cognome, data, luogo);
    this.nome = nome;
    this.cognome = cognome;
    this.data = data;
    this.luogo = luogo;
    this.email = email;
    this.password = password;
    this.emailTutore = emailTutore;
    this.medico = medico;
    this.CSTutore = CSTutore;
}
1 usage
    private String generaCodiceSanitario(String nome, String cognome, String data, String luogo) {
    String dataStandardizzata = data.replaceAll( regex: "[^0-9]",  replacement: ""); // Rimuove tutto tranne i numeri
    String dati = nome.toUpperCase() + cognome.toUpperCase() + dataStandardizzata + luogo.toUpperCase();
    int hash = dati.hashCode();
    hash = Math.abs(hash);
    return "CS" + String.valueOf(hash).substring(0, 8); // Prende i primi 8 numeri
    }
```

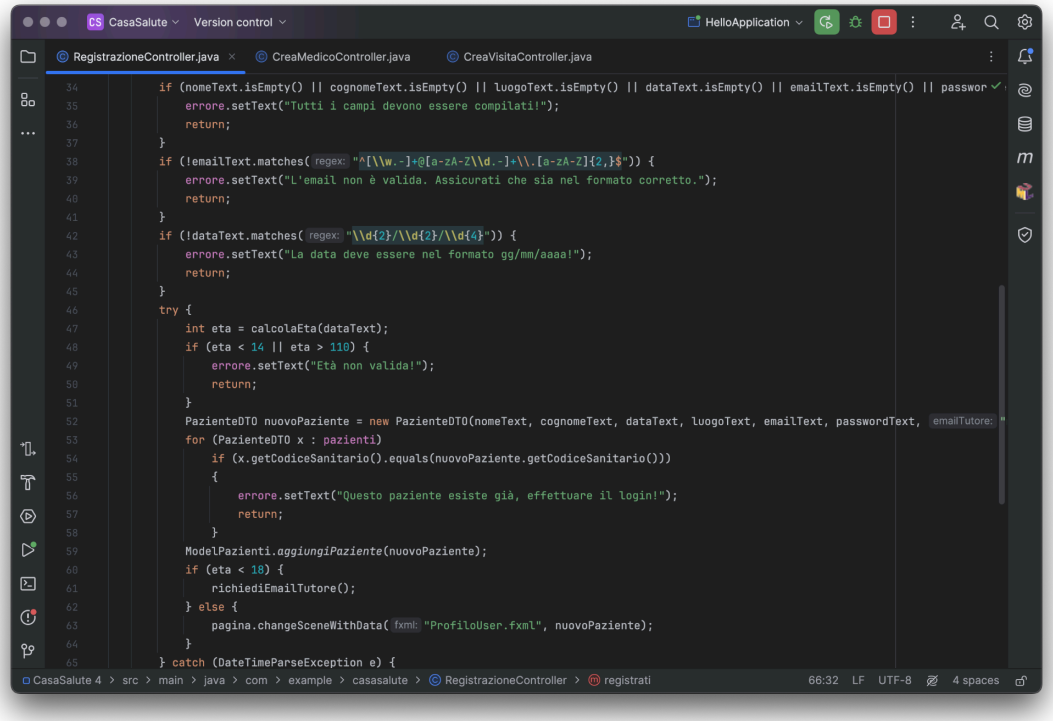
Test e validazione

Per verificare la solidità del progetto, si è scelto di svolgere la fase di test e validazione nelle fasi finali dello sviluppo, coinvolgendo sia gli sviluppatori che utenti generici per testare il software.

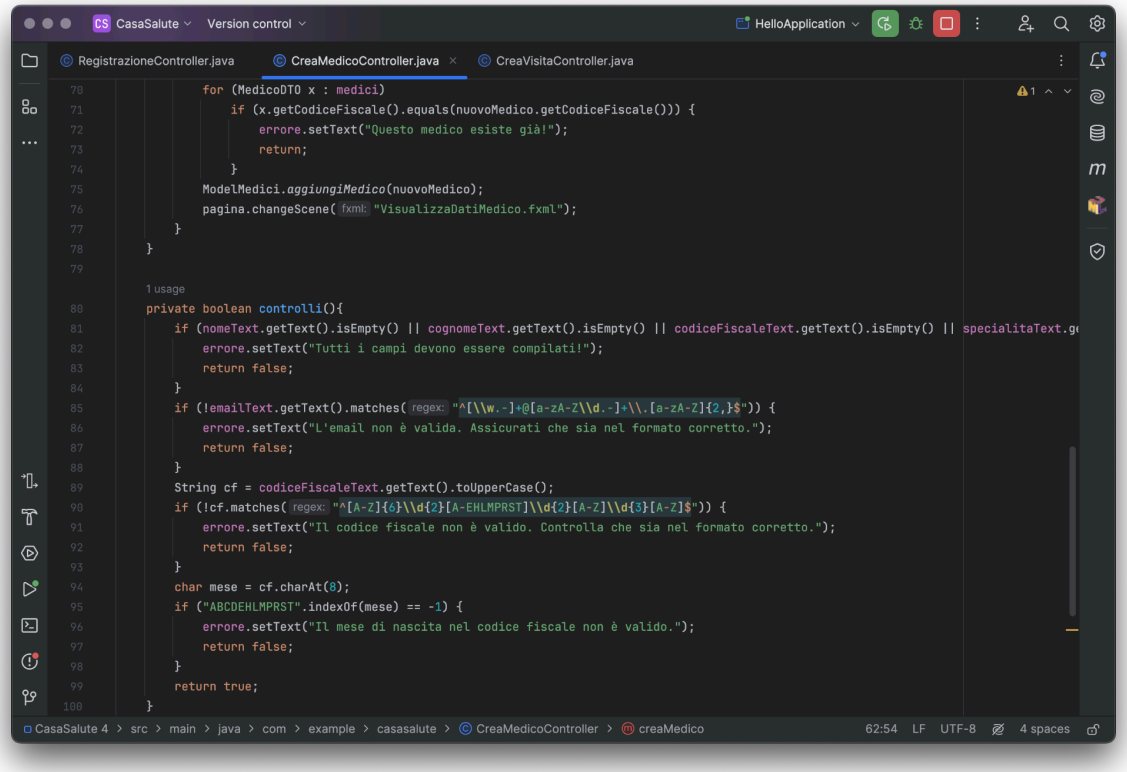
Validazione e Controlli sui Dati

Per garantire il corretto funzionamento del sistema e prevenire errori dovuti all'inserimento di dati non validi, sono stati implementati diversi controlli di validazione nelle varie sezioni dell'applicazione.

Esempio di controlli e regole di validazione per la registrazione:



Esempio di controlli e regole di validazione per la creazione di un nuovo medico:



Test degli sviluppatori

Durante lo sviluppo, sono stati effettuati test manuali per verificare la correttezza del comportamento dell'applicazione. Lo sviluppatore ha inserito sia input validi sia input errati, valutando se il sistema rispondesse in modo adeguato e prevenisse inserimenti non validi.

I principali controlli effettuati riguardano:

- Registrazione utenti: verifica della corretta gestione di campi vuoti, email non valide, formato errato della data di nascita e corretta richiesta della mail del tutore per minori di 18 anni.
- Login e accesso: controllo che email e password siano corrette, impedimento di campi vuoti e gestione di errori per credenziali errate.
- Gestione medici e infermieri: prevenzione di codici fiscali duplicati, controllo della validità delle credenziali, assegnazione corretta di medici sostituti diversi e verifica della presenza di giorni di riposo per gli infermieri.
- Prenotazione visite e prestazioni: verifica della disponibilità dei medici, rispetto del limite massimo di assenze annuali (≤28 giorni), controllo della disponibilità degli ambulatori, impedimento della sovrapposizione di visite pediatriche e blocco della prenotazione di orari già trascorsi nel giorno corrente.

Oltre a questi, sono stati eseguiti ulteriori test per assicurare l'integrità dei dati e la sicurezza del sistema.

Test degli utenti

Terminato lo sviluppo, il programma è stato sottoposto a test da parte di utenti esterni, verificando che il sistema gestisca correttamente le loro operazioni e applicando le validazioni previste.