

**Ospedale D.S.P.D.**

# **RAD – Requirements Analysis Document**

# **Indice**

## **1. Introduzione**

- 1.1. Obiettivo del sistema
- 1.2. Definizioni, acronimi e abbreviazioni

## **2. Sistema corrente**

## **3. Sistema proposto**

- 3.1. Overview
- 3.2. Requisiti funzionali
- 3.3. Requisiti non funzionali
  - 3.3.1. Interfacce grafiche e fattori umani
  - 3.3.2. Documentazione
  - 3.3.3. Considerazioni hardware
  - 3.3.4. Perfomances del sistema
  - 3.3.5. Gestione degli errori
- 3.4. Modelli del sistema
  - 3.4.1. Modelli dei casi d'uso
    - 3.4.1.1. Casi d'uso comuni
    - 3.4.1.2. Caso d'uso aggiuntivi per l'amministratore
    - 3.4.1.3. Caso d'uso Registrazione/Autenticazione
    - 3.4.1.4. Caso d'uso Gestione Visite
    - 3.4.1.5. Caso d'uso Ricovero
    - 3.4.1.6. Caso d'uso Prenotazione
      - 3.4.1.6.1. Caso d'uso esploso Nuova Prenotazione
  - 3.4.2. Modello degli oggetti
  - 3.4.3. Progetto concettuale-descrizione dell'archivio
  - 3.4.4. Modello dinamico
  - 3.4.5. Interfaccia Utente - Mock-up

## **1. Introduzione**

### **1.1. Obiettivo del sistema**

L'obiettivo del software realizzato consiste nel facilitare la gestione di un ospedale, in ogni sua parte e per ogni suo reparto, quindi dalla prenotazione di una visita, con stesura del relativo referto, alla gestione degli eventuali ricoveri.

### **1.2. Definizioni, acronimi e abbreviazioni**

Acronimi:

**ODD** Object Design Document

**RAD** Requirements Analysis Document

**SDD** System Design Document

**DB** Database

**DBMS** Data Base Management System

Sinonimi:

Nre – Codice Ricetta

Iscrizione – Registrazione

## **2. Sistema corrente**

Attualmente, nella maggior parte delle aziende ospedaliere, è presente un sistema informatico atto a organizzare la prenotazione delle visite. Tale servizio viene organizzato esclusivamente dagli operatori, quindi esclude la possibilità ai soggetti non competenti di effettuare o gestire prenotazioni per prestazioni mediche. Inoltre, gli stessi sistemi consentono ai medici di avere accesso al proprio profilo personale solamente dal terminale presente nell'ospedale, vincolando così tale operazione.

## **3. Sistema proposto**

### **3.1. Overview**

La proposta presentata prevede un sistema appositamente realizzato per permettere ai pazienti di effettuare autonomamente la prenotazione di una visita presso un ospedale, mentre ai medici di gestire più facilmente l'organizzazione dell'ambulatorio e del reparto di competenza. Questo prodotto, in definitiva, partendo dalle necessità reali di ogni utente, sulla base del ruolo svolto, permette ad ogni soggetto interessato l'utilizzo delle funzioni prodotte dal sistema.

### **3.2. Requisiti funzionali**

Gli utenti che possono fare uso del sistema ricoprono il ruolo di Paziente, Medico o Amministratore, avendo ognuno la possibilità di accedere a diverse funzionalità del sistema.

Le funzionalità minime offerte dal sistema per il paziente sono:

- **Registrazione/Autenticazione:** Il sistema permette ad una qualunque persona di registrarsi, inserendo i propri dati personali e una password di accesso. Una volta effettuata la registrazione, l'utente può accedere al sistema, usando come credenziali il proprio codice fiscale e la password scelta in fase di registrazione.
- **Prenotazione di una visita:** Il sistema permette ad ogni utente regolarmente registrato, attraverso l'apposito form, di prenotare una visita. Inoltre, la prenotazione è comunque consentita ad utenti che non si sono precedentemente registrati.
- **Visualizzazione dello storico delle proprie prenotazioni:** Ogni utente già registrato ha la possibilità di visualizzare uno storico delle proprie visite effettuate e/o da effettuare.

Le funzionalità minime offerte dal sistema per il medico sono:

- **Gestione delle visite prenotate in ambulatorio:** Ogni utente avente i permessi da medico ha la possibilità di consultare e gestire la lista delle prenotazioni, riferite al proprio turno di lavoro, presso un determinato ambulatorio e in una determinata data. Tale funzionalità è rivolta esclusivamente a chi lavora in ambulatorio.
- **Prenotazione di una visita a nome di un paziente:** Ogni medico ha la possibilità, qualora fosse necessario, di prenotare una visita per conto di un paziente.
- **Effettuare un nuovo ricovero:** Ogni medico può decidere se ricoverare o meno un paziente e, in base alla disponibilità dei posti letto liberi, questi sarà immediatamente ricoverato o inserito nell'apposita lista ricoveri, una per ogni reparto.
- **Gestione delle liste ricoveri:** Ogni medico ha la possibilità di ricoverare un paziente in coda, nel momento in cui si presentino disponibilità di posto letto, oppure di dimettere un paziente. Tale funzionalità è rivolta esclusivamente a chi lavora in reparto.

Ogni medico ha la possibilità, per mezzo dello stesso account, di effettuare la prenotazione di una visita per se stesso e di visualizzare, di conseguenza, lo storico personale.

Le funzionalità minime offerte dal sistema per l'amministratore sono:

- **Gestione dei permessi:** Ogni amministratore, dopo essersi autenticato, ha la possibilità di gestire i permessi associati ad ogni persona registrata. Egli può, in questo, inserire o eliminare un medico dal sistema.
- **Gestione turni:** Il sistema permette ad ogni amministratore, una volta autenticato, di gestire i turni di un reparto o di un ambulatorio.

Ogni amministratore ha la possibilità, per mezzo dello stesso account, di effettuare la prenotazione di una visita per se stesso e di visualizzare, di conseguenza, lo storico personale.

### 3.3. Requisiti non funzionali

#### 3.3.1. Interfacce grafiche e fattori umani

Il software è stato realizzato in modo da poter essere utilizzato da qualsiasi utente in grado di compiere le basilari operazioni sull'uso del computer. In quest'ottica, l'applicazione prevede una grafica semplice e di immediata interpretazione, attraverso la presenza di pulsanti auto esplicativi e indicazioni sui campi da compilare.

#### 3.3.2. Documentazione

Lo sviluppo del progetto viene descritto nei seguenti documenti:

- **RAD:** Documento di analisi dei requisiti
- **SDD:** Documento dell'architettura software
- **ODD:** Documento del progetto esecutivo

#### 3.3.3. Considerazioni hardware

È necessario che il sistema server presenti un hardware in grado di ospitare un DBMS, che gestisca le varie interazioni tra il DB e i vari sistemi terminali, quali quelli utilizzati dai vari utenti, ovvero pazienti, medici e amministratori.

#### 3.3.4. Performances del sistema

Risulta necessario utilizzare componenti più performanti per il server, al fine di ottimizzare tutte le operazioni effettuate dal server stesso, quali: l'invio delle E-Mail, l'invio degli SMS e le operazioni da effettuare sul DB. Per quanto

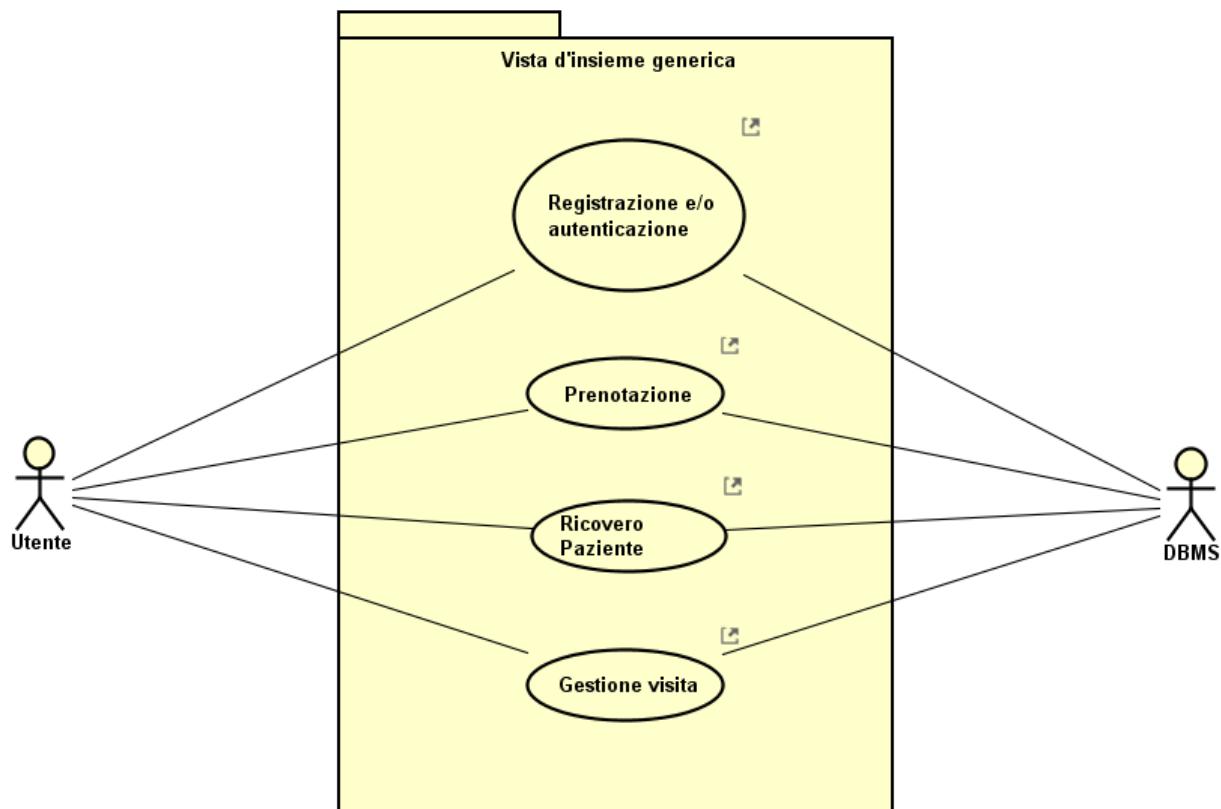
riguarda l'hardware richiesto dal software per il suo utilizzo da parte degli utenti, si richiede un'implementazione minimale.

### 3.3.5. Gestione degli errori

In caso di errore o caduta di connessione, il sistema prevede, attraverso una finestra di dialogo, la segnalazione esplicita dell'errore stesso, suggerendo eventuali operazioni che ne possano prevenire una nuova manifestazione.

## 3.4. Modelli del sistema

### 3.4.1. Modello dei casi d'uso



### 3.4.1.1. Casi d'uso comuni

CASO D'USO: CADUTA CONNESSIONE
<b>ATTORI:</b> DBMS
<b>PRECONDIZIONE:</b> Il sistema non riesce a connettersi al DMBS
<b>SEQUENZA DEGLI EVENTI:</b> <ol style="list-style-type: none"><li>1. Il caso d'uso inizia quando il DBMS non risponde all'interazione con il sistema.</li><li>2. Il sistema mostra a video la finestra "Caduta Connessione" con un messaggio di assenza di connessione e un tasto "Connettiti".</li><li>3. L'utente preme il tasto "Connettiti" e il tasto si disabilita.</li><li>4. Se il sistema riesce a connettersi al DBMS, il sistema chiude la finestra, altrimenti il tasto torna attivo.</li></ol>
<b>POSTCONDIZIONE:</b> Il sistema riprende dal punto in cui si era interrotto.

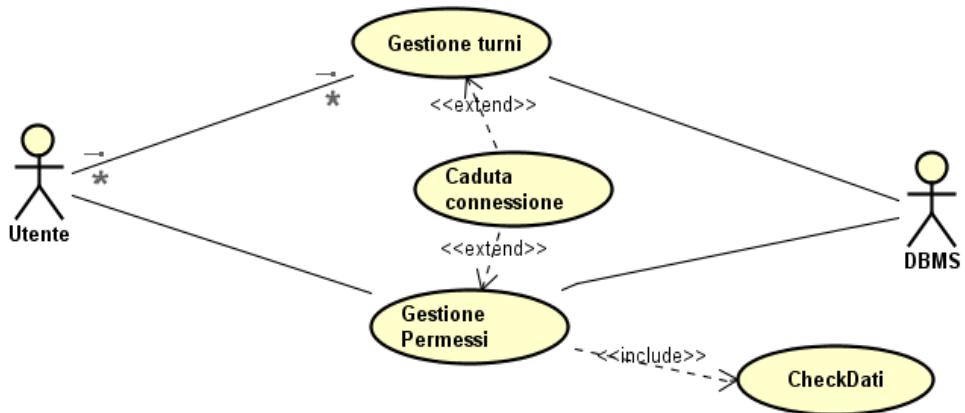
CASO D'USO: CHECK DATI
<b>ATTORI:</b> Utente, DBMS
<b>PRECONDIZIONE:</b> L'utente deve essere autenticato
<b>SEQUENZA DEGLI EVENTI:</b> <ol style="list-style-type: none"><li>1. Il caso d'uso inizia quando l'utente, dopo aver compilato il campo del codice fiscale clicca sul tasto "CHECK"</li><li>2. Il sistema legge il codice fiscale inserito e lo manda al DBMS<ol style="list-style-type: none"><li>2.1 Se il codice fiscale inserito non è corretto appare un messaggio di errore e vengono abilitati i campi per l'inserimento dei dati.</li><li>2.2 Altrimenti il sistema compila automaticamente gli appositi campi</li></ol></li></ol>

**PUNTO DI ESTENSIONE:**

Nel punto 2, se cade la comunicazione il sistema invoca il caso d'uso "CadutaConnessione"

**POSTCONDIZIONE:**

Continua il caso d'uso che lo ha invocato

**3.4.1.2. Caso d'uso aggiuntivi per l'amministratore****CASO D'USO: GESTIONE TURNI****ATTORI:**

Utente, DBMS

**PRECONDIZIONE:**

L'utente è autenticato come amministratore

**SEQUENZA DEGLI EVENTI:**

1. L'utente preme il bottone "Modifica Turni"
2. Il sistema mostra a video una schermata formata da una casella dove selezionare il reparto o l'ambulatorio di cui si vogliono gestire i turni e dai rispettivi giorni della settimana.
3. Cliccando su un giorno viene mostrato a video il nome dei medici di competenza.
4. L'utente sceglie il medico per il determinato giorno della settimana.
5. L'utente preme il bottone Conferma quando ha effettuato le modifiche necessarie.
6. Il sistema contatta il DBMS per aggiornare il turno di quell'utente.
7. Il sistema mostra un messaggio di conferma

**PUNTO DI ESTENSIONE:**

Nel punto 6, nel caso in cui ci sia Caduta di Connessione, si rimanda al caso d'uso "Caduta Connessione"

**POSTCONDIZIONE:**

L'utente ha modificato i turni del reparto selezionato.

**CASO D'USO: GESTIONE PERMESSI****ATTORI:**

Utente, DBMS

**PRECONDIZIONE:**

L'utente è autenticato come amministratore

**SEQUENZA DEGLI EVENTI:**

1. Il caso d'uso inizia quando l'utente preme il bottone "Gestione Permessi".
2. Viene mostrata una schermata formata dai seguenti campi: "Codice fiscale, Nome, cognome, email, telefono, indirizzo, città, sesso, Data di nascita, gruppo permesso e specializzazione"
3. L'utente, autenticato come amministratore, inserisce il codice fiscale dell'utente di cui vuole gestire i permessi.
4. L'utente preme il pulsante "Check" e invoca il caso d'uso "Check Dati"
  - 4.1. Se l'utente è un paziente, il campo Specializzazione resta vuoto.
  - 4.1.2. Altrimenti, se il codice fiscale inserito è quello di un utente con i permessi da medico, il campo specializzazione viene riempito con la qualifica del medico
4. L'utente sceglie tra i seguenti permessi: "Paziente, medico e amministratore".
5. L'utente preme il bottone "Modifica dati".
6. Il sistema contatta il DBMS per aggiornare il permesso di quell'utente
7. Il sistema mostra un messaggio di conferma

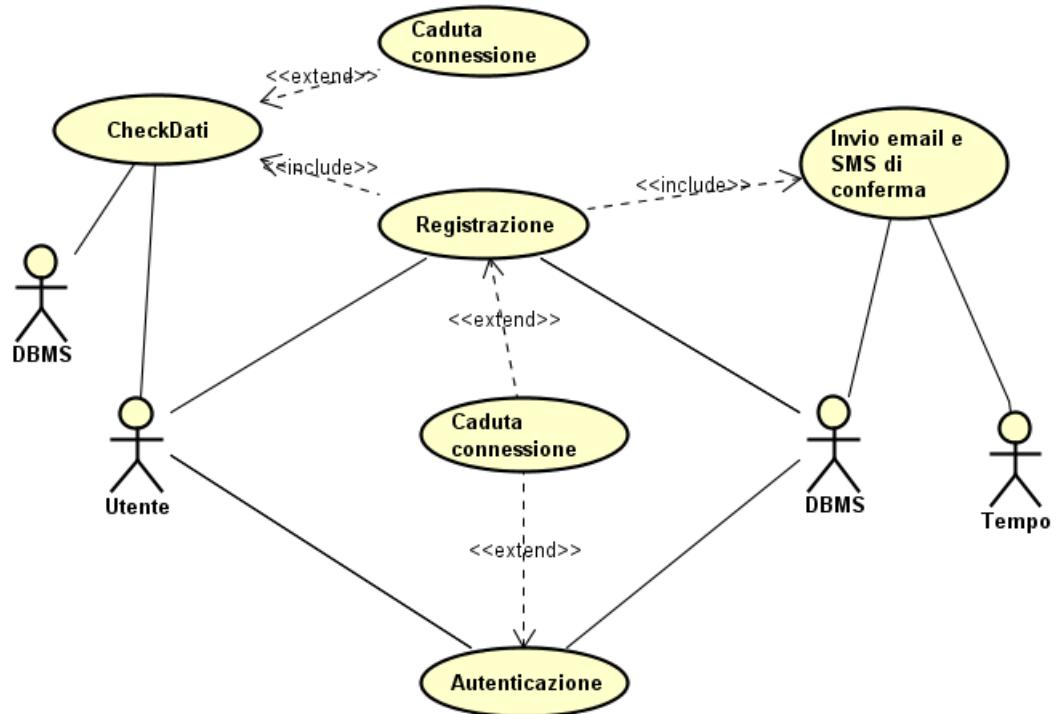
**PUNTO DI ESTENSIONE:**

Nel punto 6 , in caso di caduta di connessione , si rimanda al caso d'uso Caduta di Connessione.

#### POSTCONDIZIONE:

L'utente loggato come amministratore ha modificato il permesso dell'utente inserito.

#### 3.4.1.3. Caso d'uso Registrazione/Autenticazione



#### CASO D'USO: REGISTRAZIONE

##### ATTORI:

Utente, DBMS

##### PRECONDIZIONE:

Il sistema mostra la pagina principale

**SEQUENZA DEGLI EVENTI:**

1. Il caso d'uso inizia quando l'utente preme sul pulsante "Registrati"
2. Il sistema mostra un form di registrazione.
3. L'utente dopo avere inserito il CF nell'apposito campo preme sul tasto "Check" e inizia il caso d'uso check dati
4. L'utente, dopo aver controllato i campi ed aver inserito la propria password, preme sul pulsante "Conferma".
  - 4.1. Il sistema verifica se i dati rispettano i vari campi del form.
  - 4.2. Se i dati non sono rispettati i campi relativi a tali dati vengono evidenziati in rosso e l'utente deve reinserire i dati.
5. Il sistema comunica al DBMS i dati dell'utente.
6. Viene confermata la creazione dell'account con un messaggio.
7. Viene chiamato il caso d'uso "Invia email e SMS di conferma Registrazione"

**PUNTO DI ESTENSIONE:**

Nel punto 5 in caso di caduta di connessione richiama il caso d'uso "Caduta Connessione"

**POSTCONDIZIONE:**

Il sistema torna nella pagina principale.

**CASO D'USO: AUTENTICAZIONE****ATTORI:**

Utente, DBMS

**PRECONDIZIONE:**

Il sistema mostra la schermata principale

**SEQUENZA DEGLI EVENTI:**

1. Il caso d'uso inizia quando l'utente preme sul pulsante "Accedi"
2. Il sistema mostra a video un form dove chiede di inserire CODICE FISCALE e PASSWORD
3. L'utente inserisce i dati e clicca sul pulsante "Log in"
4. Il sistema verifica la correttezza dei dati.
  - 4.1. Se i dati inseriti non sono corretti, il campo con il dato non valido diventa rosso e viene visualizzato un messaggio di errore.
5. Il sistema invia al DBMS
  - 5.1. Se i dati sono corretti e l'utente ha i permessi da "Paziente" viene visualizzata una schermata con i tasti "Prenota" e "Gestione Visite".

- 5.2. Altrimenti, se l'utente ha i permessi da "Medico" è presente inoltre il tasto "Nuovo Ricovero".
- 5.3. Altrimenti, se l'utente ha i permessi da "Amministratore" è presente solamente il tasto "Gestione Utenti".

**PUNTO DI ESTENSIONE:**

Nel punto 5, se cade la comunicazione il sistema invoca il caso d'uso "CadutaConnessione"

**CASO D'USO: INVIO E-MAIL E SMS DI CONFERMA REGISTRAZIONE****ATTORI:**

Utente

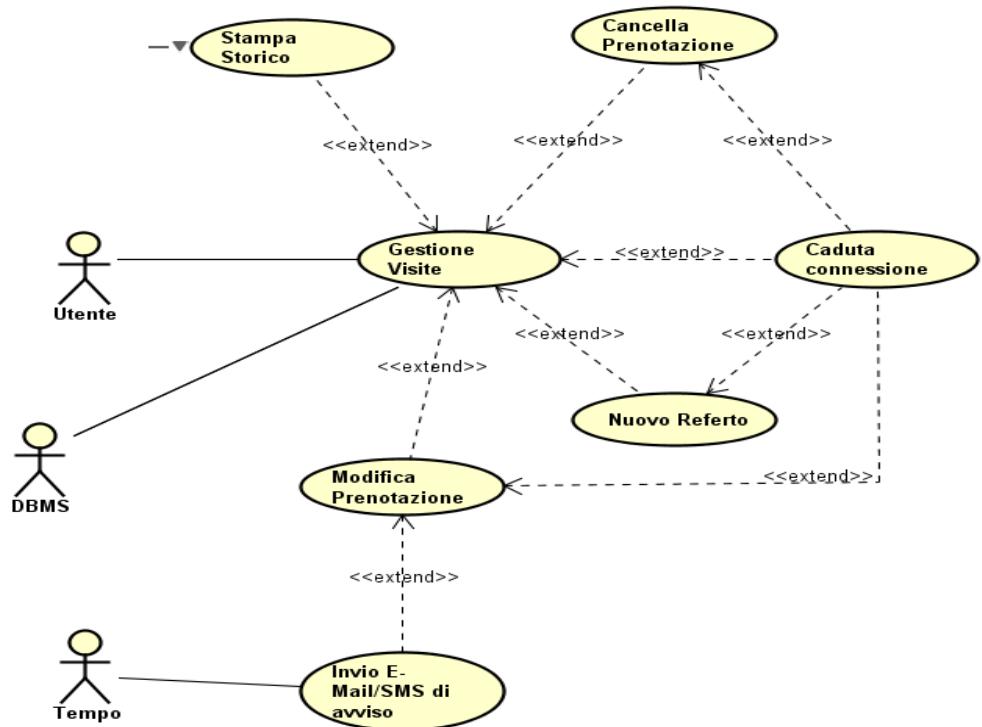
**PRECONDIZIONE:**

L'utente si è appena registrato

**SEQUENZA DEGLI EVENTI:**

1. Il caso si avvia ogni 30 secondi.
2. Il sistema contatta il DBMS per verificare se ci sono utenti registrati a cui non è stata invia la mail di registrazione.
3. Se il DBMS ottiene dei risultati, il sistema invia una e-mail e un SMS all'utente che si è appena registrato.

### 3.4.1.4. Caso d'uso Gestione Visite



CASO D'USO: GESTIONE VISITE	
<b>ATTORI:</b>	
DBMS, utente	
<b>PRECONDIZIONE:</b>	L'utente è autenticato
<b>SEQUENZA DEGLI EVENTI:</b>	<ol style="list-style-type: none"> <li>Il caso d'uso inizia quando l'utente non è un medico e preme il pulsante "Gestione Visite".</li> <li>Il sistema chiede al DBMS le prenotazioni effettuate dall'utente.</li> <li>Viene visualizzata una tabella composta da un numero di righe pari al numero di prenotazioni effettuate dall'utente e dalle seguenti colonne: NRE, Tipo Visita, Data, Ora Inizio, Ora Fine, Ambulatorio, Reparto, Medico.</li> </ol>

**SEQUENZA ALTERNATIVA 1:**

1. Il caso d'uso inizia quando l'utente ha i permessi da medico e preme il pulsante "Le mie visite":
2. Il sistema chiede al DBMS le visite relative a quel medico.
3. Viene visualizzata una tabella composta da un numero di righe pari al numero di prenotazioni effettuate dall'utente e dalle seguenti colonne: NRE, Tipo Visita, Data, Ora Inizio, Ora Fine, Ambulatorio, Reparto, Medico.

**SEQUENZA ALTERNATIVA 2:**

1. Il caso d'uso inizia quando l'utente ha i permessi da medico e preme il pulsante "Le visite dei miei pazienti":
2. Il sistema chiede al DBMS le visite relative a quel medico.
  - 2.1. Il sistema chiede al DBMS le visite effettuate dall'utente ad altri pazienti
3. Viene visualizzata una tabella composta da un numero di righe pari al numero di visite che l'utente ha effettuato o deve effettuare e dalle seguenti colonne: Codice Fiscale, Paziente, NRE, Tipo Ricetta, Data, Ora Inizio, Ora Fine, Ambulatorio.

**PUNTO DI ESTENSIONE:**

È presente il tasto "Stampa" che, se premuto, apre il caso d'uso "Stampa Storico".

Nel punto 2.1 se cade la connessione chiama il caso d'uso Caduta Connessione

**POSTCONDIZIONE:**

Viene mostrato a video l'elenco delle prenotazioni.

**NOTE:**

Facendo doppio click su una prenotazione non ancora completata (nel senso che la visita non è stata ancora fatta) è possibile modificarla e/o cancellarla.

Cliccando sulle prenotazioni completate è possibile visualizzarne il referto.

Se l'utente ha i permessi da "Medico", cliccando su una visita effettuata da lui, se il referto non è stato ancora fatto, viene mostrata una schermata per compilare il referto.

## CASO D'USO: NUOVO REFERTO

### ATTORI:

Utente, DBMS

### PRECONDIZIONE:

L'utente deve essere autenticato come medico.

### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso inizia quando l'utente ha selezionato una visita dalla lista della pagina "Ambulatorio" in "Gestione Visite".
2. Se la visita è prenotata per un giorno precedente al giorno corrente, viene mostrato a video una schermata in cui è possibile scrivere il referto.
3. L'utente compila il form che viene visualizzato a video.
4. L'utente preme il tasto "Conferma".
5. Il sistema invia al DBMS il referto.
6. Il sistema mostra a video un messaggio di riuscita.

### PUNTO DI ESTENSIONE:

Nel punto 5 se vi è caduta di connessione, il sistema invoca il caso d'uso "Caduta Connessione".

### POSTCONDIZIONE:

Il sistema mostra a video la schermata iniziale.

## CASO D'USO: CANCELLA PRENOTAZIONE

### ATTORI:

Utente, DBMS

### PRECONDIZIONE:

L'utente ha selezionato dallo storico una visita non ancora effettuata.

### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso inizia quando l'utente fa doppio click su una visita non ancora effettuata e dalla finestra preme il tasto Cancell.
2. Il sistema chiede al DBMS di cancellare la prenotazione.
3. Il sistema stampa a video un messaggio di avvenuta cancellazione.

### PUNTO DI ESTENSIONE:

Nel punto 2. In caso di caduta connessione si rimanda al caso d'uso Caduta Connessione

### POSTCONDIZIONE:

Il sistema torna alla schermata "Gestione Visite".

## CASO D'USO: STAMPA STORICO

### ATTORI:

DBMS, utente

### PRECONDIZIONE:

L'utente deve aver effettuato almeno una prenotazione.

### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso inizia quando l'utente preme il pulsante "Stampa".
2. Il sistema stampa la tabella generata al punto 3 nel caso d'uso "Gestione visite".

### POSTCONDIZIONE:

L'utente riceve la stampa dello storico.

## CASO D'USO: MODIFICA PRENOTAZIONE

### ATTORI:

Utente, DBMS

### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso inizia quando l'utente fa doppio click su una visita non ancora effettuata e dalla finestra preme il tasto Modifica
2. Se l'utente sta modificando la propria prenotazione, il sistema mostra il calendario.
  - 2.1. Altrimenti, il sistema mostra una finestra dove scrivere il motivo della modifica della data e successivamente si apre il calendario.
3. Dopo che l'utente ha cambiato la data e l'ora, preme il tasto "Salva".
4. Il sistema invia i dati aggiornati al DBMS.
5. Il sistema mette a video un messaggio di modifica effettuata.

### POSTCONDIZIONE:

Il sistema torna alla schermata "Gestione Visite".

Se il punto 2.1. è andato a buon fine allora viene invocato il caso d'uso "Invio E-Mail/SMS di avviso".

**NOTE:** La pagina di gestione della prenotazione si apre facendo doppio click sulla prenotazione dopo averla selezionata dal "Gestione visite".

## CASO D'USO: INVIO E-MAIL/SMS DI AVVISO

### ATTORI:

Tempo, DBMS

### PRECONDIZIONE:

Una prenotazione è stata spostata dal sistema o da un utente autenticato come medico.

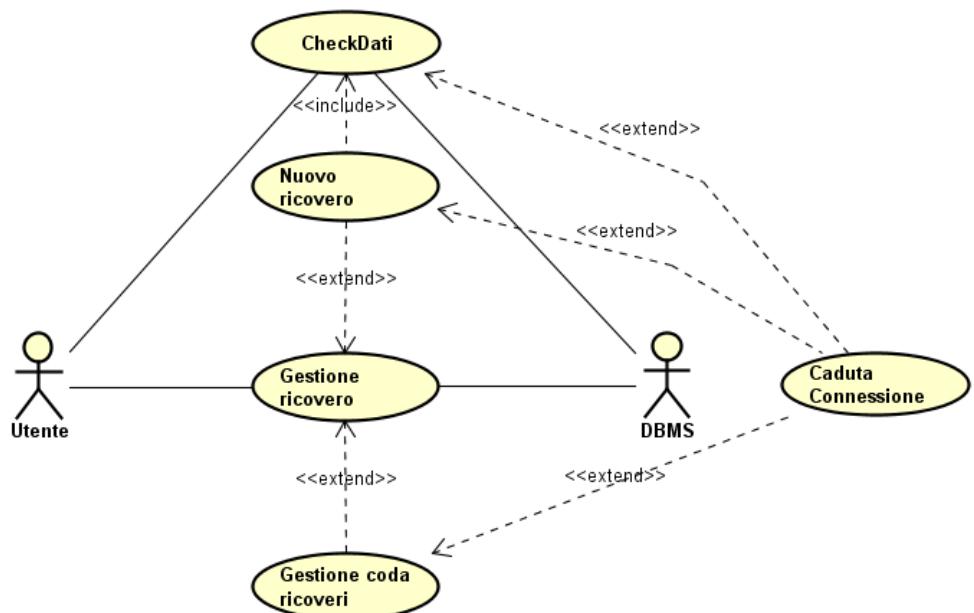
### SEQUENZA DEGLI EVENTI:

1. Il caso si avvia ogni 30 secondi.
2. Il sistema contatta il DBMS per verificare se ci sono nuove prenotazioni con un avviso non ancora inviato.
3. Se il DBMS ottiene dei risultati, il sistema invia al paziente una e-mail e un SMS contenente:

3.1. Nuovo data e ora della prenotazione.

3.2. Motivo per il quale la prenotazione è stata modificata.

### 3.4.1.5. Caso d'uso Ricovero



#### CASO D'USO: NUOVO RICOVERO

##### ATTORI:

Utente, DBMS

##### PRECONDIZIONE:

L'utente deve avere i permessi da medico.

##### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso inizia quando l'utente preme il pulsante "Nuovo ricovero" dalla schermata di Gestione Ricovero.
2. L'utente inserisce il codice fiscale dell'utente e preme su check, invocando il caso d'uso "CheckDati"
3. L'utente preme su "ConfermaDati"

4. Il Sistema contatta il DBMS chiedendo di controllare la disponibilità dei letti per il reparto in cui lavora l'utente.
5. Se ci sono posti letti disponibili viene mostrato a video il bottone "Ricovera".
  - 5.1. L'utente preme il tasto "Ricovera"
  - 5.2. Il sistema manda al DBMS i dati del paziente e la data di occupazione del posto letto.
6. Se non ci sono posti letti disponibili viene mostrato a video il bottone "Metti in coda".
  - 6.1. L'utente preme il tasto "Metti in coda"
  - 6.2. Il sistema manda al DBMS i dati del paziente.

**PUNTO DI ESTENSIONE:**

Nel punto 4,5.2,6.2 se vi è caduta di connessione, il sistema invoca il caso d'uso "Caduta Connessione".

**POSTCONDIZIONE:**

Si viene rimandati alla pagina principale.

## CASO D'USO: GESTIONE CODA RICOVERI

### ATTORI:

Utente, DBMS

### PRECONDIZIONE:

- 1.L'Utente deve essere autenticato
- 2.L'Utente è autenticato come medico

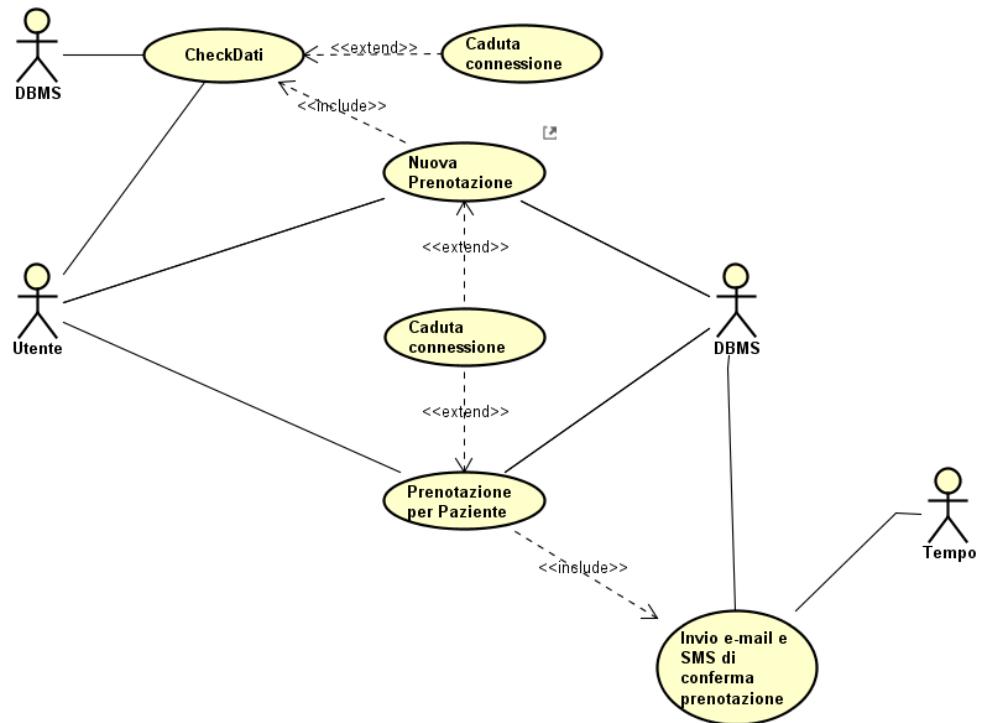
### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso inizia quando l'utente preme sul tasto "Gestione Coda".
2. Il sistema mostra a video una schermata con due liste, una dei pazienti ricoverati ed una dei pazienti in coda.
3. Il sistema contatta il DBMS chiedendo di controllare la disponibilità dei letti per il reparto scelto dall'utente.
  - 3.1. Se ci sono posti letto disponibili, il sistema abilita il tasto "Ricovera".
4. L'utente può selezionare un paziente da una di queste due liste
  - 4.1. Se l'utente seleziona un paziente dalla lista dei pazienti ricoverati:
    - 4.1.1. Il sistema abilita il tasto "DIMETTI PAZIENTE"
    - 4.1.2. L'utente preme sul tasto "DIMETTI PAZIENTE"
    - 4.1.3. Il sistema contatta il DBMS per eliminare il paziente dalla lista dei pazienti ricoverati.
    - 4.1.4. Una volta ricevuta risposta dal DBMS il sistema elimina dalla lista dei pazienti ricoverati il paziente selezionato
  - 4.2. Se l'utente seleziona un paziente dalla lista dei pazienti in coda:
    - 4.2.1. L'utente preme sul tasto "RICOVERA"
    - 4.2.2. Il sistema contatta il DBMS per eliminare dalla coda dei pazienti il paziente selezionato e aggiungerlo alla lista dei pazienti ricoverati
    - 4.2.3. Il sistema una volta ricevuta risposta dal DBMS elimina dalla lista dei pazienti in coda il paziente selezionato e lo aggiunge alla lista dei pazienti ricoverati

## PUNTO DI ESTENSIONE:

Nei punti 3, 4.1.3, 4.1.4. e 4.2.2, se cade la comunicazione il sistema invoca il caso d'uso "CadutaConnessione"

### 3.4.1.6. Caso d'uso Prenotazione



## CASO D'USO: NUOVA PRENOTAZIONE

### ATTORI:

Utente, DBMS

### PRECONDIZIONE:

L'utente deve essere autenticato o essere in modalità ospite.

### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso inizia quando l'utente clicca sul pulsante "Prenota".
- 1.1. Se l'utente non è autenticato si va alla sequenza alternativa 1.
2. Il sistema mostra a video un form e riempie automaticamente i campi relativi ai dati anagrafici del paziente.
3. L'utente, dopo aver inserito il codice NRE preme sul tasto "Conferma".

4. Il sistema si connette al DBMS per verificare l'esistenza della ricetta.
  - 4.1. Se la ricetta non esiste il sistema genera una finestra di errore.
  - 4.2. Altrimenti viene abilitato il campo per inserire l'ambulatorio.
5. L'utente preme su Prenota
6. Il sistema chiama il caso d'uso "Controllo prerequisiti prenotazione".
7. Il sistema legge dalla ricetta l'urgenza della visita e apre il calendario.
8. Se la visita ha priorità U:
  - 8.1. Il sistema mostra in verde le date fino a 2 giorni dopo l'emissione della ricetta.
9. Se la visita ha priorità P:
  - 9.1. Il sistema mostra in verde le date fino a 3 mesi dopo l'emissione della ricetta.
10. Se la visita ha priorità B:
  - 10.1. Il sistema mostra in verde le date fino a 10 giorni dopo l'emissione della ricetta.
11. Se la visita ha priorità D:
  - 11.1. Il sistema mostra in verde le date fino a 60 giorni dopo l'emissione della ricetta.
12. Viene chiamato il caso d'uso "Seleziona Data".
13. Viene chiamato il caso "Invia Email e SMS di conferma prenotazione".

#### **SEQUENZA ALTERNATIVA:**

1. Il sistema mostra a video un form dove vanno inseriti i dati dell'utente e il codice NRE
2. L'utente dopo aver inserito i dati preme sul tasto "Mostra Calendario".
3. Il sistema si connette al DBMS per verificare l'esistenza della ricetta.
  - 3.1. Se la ricetta non esiste il sistema genera una finestra di errore.
4. Il sistema chiama il caso d'uso "Controllo prerequisiti prenotazione".
5. Il sistema legge l'urgenza della visita.
6. Se la visita ha priorità U:
  - 6.1. Il sistema mostra in verde le date fino a 2 giorni dopo la richiesta del medico.
7. Se la visita ha priorità P:
  - 7.1. Il sistema mostra in verde le date fino a 3 mesi dopo la richiesta del medico.
8. Se la visita ha priorità B:

- 8.1. Il sistema mostra in verde le date fino a 10 giorni dopo la richiesta del medico.
- 9. Se la visita ha priorità D:
  - 9.1. Il sistema mostra in verde le date fino a 60 giorni dopo la richiesta del medico.

**PUNTO DI ESTENSIONE:**

Nel punto 4 della sequenza degli eventi o 3 della sequenza alternativa, se cade la comunicazione col DBMS il sistema invoca il caso d'uso “CadutaConnessione”.

**POSTCONDIZIONE:**

Il sistema mostra il calendario rendendo disponibili gli orari in base a: urgenza e ambulatorio scelto.

**CASO D'USO: PRENOTA PER PAZIENTE**

**ATTORI:**

Utente, DBMS

**PRECONDIZIONE:**

L'utente deve avere i permessi da medico.

**SEQUENZA DEGLI EVENTI:**

1. Il caso d'uso inizia quando l'utente preme il tasto “Prenota per paziente”.
2. Il sistema mostra a video un form dove vanno inseriti i dati del paziente e il codice NRE
3. L'utente dopo aver inserito i dati preme sul tasto “Conferma”
4. Il sistema si connette al DBMS per avere il reparto della ricetta
5. Il sistema abilita il campo per inserire l'ambulatorio.
6. L'utente preme su Prenota.
7. Il sistema chiama il caso d'uso “Controllo prerequisiti prenotazione”.
8. Il sistema legge l'urgenza della visita e apre il calendario
9. Se la visita ha priorità U:
  - 9.1. Il sistema mostra disponibili le date fino a 2 giorni dopo la richiesta del medico.

10. Se la visita ha priorità P:

10.1. Il sistema mostra disponibili le date fino a 3 mesi dopo la richiesta del medico.

11. Se la visita ha priorità B:

11.1. Il sistema mostra disponibili le date fino a 10 giorni dopo la richiesta del medico.

12. Se la visita ha priorità D:

12.1. Il sistema mostra disponibili le date fino a 60 giorni dopo la richiesta del medico.

13. Viene chiamato il caso d'uso Selezione Data ultimo il quale viene invocato il caso d'uso "Invia Email e SMS di conferma prenotazione"

#### **PUNTO DI ESTENSIONE:**

Nel punto 4, se cade la comunicazione col DBMS il sistema invoca il caso d'uso "CadutaConnessione".

#### **POSTCONDIZIONE:**

Il sistema mostra il calendario in base al tipo di visita e all'urgenza.

### **CASO D'USO: INVIO E-MAIL E SMS DI CONFERMA PRENOTAZIONE**

#### **ATTORI:**

Utente

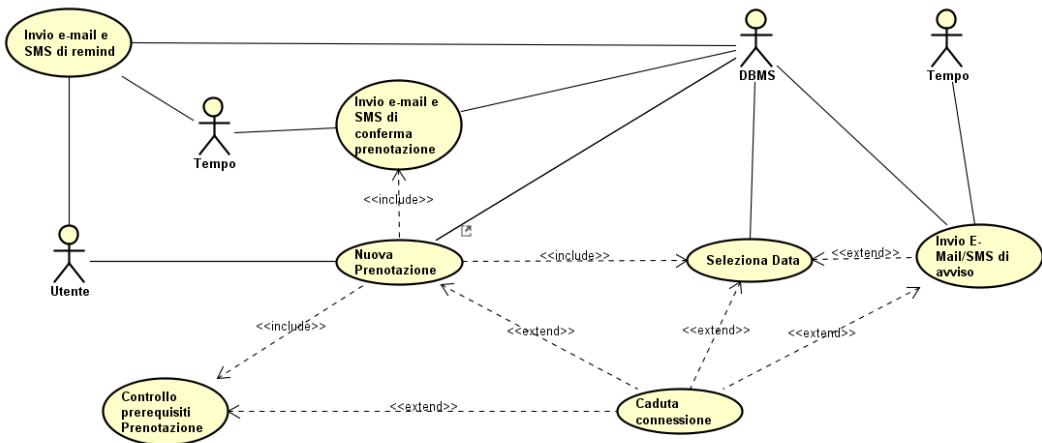
#### **PRECONDIZIONE:**

L'utente ha appena effettuato una prenotazione.

#### **SEQUENZA DEGLI EVENTI:**

1. Il caso si avvia ogni 30 secondi.
2. Il sistema contatta il DBMS per verificare se ci sono nuove prenotazioni.
3. Se il DBMS ottiene dei risultati, il sistema invia una e-mail e un SMS al paziente che ha appena effettuato una prenotazione contenente:
  - 3.3. Data e ora della prenotazione.
  - 3.4. Nome dell'ambulatorio e del reparto in cui avverrà la visita

### 3.4.1.6.1. Caso d'uso esploso Nuova Prenotazione



#### CASO D'USO: INVIO E-MAIL E SMS DI REMIND

##### ATTORI:

Tempo, DBMS

##### PRECONDIZIONE:

Almeno un utente ha una visita tra meno di 24 ore

##### SEQUENZA DEGLI EVENTI:

1. Il caso d'uso si avvia automaticamente ogni 30 secondi
2. Il sistema contatta il DBMS per estrarre le ricette che avranno luogo tra 24 ore con le relative e-mail e numero di telefono del paziente.
3. Il sistema invia un SMS e una e-mail contenente le seguenti informazioni:
  - 3.1. Data e ora della prenotazione.
  - 3.2. Nome dell'ambulatorio in cui avverrà la visita.
  - 3.3. Eventuali documenti da portare alla visita.

<b>CASO D'USO: CONTROLLO PREREQUISITI PRENOTAZIONE</b>
<b>ATTORI:</b>
DBMS, Utente
<b>SEQUENZA DEGLI EVENTI:</b>
<ol style="list-style-type: none"> <li>1. Il sistema contatta il DBMS per vedere se per la visita da prenotare serve avere della documentazione aggiuntiva.</li> <li>2. Il DBMS comunica al sistema la documentazione che serve.</li> <li>3. Se sono necessari dei documenti, il sistema mostra a video una finestra con l'elenco dei documenti</li> <li>4. Se l'utente conferma di averli, preme sul tasto SI e prosegue il caso d'uso.</li> <li>5. Se l'utente preme sul pulsante NO, non sarà possibile prenotare</li> </ol>
<b>PUNTO DI ESTENSIONE:</b>
Nel punto 2, se cade la comunicazione col DBMS il sistema invoca il caso d'uso "CadutaConnessione".

<b>CASO D'USO: SELEZIONA DATA</b>
<b>ATTORI:</b>
Utente, DBMS
<b>PRECONDIZIONE:</b>
Il sistema mostra a video il calendario.
<b>SEQUENZA DEGLI EVENTI:</b>
<ol style="list-style-type: none"> <li>1. Il caso d'uso inizia quando l'utente sceglie una data disponibile tra quelle presenti.</li> <li>2. Il sistema contatta il DBMS per controllare gli orari disponibili per quella data.</li> <li>3. Il sistema mostra gli orari disponibili per la data selezionata.</li> <li>4. Gli orari delle visite già prenotate con priorità inferiore non vengono mostrate occupate.</li> <li>5. L'utente seleziona un orario e preme sul pulsante "Salva"</li> <li>6. Il sistema contatta il DBMS per verificare se per quella data e ora è già prenotata una visita con priorità inferiore. <ul style="list-style-type: none"> <li>6.1. Se il sistema ottiene dei risultati dal DBMS, il sistema cerca il primo appuntamento libero possibile.</li> </ul> </li> </ol>

- |  |
|--|
| <p>6.2. Il sistema contatta il DBMS per spostare la prenotazione con priorità inferiore.</p> <p>6.3. Viene invocato il caso d'uso "Invio E-Mail SMS di avviso".</p> <p>7. Il sistema contatta il DBMS per salvare la visita prenotata dall'utente.</p> |
|--|

**PUNTO DI ESTENSIONE:**

Nel punto 2 se vi è caduta di connessione, il sistema invoca il caso d'uso "Caduta Connessione".

**POSTCONDIZIONE:**

Viene mostrato a video un avviso di riuscita.

<b>CASO D'USO: INVIO E-MAIL/SMS DI AVVISO</b>
---

**ATTORI:**

Tempo, DBMS

**PRECONDIZIONE:**

Una prenotazione è stata spostata dal sistema o da un utente autenticato come medico.

**SEQUENZA DEGLI EVENTI:**

1. Il caso si avvia ogni 30 secondi.
2. Il sistema contatta il DBMS per verificare se ci sono nuove prenotazioni con un avviso non ancora inviato.
3. Se il DBMS ottiene dei risultati, il sistema invia al paziente una E-Mail e un SMS contenente:
  - 3.5. Nuovo data e ora della prenotazione.
  - 3.6. Motivo per il quale la prenotazione è stata modificata.

### 3.4.2. Modello degli oggetti

Gli oggetti che appartengono al sistema sono stati identificati applicando l'euristica di Abbot e sono stati divisi in oggetti entity, boundary e control:

#### Classi Entity

Nome	Descrizione
<b>Utente</b>	Classe per immagazzinare i dati di un utente
<b>Ricetta</b>	Classe per immagazzinare i dati di una ricetta
<b>Prenotazione</b>	Classe per immagazzinare i dati di una prenotazione

#### Classi Boundary

Nome	Descrizione
<b>AdministratorModeWindow</b>	Classe utilizzata per interfacciare il sottosistema relativo all'amministratore
<b>ManagePermissionWindow</b>	Classe utilizzata, dall'utente con permessi di Amministratore, per modificare i permessi degli altri utenti del sistema
<b>ManageTurnWindow</b>	Classe utilizzata, dall'utente con permessi di Amministratore, per modificare i turni dei dottori riguardo ambulatori o reparti
<b>BookWindow</b>	Classe utilizzata da un utente per poter prenotare una visita inserendo, negli appositi campi, i dati della persona e della visita.
<b>DocumentDialog</b>	Classe utilizzata in fase di prenotazione per visualizzare l'elenco dei documenti da avere per effettuare una determinata visita
<b>CadutaConnessioneDialog</b>	Classe utilizzata per comunicare all'utente che il sistema non riesce a connettersi al server

<b>CalendarWindow</b>	Classe utilizzata in fase di prenotazione per mostrare il calendario
<b>EditCredentialsWindow</b>	Classe utilizzata da un utente per modificare le credenziali di accesso
<b>GuestModeWindow</b>	Classe utilizzata per interfacciare il sottosistema relativo al paziente non autenticato
<b>ManageHospitalitationWindow</b>	Classe utilizzata, dall'utente con permessi di Medico, per scegliere se: gestire la coda dei ricoveri o ricoverare un paziente
<b>ManageQueueWindow</b>	Classe utilizzata, dall'utente con permessi di Medico, per gestire la coda dei ricoveri
<b>NewHospitalitationWindow</b>	Classe utilizzata, dall'utente con permessi di Medico, per ricoverare un paziente
<b>LoginWindow</b>	Classe utilizzata per l'autenticazione di un utente
<b>MainWindow</b>	Classe utilizzata per fare scegliere a chi apre il software di: registrarsi, autenticarsi, prenotare da ospite
<b>ManageVisitWindow</b>	Classe utilizzata da un utente con permessi di Paziente o di Medico, per visualizzare le prenotazioni concluse e quelle ancora da effettuare
<b>ManageVisitDoctorWindow</b>	Classe utilizzata da un utente con permessi di Medico, per visualizzare le prenotazioni concluse e quelle ancora da effettuare riguardo i suoi pazienti
<b>ModifyBookWindow</b>	Classe utilizzata da un utente per modificare la data di prenotazione o fare la cancellazione di una visita
<b>ShowReportWindow</b>	Classe utilizzata per mostrare il referto di una visita effettuata

<b>MedicModeWindow</b>	Classe utilizzata per interfacciare il sottosistema relativo al medico
<b>NotificationDialog</b>	Classe utilizzata per mostrare all'utente i messaggi da parte del sistema riguardo eventuali errori e operazioni effettuate
<b>SignUpWindow</b>	Classe utilizzata per mostrare il form di registrazione, l'utente di default avrà i permessi di Paziente
<b>UserModeWindow</b>	Classe utilizzata per interfacciare il sottosistema relativo al paziente autenticato
<b>CausalWindow</b>	Classe utilizzata da un utente con permessi di Medico, per avvertire il paziente dello slittamento di una visita
<b>NewReportWindow</b>	Classe utilizzata da un utente con permessi di Medico, per scrivere un referto di una visita appena conclusa
<b>VisiteAmbulatorioWindow</b>	Classe utilizzata da un utente con permessi di Medico, per mostrare tutte le visite ancora da effettuare presso l'ambulatorio in cui lavora

### Classi Control

Nome	Descrizione
<b>AdministratorModeControl</b>	Classe di controllo utilizzata per gestire le funzionalità a cui accede l'utente con il permesso da amministratore.
<b>ManagePermissionControl</b>	Classe di controllo, riservata esclusivamente per l'utente che accede come amministratore, utilizzata per gestire i permessi degli altri utenti.

<b>ManageTurnControl</b>	Classe di controllo, riservata solo all'utente che accede come amministratore, utilizzata per gestire i turni dei medici.
<b>BookControl</b>	Classe di controllo, riservata ad un qualsiasi utente o ospite, utilizzata per prenotare una visita.
<b>DocumentControl</b>	Classe di controllo per i documenti necessari per la prenotazione di una visita
<b>CadutaConnessioneControl</b>	Classe di controllo utilizzata per gestire la funzionalità relativa alla Caduta di connessione ed effettuare la riconnessione al server
<b>CalendarControl</b>	Classe di controllo utilizzata per gestire la funzionalità relativa all'utilizzo del calendario
<b>EditCredentialsControl</b>	Classe di controllo utilizzata per gestire la funzionalità di modifica credenziali
<b>GuestModeControl</b>	Classe di controllo utilizzata per gestire la funzionalità di accesso come ospite
<b>ManageHospitalitationControl</b>	Classe di controllo utilizzata per gestire le funzionalità di gestione di nuovi ricoveri e code di ricovero
<b>ManageQueueControl</b>	Classe di controllo utilizzata per gestire la funzionalità di modifica coda di ricovero e per dimissione di un paziente
<b>NewHospitalitationControl</b>	Classe di controllo utilizzata per gestire la funzione di inserimento di un nuovo ricovero
<b>LoginControl</b>	Classe di controllo utilizzata per gestire la funzione di accesso per l'utente
<b>MainControl</b>	Classe di controllo utilizzata per gestire la schermata di accesso principale

<b>ManageVisitControl</b>	Classe di controllo utilizzata per gestire la funzionalità di gestione visite da parte di un utente
<b>ManageVisitDoctorControl</b>	Classe di controllo utilizzata per gestire le visite effettuate da un utente autenticato come medico
<b>ModifyBookControl</b>	Classe di controllo utilizzata per gestire la funzionalità di modifica della prenotazione
<b>ShowReportControl</b>	Classe di controllo per gestire la visualizzazione dei referti delle visite conclusive
<b>MedicModeControl</b>	Classe di controllo utilizzata per gestire la funzionalità di gestione della modalità medico
<b>NotificationControl</b>	Classe di controllo utilizzata per gestire la funzionalità di notifica al momento di un opportuno errore
<b>SignUpControl</b>	Classe di controllo utilizzata per gestire la funzionalità di registrazione per un utente
<b>UserModeControl</b>	Classe di controllo utilizzata per gestire la funzionalità della modalità utente
<b>CausalControl</b>	Classe di controllo utilizzata per gestire la funzionalità di spostamento visita da parte di un medico
<b>NewReportControl</b>	Classe di controllo utilizzata per gestire la funzionalità di inserimento di nuovo referto da parte di un medico al termine di una visita
<b>VisiteAmbulatorioControl</b>	Classe di controllo utilizzata per gestire la funzionalità di gestione visite in ambulatorio

### 3.4.3. Progetto concettuale – descrizione dell’archivio

L’obiettivo è la realizzazione di un archivio contenente: utenti, ricette, reparti, ambulatori, prenotazioni e ricoveri.

La notazione utilizzata è la seguente: **entità** e **relazioni**.

Ogni **persona** è caratterizzata da:

- Codice fiscale
- Nome
- Cognome
- Data di nascita
- Città
- Indirizzo di residenza
- Indirizzo E-Mail
- Numero di Telefono

Ogni **utente appartenente** a un **gruppo permessi** è caratterizzato da:

- Codice fiscale
- Password
- Il gruppo di permessi a cui appartiene
- Un campo che segnala l’avvenuta spedizione dell’E-Mail/SMS in fase di registrazione

Il **gruppo di permessi** è caratterizzato da:

- Codice
- Nome

Una **persona**, in possesso di una **ricetta**, può prenotare una **visita**.

Una **ricetta** è caratterizzata da:

- Un codice NRE
- Priorità
- Data di emissione
- CodiceTipo

Ogni **ricetta appartiene** a un **tipo**, il quale è caratterizzato da:

- Un codice
- Il nome del tipo
- Il reparto in cui effettuare la visita

In funzione del **tipo** di **ricetta**, questo, **richiede** dei **documenti**, i quali sono caratterizzati da:

- Codice
- Nome

Una **visita** è caratterizzata da:

- Codice NRE della ricetta
- Data
- Ora inizio
- Ora fine
- Un campo che segnala l'avvenuta spedizione dell'e-mail/sms dopo la prenotazione
- Un campo che segnala l'avvenuta spedizione dell'e-mail/sms di remind
- Codice fiscale del medico che ha effettuato la visita.

Ogni **visita**, se completata è **dotata** di un **referto**, il quale è caratterizzato da:

- Codice NRE
- Referto

Ogni **visita**, inoltre, può **avere** degli **avvisi**, i quali sono caratterizzati da:

- Codice NRE
- Descrizione Avviso
- Un campo che segnala l'avvenuta spedizione dell'e-mail/sms dell'avviso.

Un **medico** è caratterizzato da:

- Codice Fiscale
- Specializzazione

Un **medico lavora** o in un **ambulatorio** o in un **reparto**.

L'**ambulatorio** è caratterizzato da:

- Codice
- Nome

Un **reparto** è caratterizzato da:

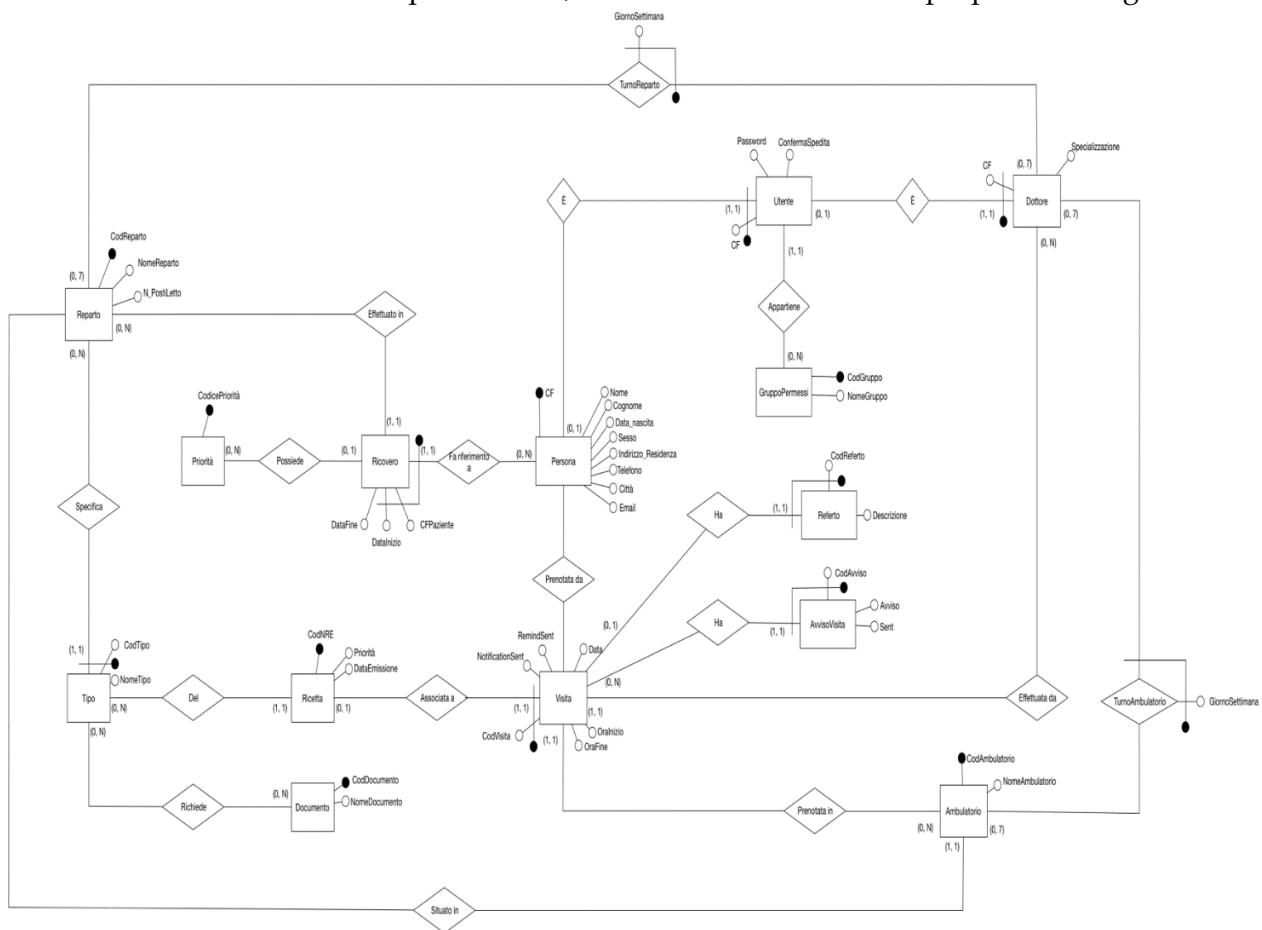
- Codice
- Nome
- Numero di posti letto

Un **medico** può **richiedere** un **ricovero** per una **persona**.

Il **ricovero** è caratterizzato da:

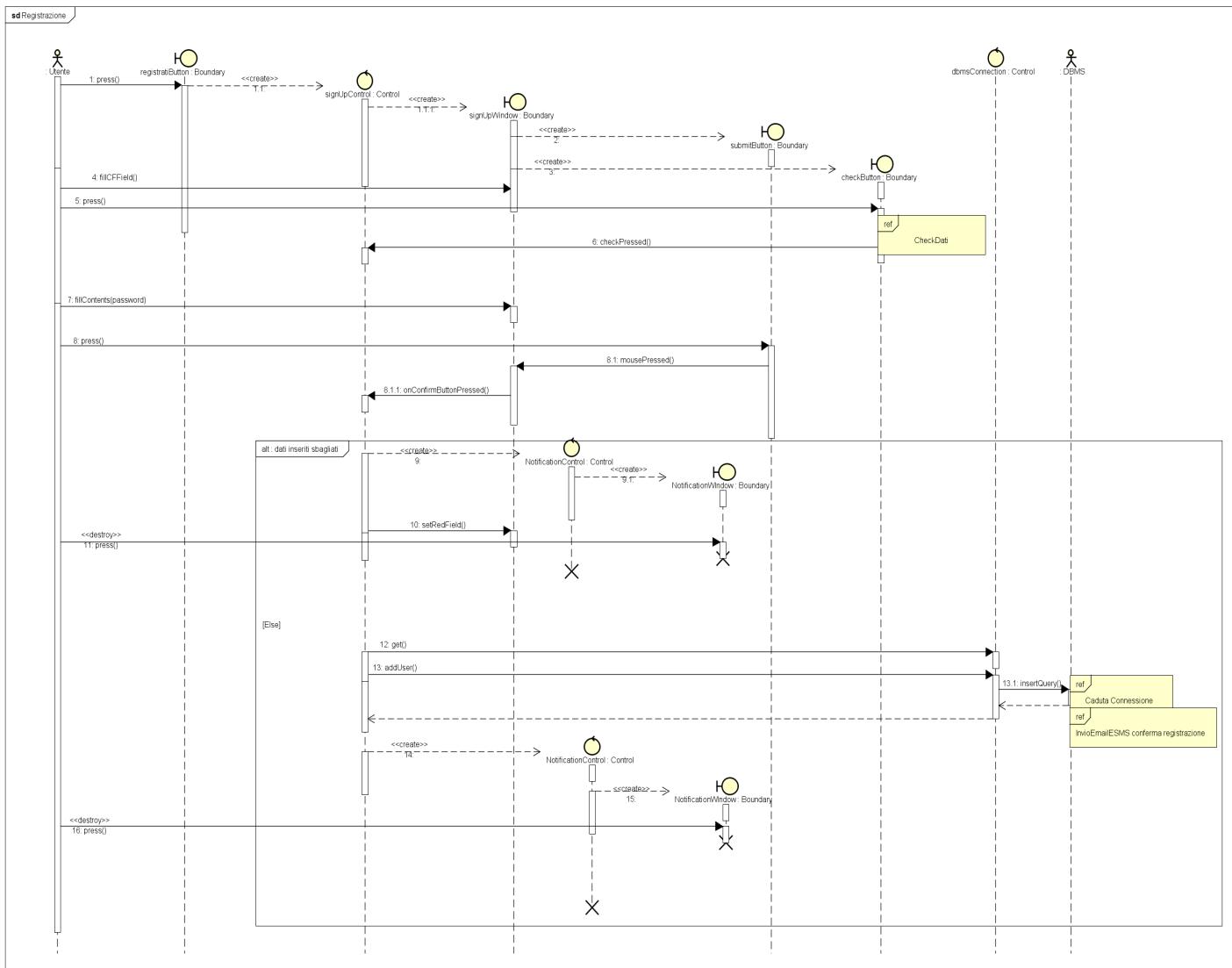
- Codice Fiscale del paziente
- Data di inizio del ricovero
- Data di dimissione
- Codice del reparto
- Numero di priorità

Una volta individuate queste entità, lo schema entità-relazione proposto è il seguente:

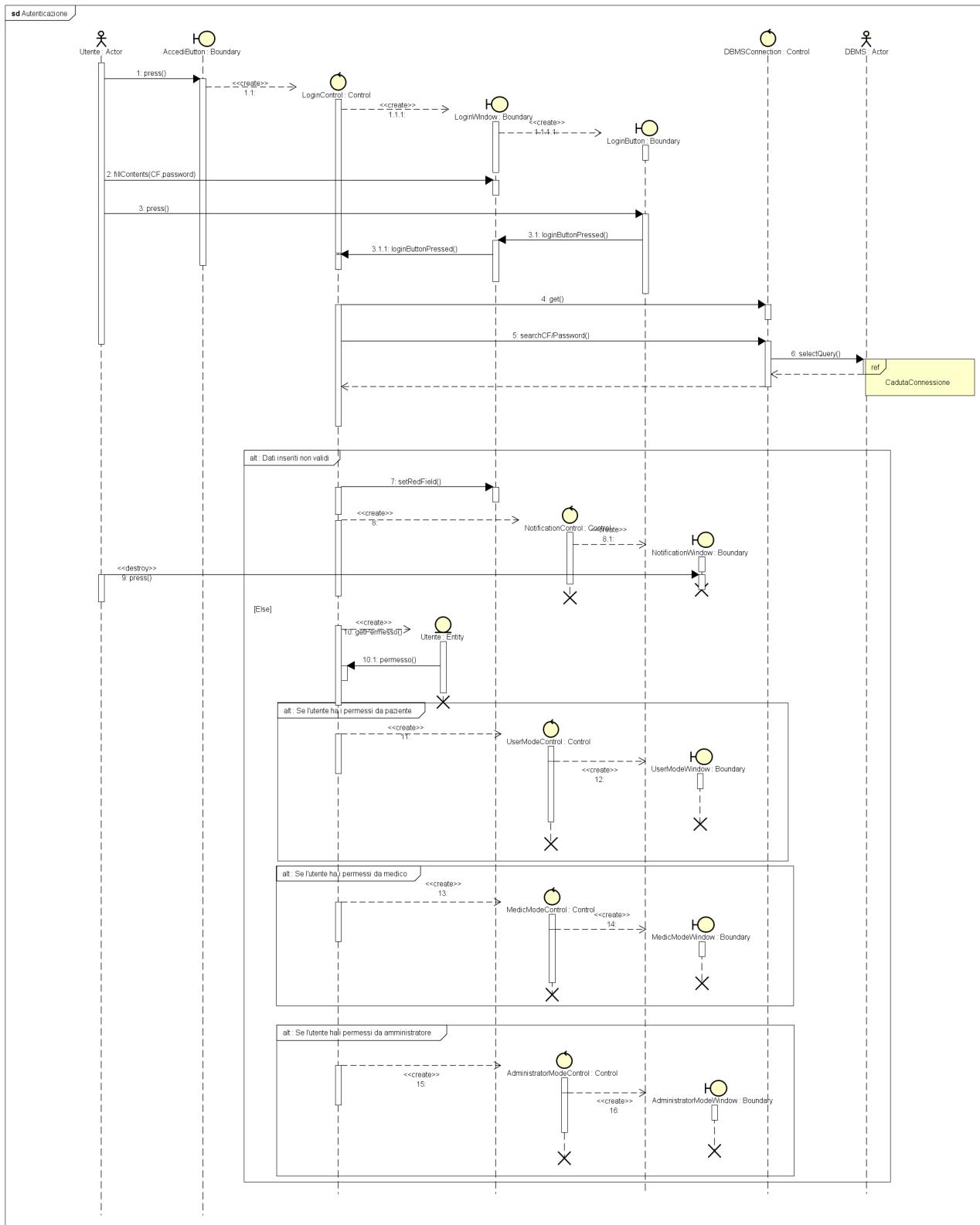


### 3.4.4. Modello dinamico

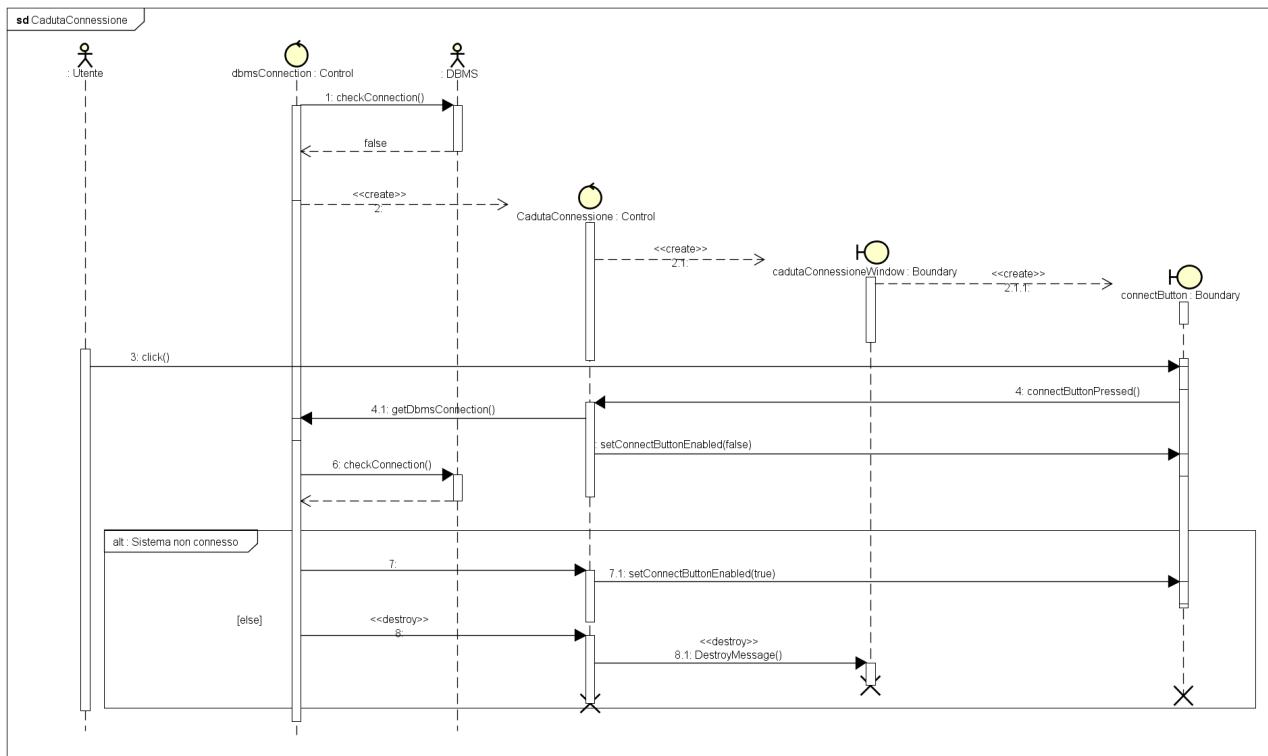
#### Diagramma di sequenza: Registrazione



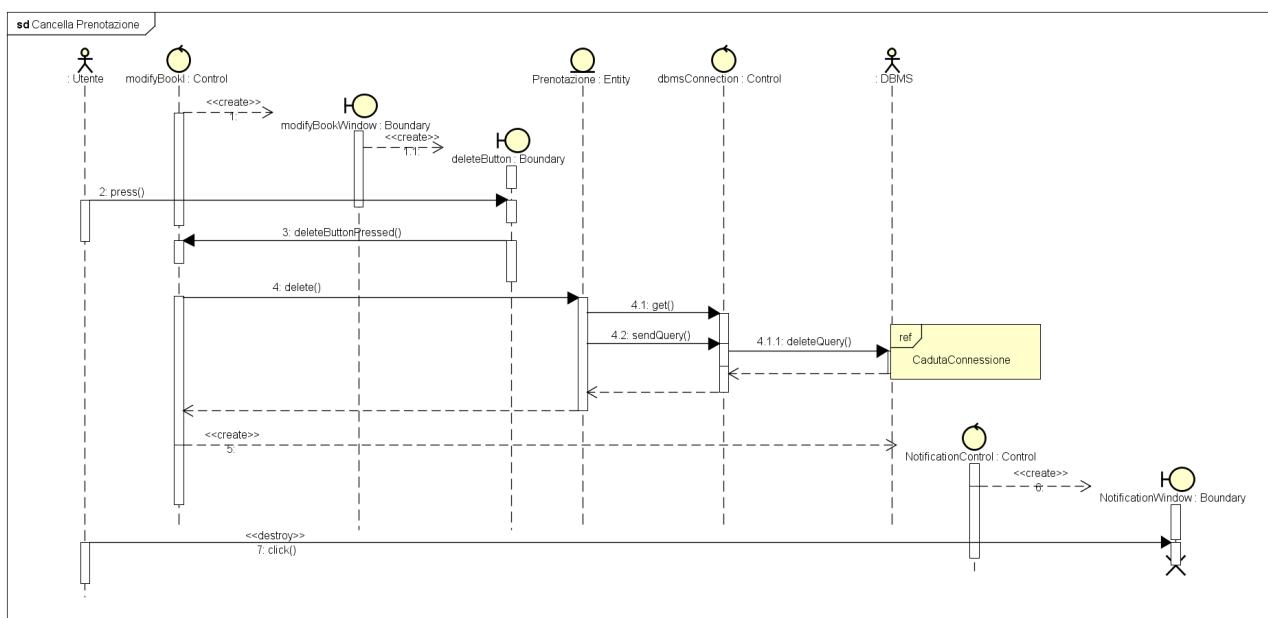
## Diagramma di sequenza: Autenticazione



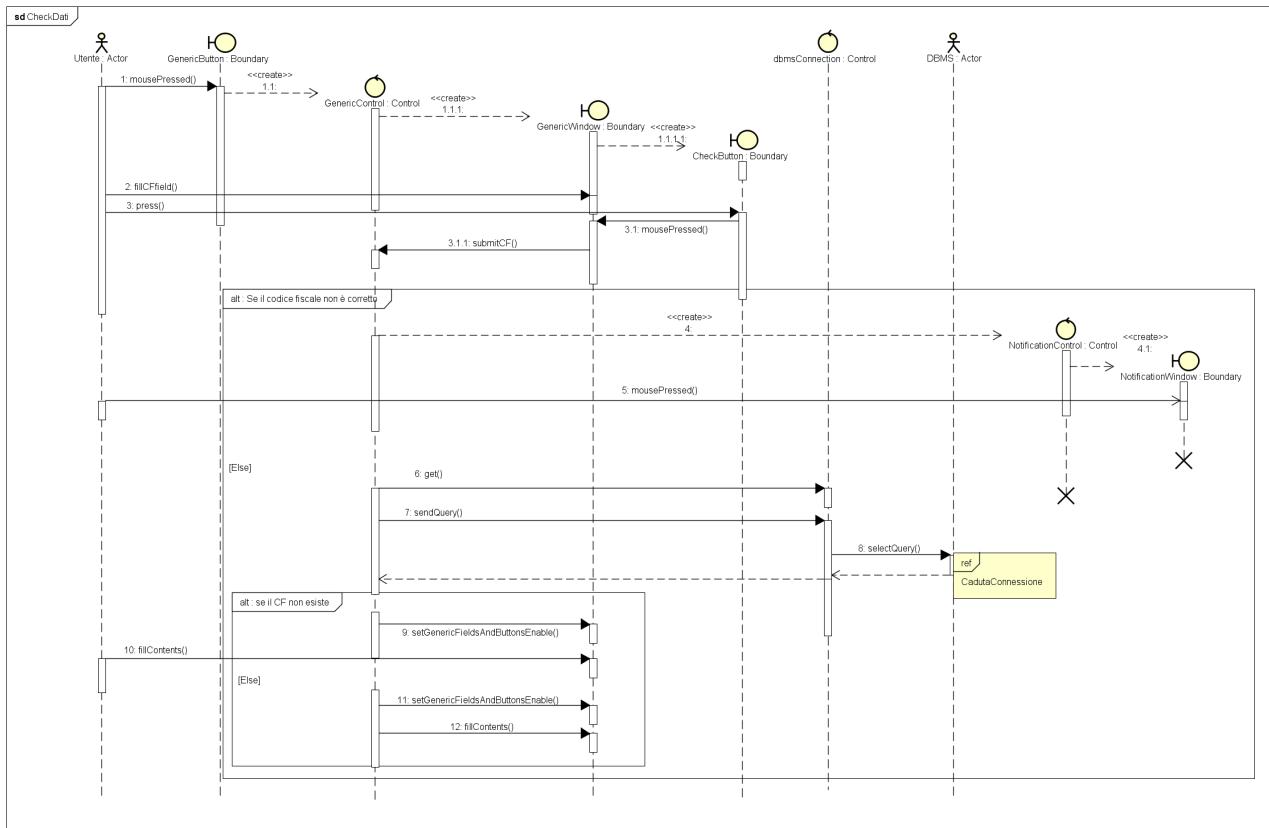
## Diagramma di sequenza: Caduta di connessione



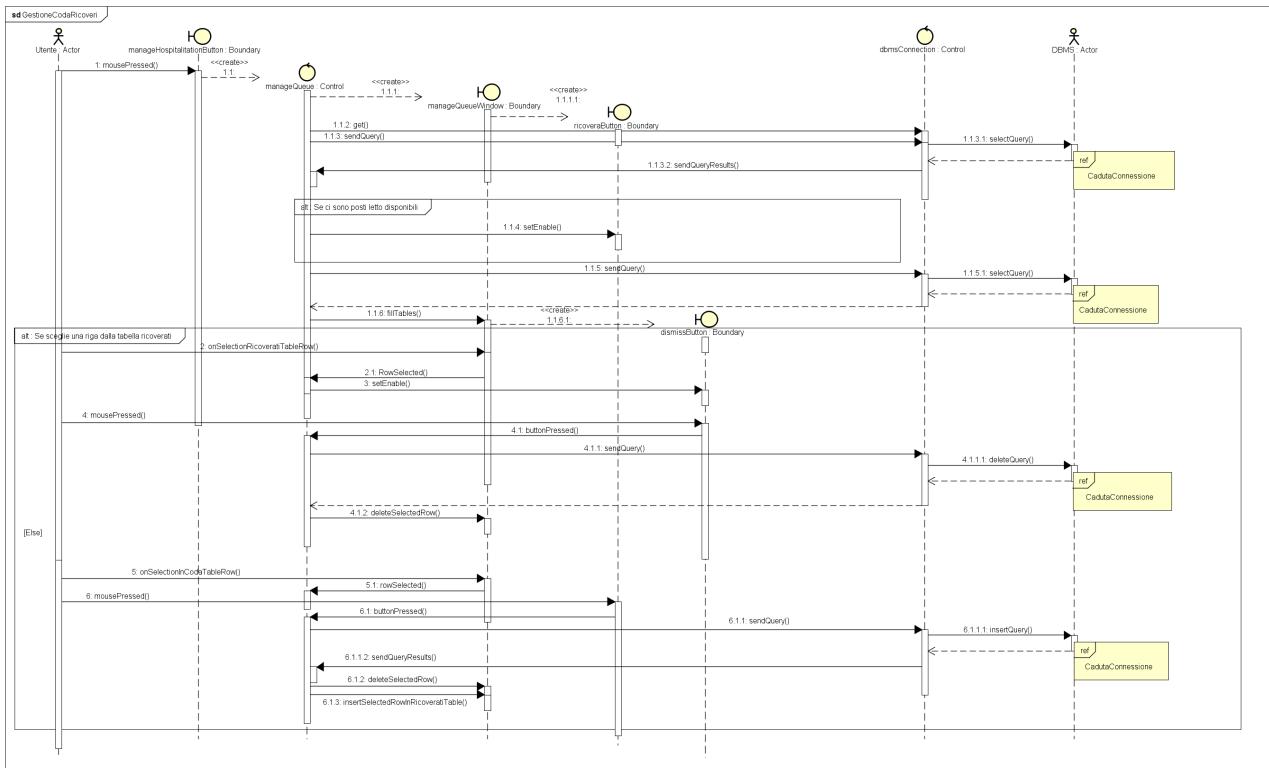
## Diagramma di sequenza: Cancella Prenotazione



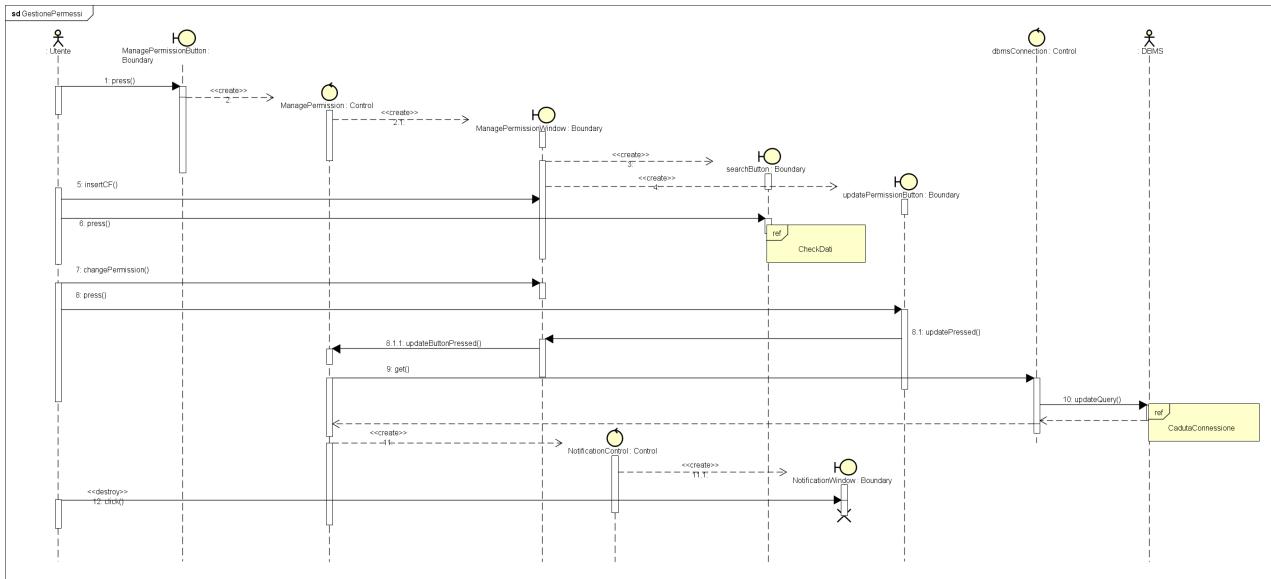
## Diagramma di sequenza: CheckDati



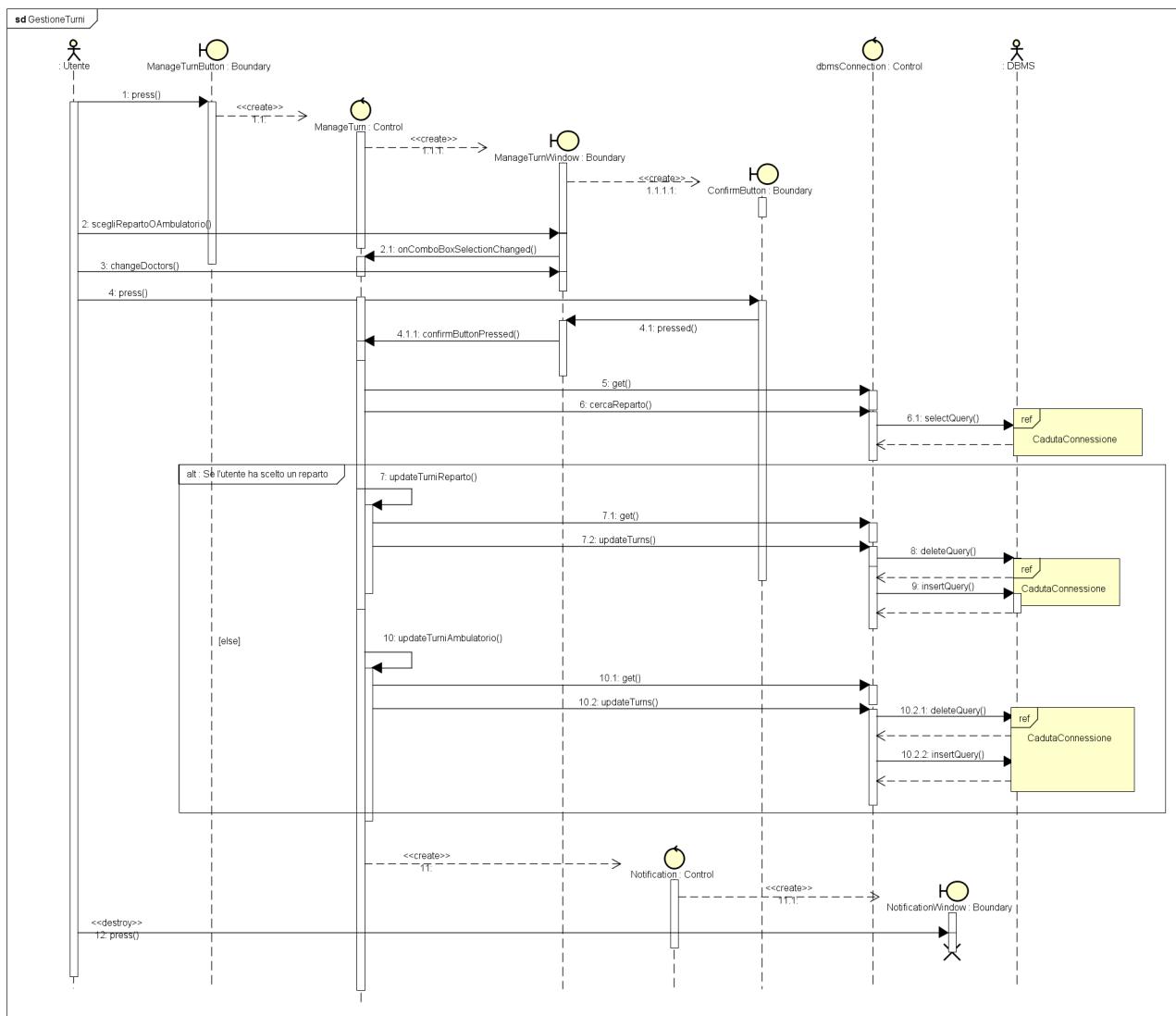
## Diagramma di sequenza: Gestione Coda Ricoverati



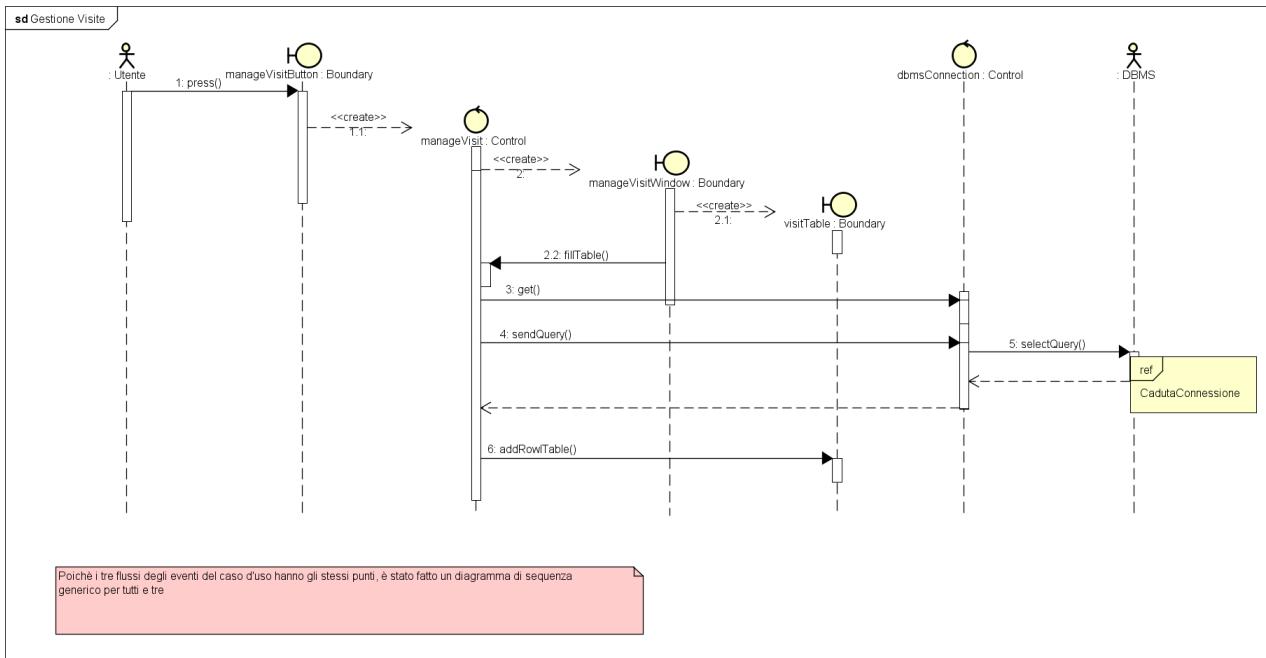
## Diagramma di sequenza: Gestione Permessi



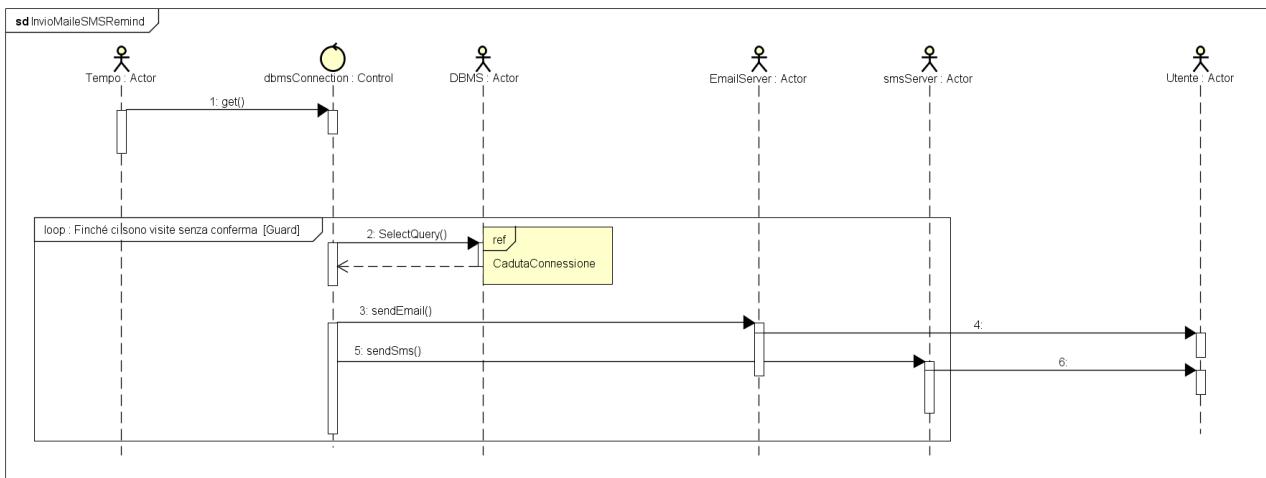
## Diagramma di sequenza: Gestione Turni



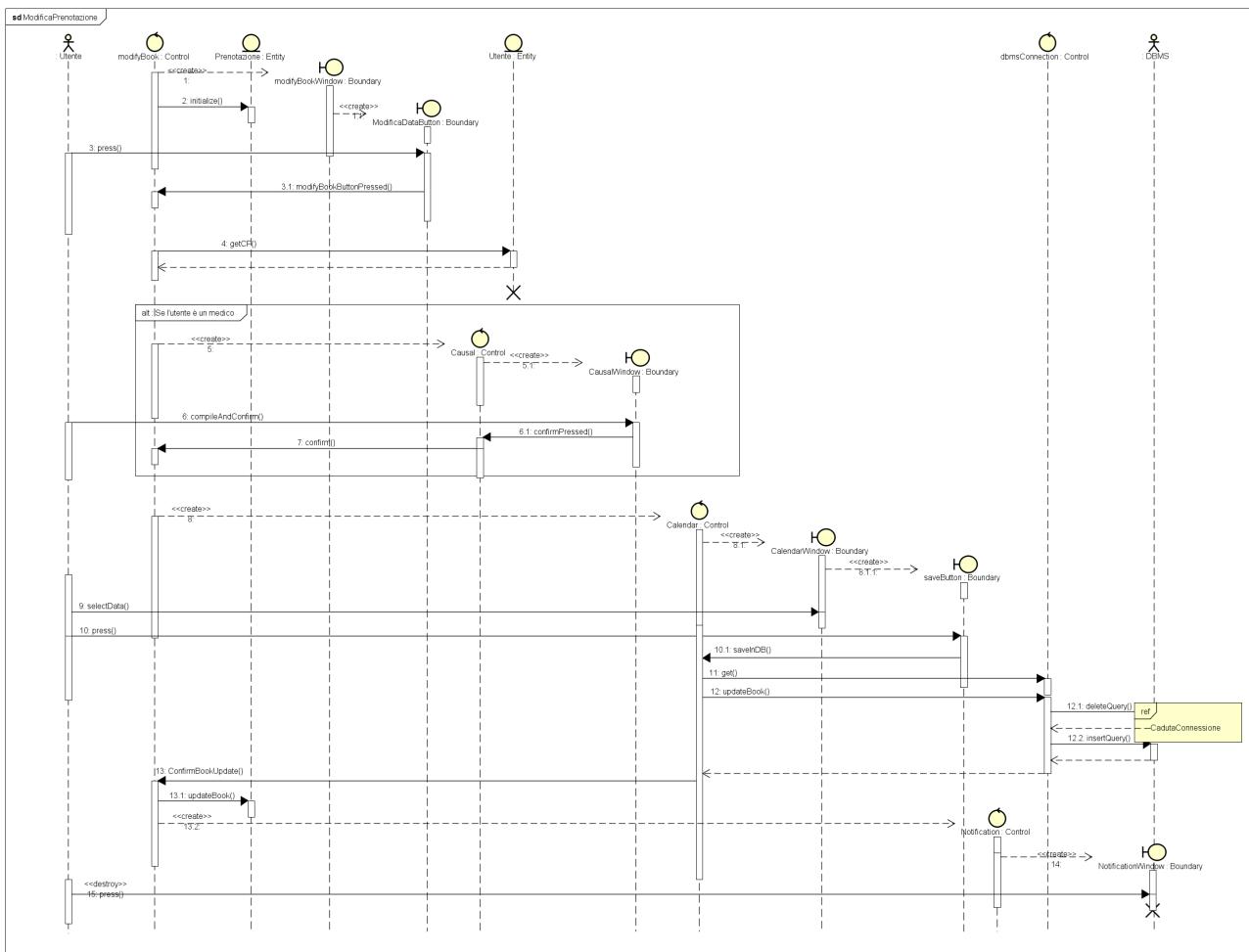
## Diagramma di sequenza: Gestione visite



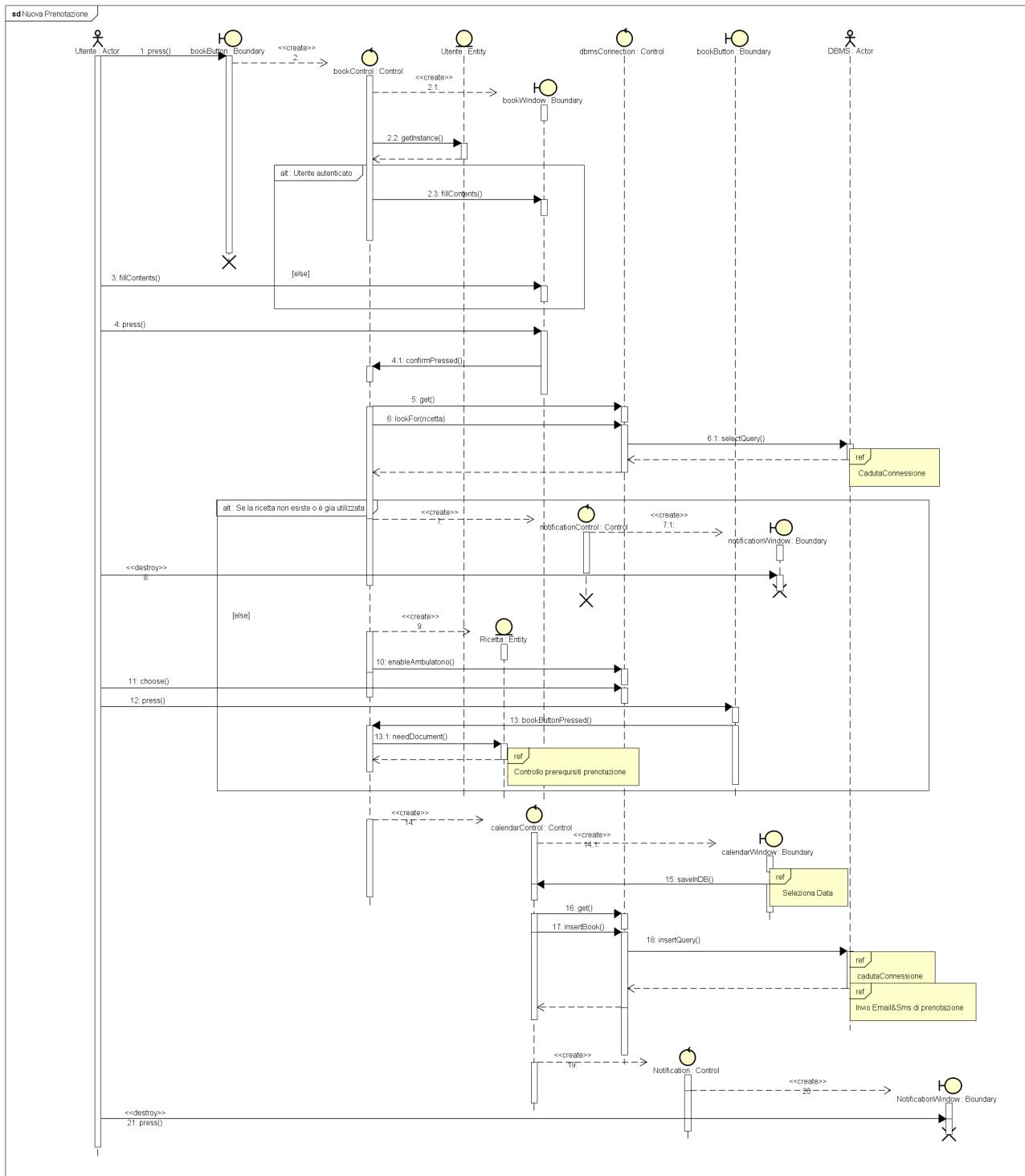
## Diagramma di sequenza: Invio E-Mail e SMS Remind



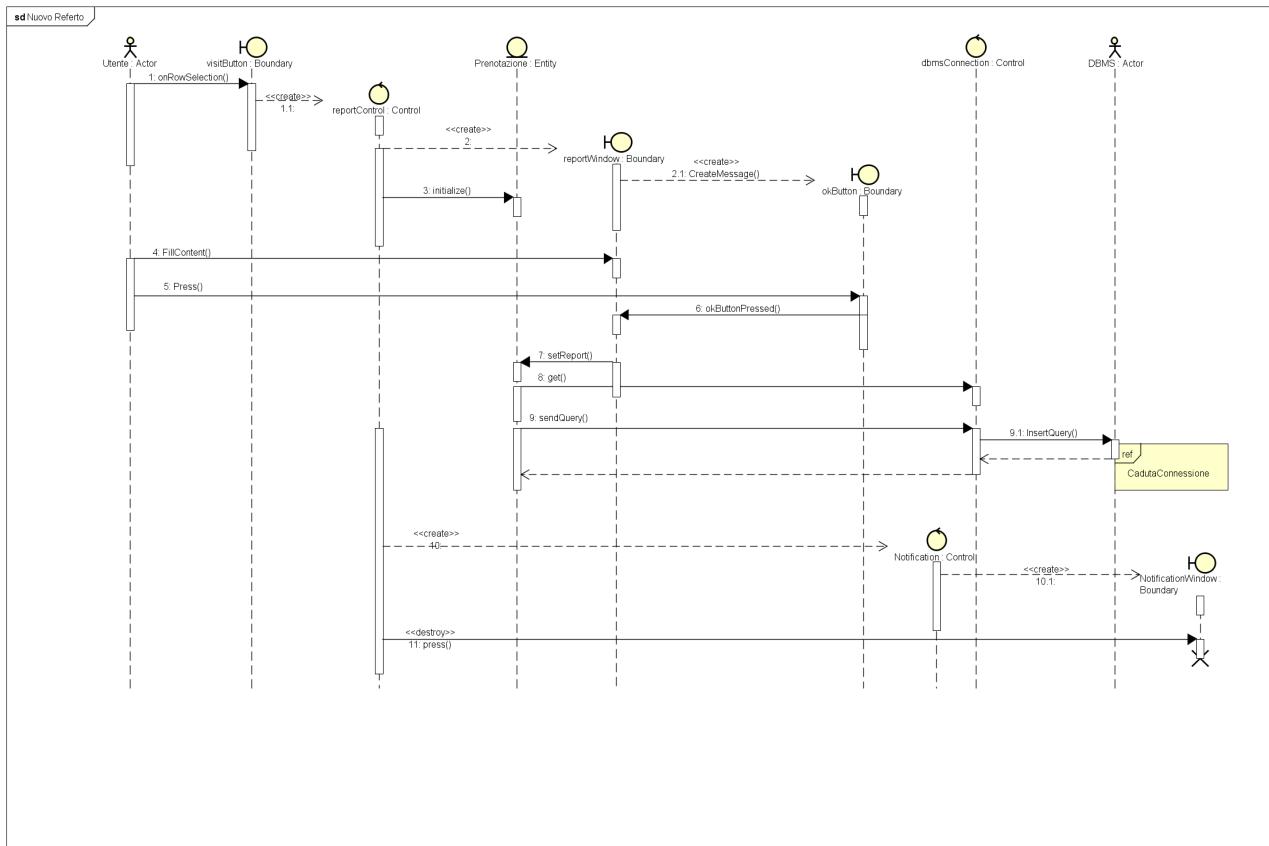
## Diagramma di sequenza: Modifica Prenotazione



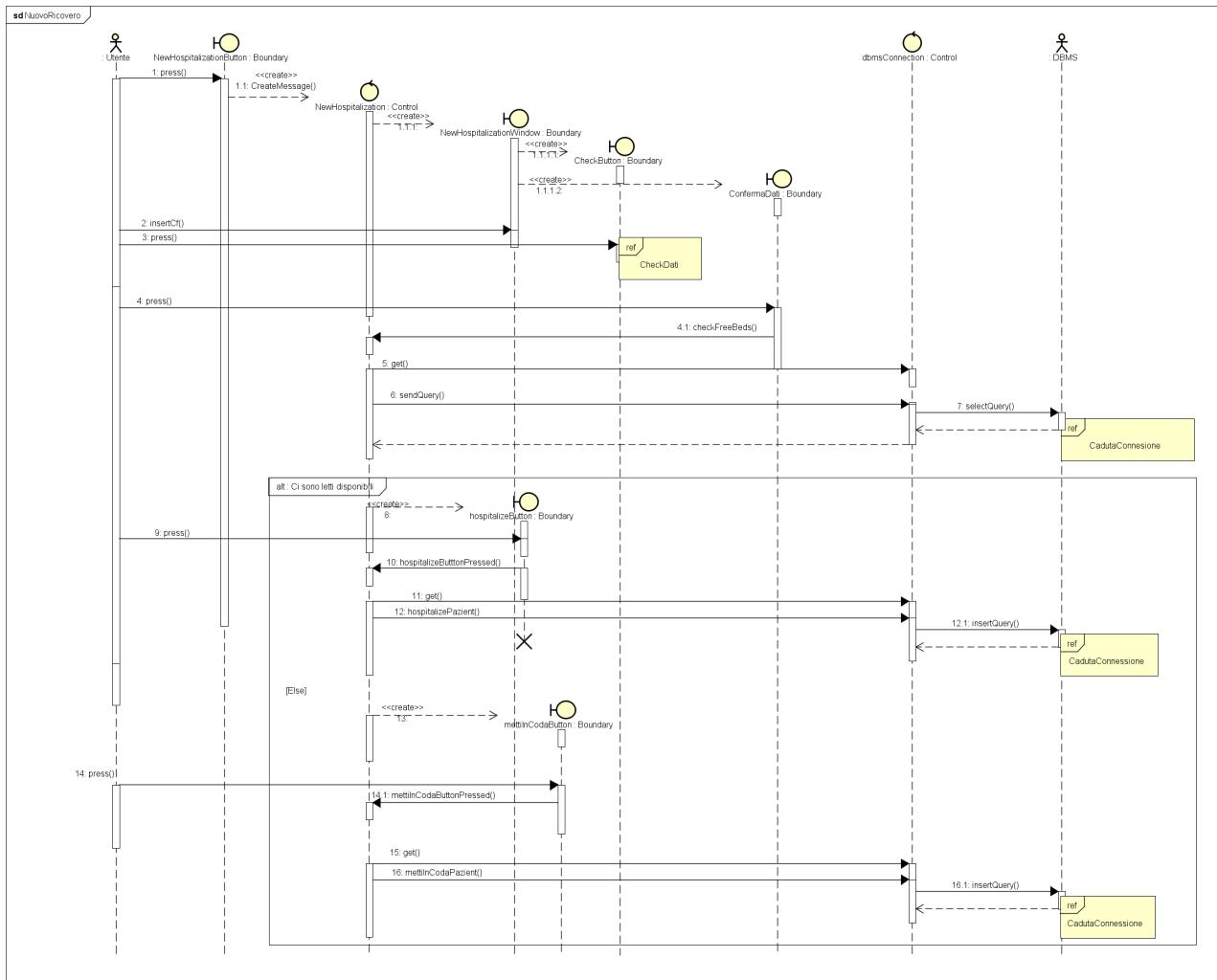
## Diagramma di sequenza: Nuova Prenotazione



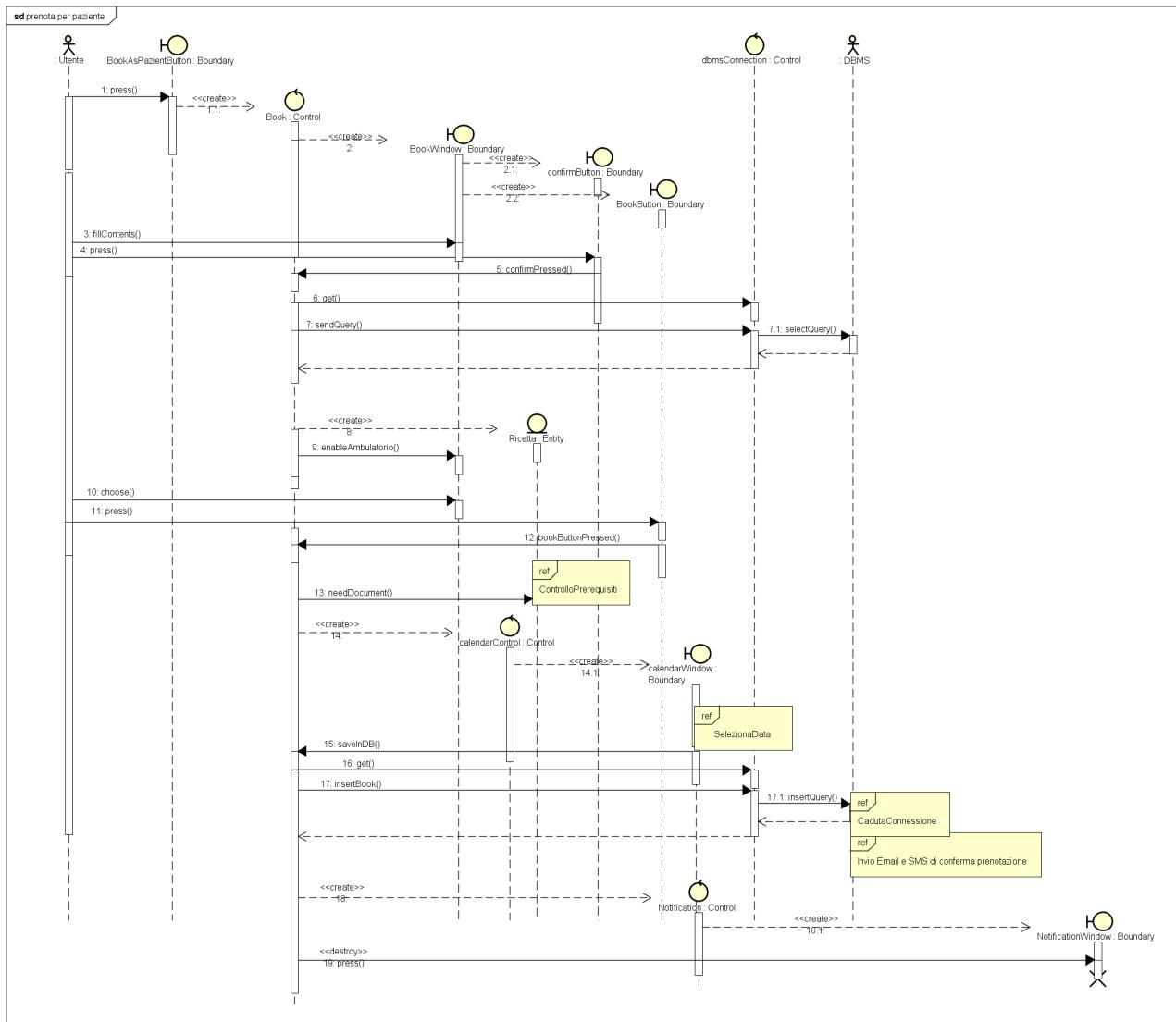
## Diagramma di sequenza: Nuovo Referto



## Diagramma di sequenza: Nuovo Ricovero

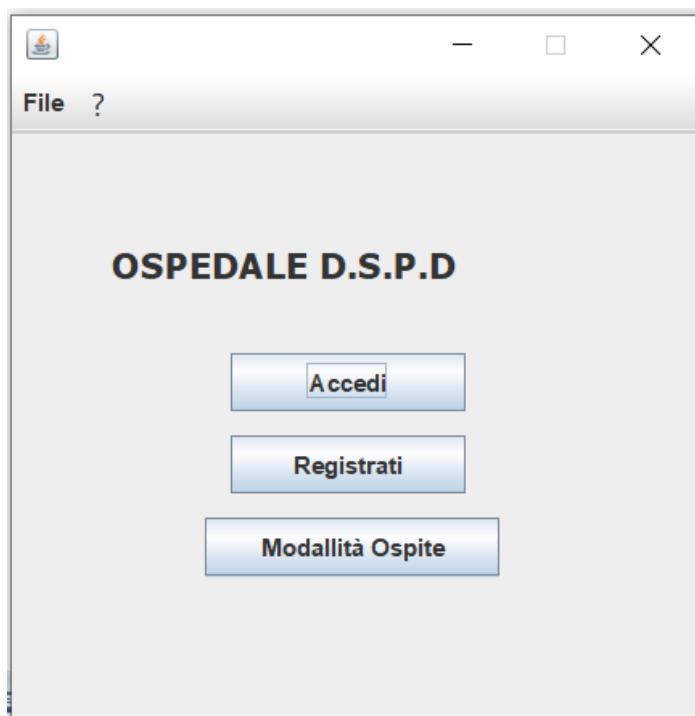


## Diagramma di sequenza: Prenota per paziente



### 3.4.5. Interfaccia Utente: Mock-up

Schermata principale

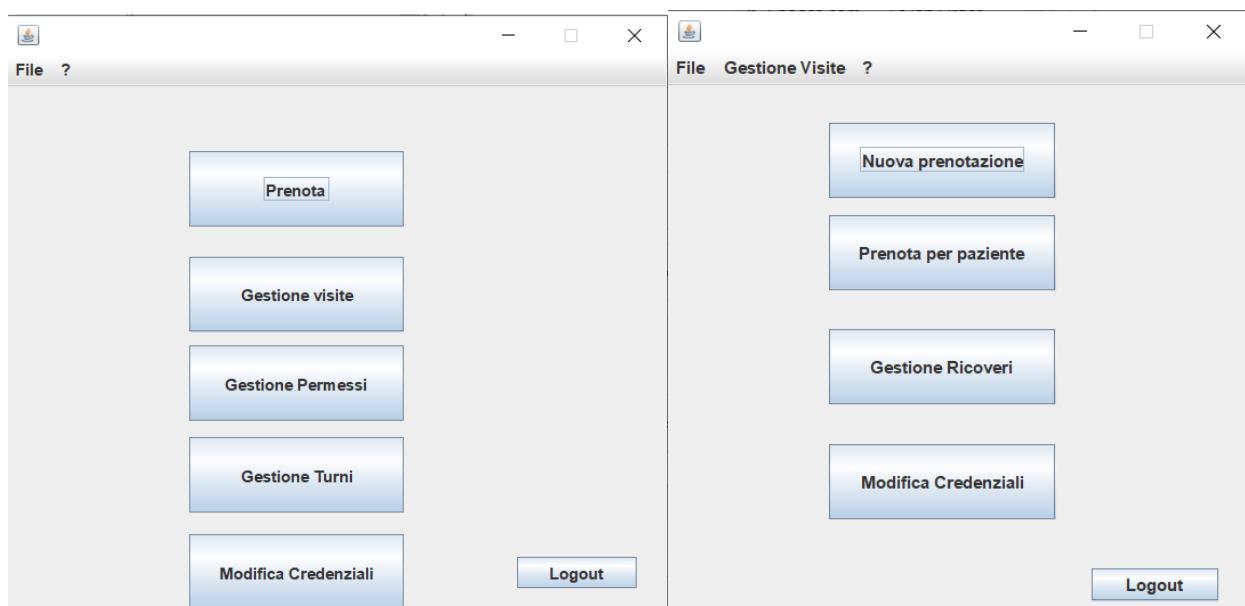


Schermata di Login

The image shows a login screen titled "Schermata di accesso" with fields for "Codice Fiscale" and "Password", and buttons for "Login" and "Indietro".

Pagine principali delle varie tipologie di accesso:

- Modalità ospite (in alto)
- Modalità amministratore (sinistra)
- Modalità medico (destra)
- Modalità utente (in basso)



## Pagina di inserimento dati prenotazione

This window allows users to input personal information for a booking. Fields include:

- Codice Fiscale: [Text Box]
- Nome: [Text Box]
- Cognome: [Text Box]
- Data di Nascita: [Day Month Year dropdowns]
- Sesso: [Gender dropdown: M]
- Via: [Text Box]
- Città: [Text Box]
- Email: [Text Box]
- Telefono: [Text Box]
- Codice NRE ricetta: [Text Box]
- Conferma: [Button]
- Reparto: [Text Box]
- Ambulatorio: [Dropdown menu]

Buttons at the bottom: Prenota (Book) and Indietro (Back).

## Calendario per selezione data

This calendar window shows the month of February 2019. Green numbers indicate available days for booking. A memo note is present on the right side.

DOM	LUN	MAR	MER	GIO	VEN	SAB
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

Memo: 2/9/2019

Buttons at the bottom: Salva (Save) and Indietro (Back). A note at the bottom says: "Per non perdere la priorità, prenotare nei giorni segnati di verde" (To avoid losing priority, book on the days marked in green).

## Pagina di modifica dati prenotazione

This window displays existing booking details for modification. Fields include:

- NRE: 30H00846963515
- Tipo Visita: fisioterapeutica
- Data: 2019-02-11
- Ora Inizio: 15:00:00
- Ora Fine: 15:30:00
- Ambulatorio: Fisioterapia1
- Reparto: Fisioterapia

Buttons at the bottom: Modifica data (Modify data), Elimina prenotazione (Delete booking), and Indietro (Back).

## Pagina di modifica dei dati personali

This screenshot shows a Windows-style application window titled 'File ?'. The form contains fields for personal information:

- Nome: Giuseppe
- Cognome: Peri
- Città: Scillato
- Indirizzo: Luigi Pirandello, 4
- Codice fiscale: pregpp97m06l112b
- Sesso: M
- Data di nascita: 06/08/1997
- Vecchia password: (empty)
- Nuova password: (empty)
- Ripeti password: (empty)
- Email: peppepe97@gmail.com
- Numero di telefono: 3922474469

At the bottom are 'Salva' and 'Indietro' buttons.

## Pagina gestione dei permessi

This screenshot shows a Windows-style application window titled 'Pagina gestione permessi.'. The form includes a search function and fields for filtering data:

- Codice Fiscale: (input field)
- Cerca: (button)
- Nome: (input field)
- Cognome: (input field)
- Email: (input field)
- Telefono: (input field)
- Indirizzo: (input field)
- Città: (input field)
- Sesso: M
- Data di Nascita: (dropdown menus for Giorno, Mese, Anno)
- Gruppo Permesso: Paziente
- Specializzazione: (input field)

At the bottom are 'Modifica Dati' and 'Indietro' buttons.

## Pagina di registrazione

This screenshot shows a Windows-style application window titled "Pagina di registrazione". The interface includes a "File" menu and a toolbar with a question mark icon. The main area contains the following fields:

- Codice Fiscale: An input field with a "Check" button to its right.
- Password: An input field.
- Ripeti Password: An input field.
- Nome: An input field.
- Cognome: An input field.
- Data di nascita: Three dropdown menus for Day, Month, and Year.
- Numero di telefono: An input field.
- Sesso: A dropdown menu showing "M".
- Email: An input field.
- Indirizzo: An input field.
- Città: An input field.

At the bottom left, a message states "Tutti i campi sono obbligatori". At the bottom right, there are "Indietro" and "Conferma" buttons.

## Pagina di gestione coda ricoveri

This screenshot shows a Windows-style application window titled "Pazienti ricoverati". The interface includes a "File" menu and a toolbar with a question mark icon. The main area is divided into two sections:

**Pazienti ricoverati**

Codice Fiscale	Paziente	Ricoverato dal

**Pazienti in coda**

Codice Fiscale	Paziente	In coda dal	Numero

Buttons available in the interface include "Dimetti Paziente", "Ricovera", and "Indietro".

## Pagina di inserimento nuovo ricovero

The screenshot shows a Windows-style application window titled "Inserire i dati del paziente". The interface includes:

- A "Codice Fiscale" input field with a "Check" button to its right.
- "Nome" and "Cognome" input fields.
- "Data Nascita" (Birth Date) input field with dropdowns for "Giorno" (Day), "Mese" (Month), and "Anno" (Year).
- "Sesso" (Sex) input field showing "M" (Male).
- "EMail" and "Telefono" input fields.
- "Indirizzo" and "Città" (City) input fields.
- A "Conferma Dati" (Confirm Data) button at the bottom.
- An "Indietro" (Back) button at the bottom right.

## Pagina di gestione dei turni

The screenshot shows a Windows-style application window titled "Selezione un reparto o un ambulatorio". The interface includes:

- A dropdown menu labeled "-- Reparti --".
- A grid of dropdown menus for each day of the week:
  - Lunedì
  - Martedì
  - Mercoledì
  - Giovedì
  - Venerdì
  - Sabato
  - Domenica
- A "Conferma" (Confirm) button and an "Indietro" (Back) button at the bottom.

# **System Design Document**

# **Indice**

- 1. Introduzione**
- 2. Architettura software corrente**
- 3. Architettura software proposta**
  - 3.1. Decomposizione in sottosistemi
  - 3.2. Mapping hardware-software
  - 3.3. Gestione dati persistenti
    - 3.3.1. Progetto logico
  - 3.4. Controllo di accesso e sicurezza
  - 3.5. Condizioni Boundary

## 1. Introduzione

Il sistema realizzato si presenta come un sistema distribuito, con una base di dati centrale, comune alle diverse parti remote.

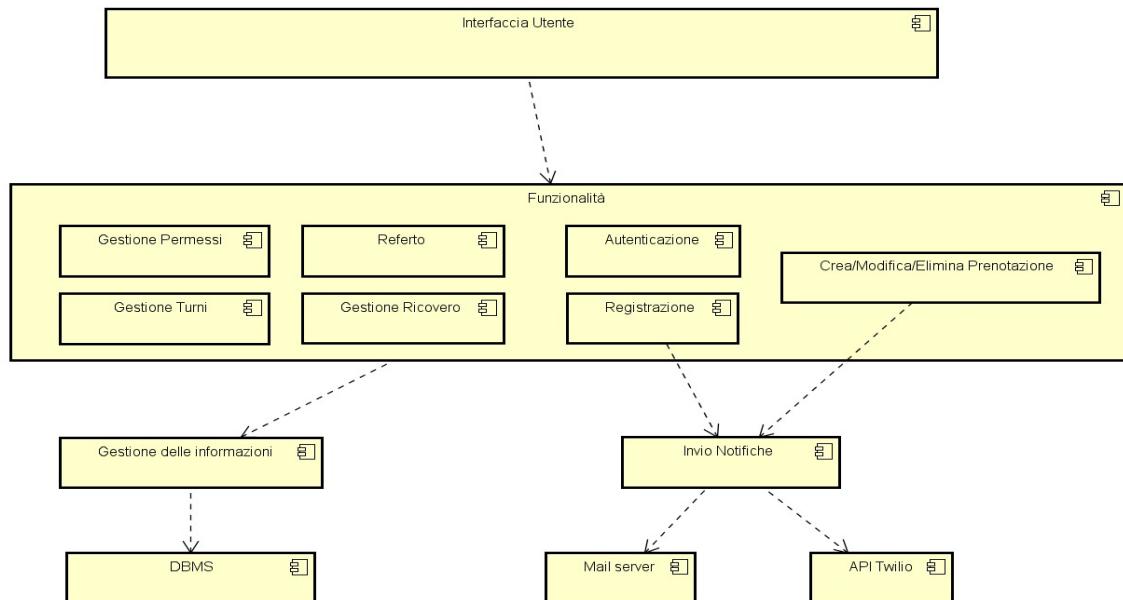
## 2. Architettura software corrente

Attualmente, la maggior parte dei sistemi informatici adoperati dalle aziende ospedaliere non permette una prenotazione online, da parte dei pazienti, tramite un programma messo a disposizione dall'ospedale stesso. In generale, gli stessi sistemi, non permettono al medico di accedere alla propria pagina personale, se non utilizzando i computer dell'ospedale. Tali computer, infatti, sono inseriti in una Intranet, ovvero una rete aziendale privata, che utilizza il protocollo TCP/IP, con possibilità di estendersi anche con collegamenti WAN e VPN.

## 3. Architettura software proposta

### 3.1. Decomposizione in sottosistemi

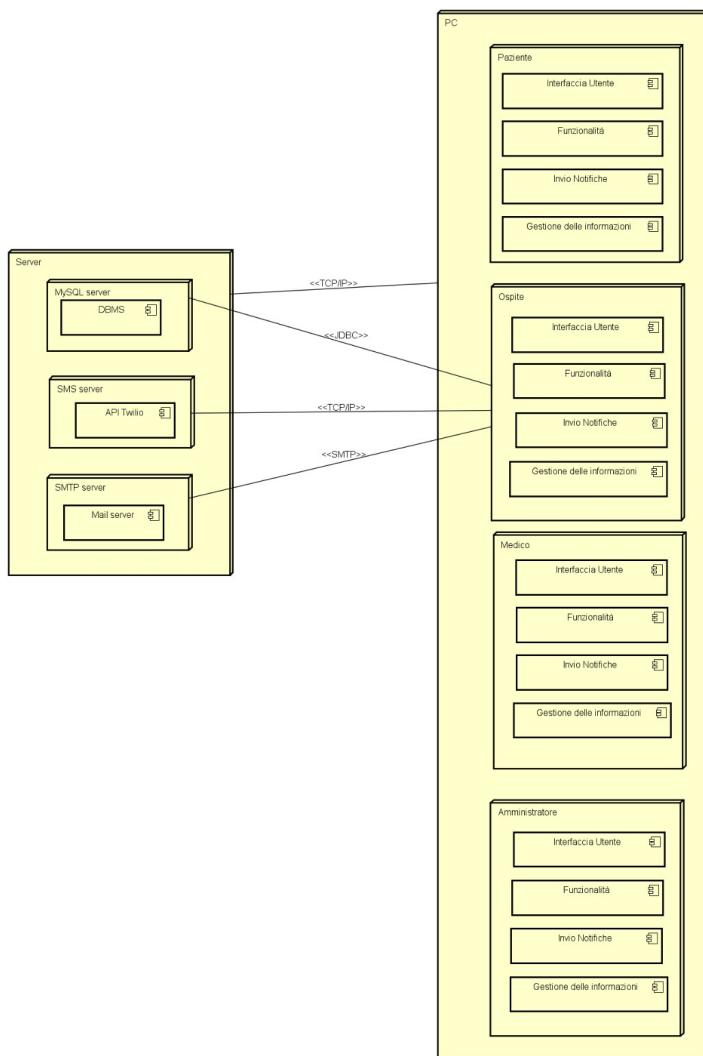
L'architettura proposta è un'architettura a Layer, secondo il seguente schema:



### 3.2. Mapping hardware-software

Il sistema è composto da due nodi, che comunicheranno tramite protocollo TCP/IP. Tali nodi consistono in:

- Il server conterrà un'istanza di MySQL e si occuperà di gestire i contenuti del Database, ovvero di fornirli e modificarli a seguito delle richieste, che saranno gestite a livello software da JDBC. Inoltre, conterrà un'istanza di un'E-Mail e un SMS server, utili all'invio delle notifiche all'utente. Queste saranno gestite a livello software, rispettivamente, dal protocollo SMTP e TCP/IP.
- Il pc rappresenta un qualunque personal computer adibito all'installazione del sistema proposto, il quale cambierà interfaccia e funzionalità in base all'autentificazione dell'utente, secondo l'architettura a Layer, precedentemente proposta.



**N.B:** Per motivi di ordine a livello grafico, si è preferito inserire le connessioni (con i relativi nomi) di una sola tipologia di utente (ospite) in quanto, sia per medico, sia per amministratore e sia per paziente, i tipi di connessione sono analoghi.

### 3.3. Gestione dei dati del sistema

#### 3.3.1. Progetto logico

Si utilizza un DB relazionale, basato sulle relazioni esistenti fra le varie tabelle. Il DB è stato realizzato con l'ausilio di MySQL.

In base all'analisi fatta nel RAD e allo schema E-R, date le cardinalità (1,1), sono state eliminate le seguenti relazioni:

1. La relazione “E” tra “Utente” e “Persona” si trasferisce su “Utente”, rinominando l’attributo “CF” in “RefCF”.
2. La relazione “Appartiene” tra “Utente” e “Gruppo Permessi” viene rimossa mediante la chiave esterna da “RefGruppoPermessi” di “Utente” a “CodGruppo” di “GruppoPermessi”.
3. La relazione “Prenotata in” tra “Visita” e “Ambulatorio” viene rimossa mediante la chiave esterna da “RefAmbulatorio” di “Visita” a “CodAmbulatorio” di “Ambulatorio”.
4. La relazione “Associata a” tra “Visita” e “Ricetta” viene rimossa mediante la chiave esterna da “RefNRE” di “Visita” a “CodNRE” di “Ricetta”.
5. La relazione “Prenotata da” tra “Visita” e “Persona” viene rimossa mediante la chiave esterna da “RefCfPaziente” di “Visita” a “CF” di “Persona”.
6. La relazione “Situato in” tra “Ambulatorio” e “Reparto” viene rimossa mediante la chiave esterna da “RefReparto” di “Ambulatorio” a “CodReparto” di “Reparto”.
7. La relazione “Del” tra “Ricetta” e “Tipo” viene rimossa mediante la chiave esterna da “RefTipo” di “Ricetta” e “CodTipo” di “Tipo”.
8. La relazione “Fa riferimento a” tra “Ricovero” e “Persona” viene rimossa mediante la chiave esterna da “ReCFPaziente” di “Ricovero” a “CF” di “Persona”.
9. La relazione “E” tra “Dottore” e “Utente” viene rimossa mediante la chiave esterna da “RefCFDottore” in “Dottore” a “RefCF” di “Utente”.

10. La relazione “**Possiede**” tra “Ricovero” e “Priorità” viene rimossa mediante l’attributo “Priorità” in “Ricovero”.
11. La relazione “**Effetuato in**” tra “Ricovero” e “Reparto” viene rimossa mediante la chiave esterna da “RefReparto” di “Ricovero” a “CodReparto” di “Reparto”.
12. La relazione “**Specifica**” tra “Tipo” e “Reparto” viene rimossa mediante la chiave esterna da “RefReparto” di “Tipo” a “CodReparto” di “Reparto”.
13. La relazione “**Ha**” tra “Referto” e “Visita” si trasferisce su “Referto”, rinominando l’attributo “CodReferto” in “RefNRE”.
14. La relazione “**Ha**” tra “AvvisoVisita” e “Visita” si trasferisce su “AvvisoVisita”, rinominando l’attributo “CodAvviso” in “RefNRE”.
15. La relazione “**Effettuata da**” tra “Dottore” e “Visita” viene rimossa mediante la chiave esterna da “RefCfDottore” in “Visita” a “RefCfDottore” in “Dottore”.

In conclusione, il risultato finale è il seguente:

**Entità:**

**Ambulatorio** (CodAmbulatorio, NomeAmbulatorio, RefReparto)

**AvvisoVisita** (RefNRE, Avviso, Sent)

**Documento** (CodDocumento, NomeDocumento)

**Dottore** (RefCFDottore, Specializzazione)

**GruppoPermesso** (CodGruppo, NomeGruppo)

**Persona** (CF, Nome, Cognome, Data\_Nascita, Sesso, Indirizzo, Residenza, Telefono, Citta, E-mail)

L’entità “**Priorità**” è stata eliminata.

**Referto** (RefVisita, Descrizione)

**Reparto** (CodReparto, NomeReparto, N\_PostiLetto)

**Ricovero** (RefCFPaziente, DataInizio, DataFine, RefReparto, Priorità)

**Ricetta** (CodNRE, Priorità, RefTipo, DataEmissione)

**Tipo** (CodTipo, NomeTipo, RefReparto)

**Utente** (RefCF, Password, RefGruppoPermessi, ConfermaSpedita)

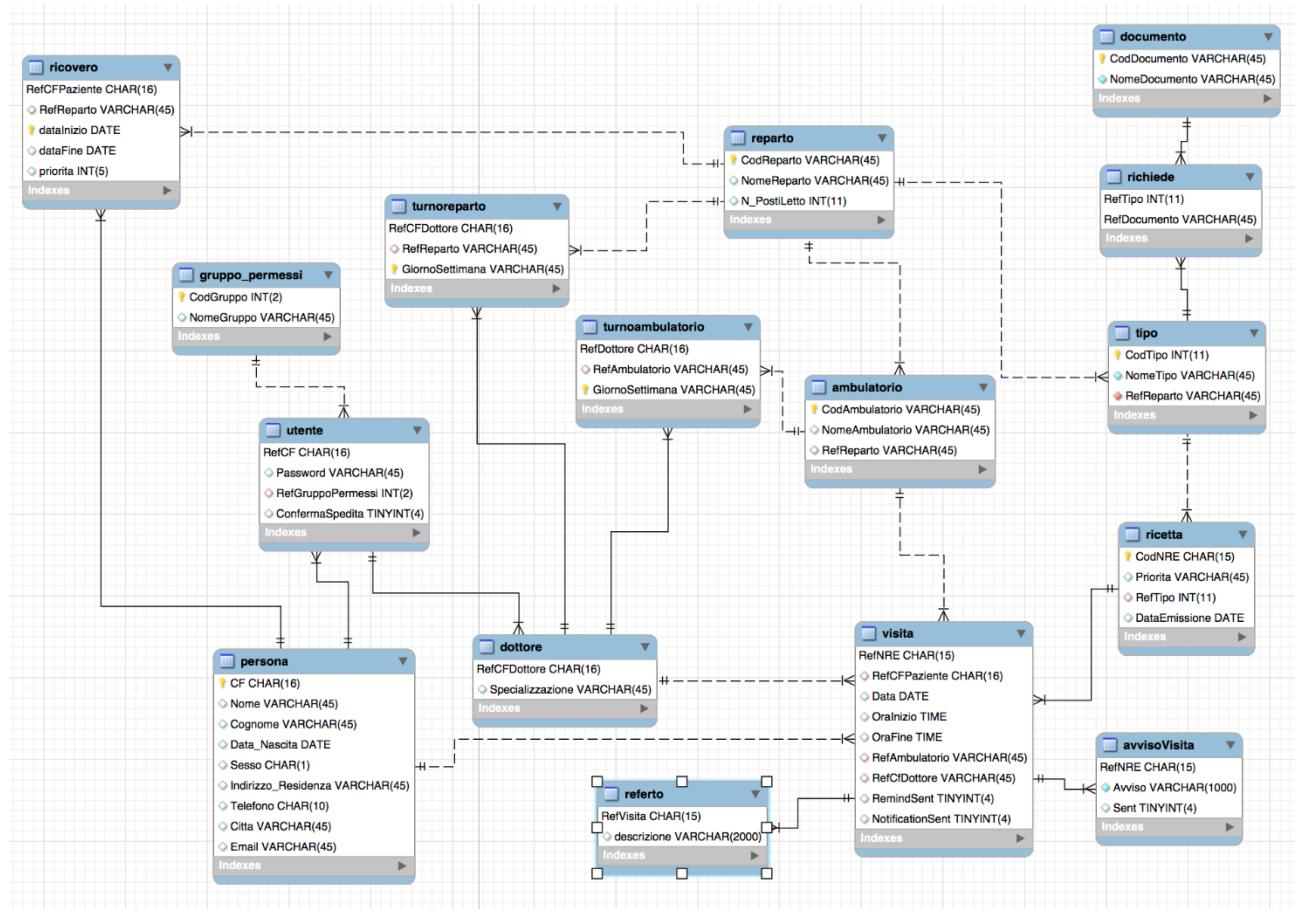
**Visita** (RefNRE, RefCFPaziente, Dara, OraInizio, OraFine, RefAmbulatorio, RefCfDottore, RemindSent, NotificationSent)

**Relazioni:**

**Richiede** (RefTipo, RefDocumento)

**TurnoAmbulatorio** (RefCFDottore, GiornoSettimana, RefAmbulatorio)

**TurnoReparto** (RefCFDottore, GiornoSettimana, RefReparto)



Le tabelle che costituiscono il DB sono:

## RICOVERO

Campo	Tipo
<b>RefCfPaziente</b>	char(16) (richiesto: si, chiave primaria: si )
<b>RefReparto</b>	Varchar(45) (richiesto: si, chiave primaria: no)
<b>dataInizio</b>	Date (richiesto: si, chiave primaria: si)
<b>dataFine</b>	Date (richiesto: si, chiave primaria: no)
<b>Priorità</b>	Int(5) (richiesto: si, chiave primaria: no)

## PERSONA

Campo	Tipo
CF	char (16) (richiesto: si, chiave primaria: si)
Nome	varchar(45) (richiesto: si, chiave primaria: no)
Cognome	Varchar(45) (richiesto: si, chiave primaria: no)
Data_Nascita	Date (richiesto: si, chiave primaria: no)
Sesso	Char(1) (richiesto: si, chiave primaria: no)
Indirizzo_Residenza	Varchar(45) (richiesto: si, chiave primaria: no )
Telefono	Char(10) (richiesto: si, chiave primaria: no)
Citta	Varchar(45) (richiesto: si, chiave primaria: no )
Email	Varchar(45) (richiesto: si, chiave primaria: no)

## UTENTE

Campo	Tipo
RefCF	Char(16) (richiesto: si, chiave primaria: no)
Password	Varchar(45) (richiesto: si, chiave primaria: no)
RefGruppoPermessi	Int(2) (richiesto: si, chiave primaria: no)
ConfermaSpedita	Tinyint (4) (richiesto: si, chiave primaria: no)

## DOTTORE

Campo	Tipo
RefCFDottore	Char(16) (richiesto: si, chiave primaria: si)
Specializzazione	Varchar(45) (richiesto: si, chiave primaria: no)

## GRUPPO PERMESSI

Campo	Tipo
CodGruppo	Int(2) (richiesto: si, chiave primaria: si)
NomeGruppo	Varchar(45) (richiesto: si, chiave primaria: no )

## TURNO REPARTO

Campo	Tipo
RefCFDottore	Char(16) (richiesto: si, chiave primaria: si)
RefReparto	Varchar(45) (richiesto: si, chiave primaria: no )
GiornoSettimana	Varchar(45) (richiesto: si, chiave primaria: si)

## REPARTO

Campo	Tipo
CodReparto	Varchar(45) (richiesto: si, chiave primaria: si)
NomeReparto	Varchar(45) (richiesto: si, chiave primaria: no )
N_PostoLetto	Int(11) (richiesto: si, chiave primaria: no)

## TURNO AMBULATORIO

Campo	Tipo
RefDottore	Char(16) (richiesto: si, chiave primaria: si)
RefAmbulatorio	Varchar(45) (richiesto: si, chiave primaria: no)
GiornoSettimana	Varchar(45) (richiesto: si, chiave primaria: si)

## AMBULATORIO

Campo	Tipo
CodAmbulatorio	Varchar(45) (richiesto: si, chiave primaria: si)
NomeAmbulatorio	Varchar(45) (richiesto: si, chiave primaria: no )
RefReparto	Varchar(45) (richiesto: si, chiave primaria: no )

## DOCUMENTO

Campo	Tipo
CodDocumento	Varchar(45) (richiesto: si, chiave primaria: si)
NomeDocumento	Varchar(45) (richiesto: si, chiave primaria: si)

## RICHIEDE

Campo	Tipo
RefTipo	Int(11) (richiesto: si, chiave primaria: si)
RefDocumento	Varchar(45) (richiesto: si, chiave primaria: si)

## TIPO

Campo	Tipo
CodTipo	Int(11) (richiesto: si, chiave primaria: si)
NomeTipo	Varchar(45) (richiesto: si, chiave primaria: si)
RefReparto	Varchar(45) (richiesto: si, chiave primaria: no )

## REFERTO

Campo	Tipo
RefVisita	Char(15) (richiesto: si, chiave primaria: si)
Descrizione	Varchar(2000) (richiesto: si, chiave primaria: no)

## RICETTA

Campo	Tipo
CodNRE	Char(15) (richiesto: si, chiave primaria: si)
Priorita	Varchar(45) (richiesto: si, chiave primaria: no )
RefTipo	Int(11) (richiesto: si, chiave primaria: no)
DataEmissione	Date (richiesto: si, chiave primaria: si)

## VISITA

Campo	Tipo
RefNRE	Char(15) (richiesto: si, chiave primaria: si)
RefCFPaziente	Char(16) (richiesto: si, chiave primaria: no)
Data	Date (richiesto: si, chiave primaria: no)
OraInizio	Date (richiesto: si, chiave primaria: no)
OraFine	Date (richiesto: si, chiave primaria: no)

<b>RefAmbulatorio</b>	Varchar(45) (richiesto: si, chiave primaria: no )
<b>RefCFDottore</b>	Char(16) (richiesto: si, chiave primaria: no)
<b>RemindSent</b>	Tinyint(4) (richiesto: si, chiave primaria: no)
<b>NotificationSent</b>	Tinyint(4) (richiesto: si, chiave primaria: no)

## AVVISO VISITA

Campo	Tipo
<b>RefNRE</b>	Char(15) (richiesto: si, chiave primaria: si)
<b>Avviso</b>	Varchar(1000) (richiesto: si, chiave primaria: no)
<b>Sent</b>	Tinyint(4) (richiesto: si, chiave primaria: si)

### 3.4. Controllo di accesso e sicurezza

Gli attori del sistema hanno la possibilità di accedere alle diverse funzioni offerte dal software, che prevedere un controllo di accesso, eseguito quando si effettua l'autenticazione.

### 3.5. Condizioni Boundary

L'unico vincolo presente nel software è il seguente:

Il server MySQL, nel caso in cui debba essere consultato, deve essere avviato prima dei client, in quanto è quest'ultimo a fornirgli i servizi atti a soddisfare le richieste degli utenti; in caso contrario, invece, viene mostrata una schermata, che segnala all'utente un errore di collegamento con il server e, quindi, impedisce l'utente a proseguire nell'uso del software, fino al ristabilimento della connessione.

# **Object Design Document**

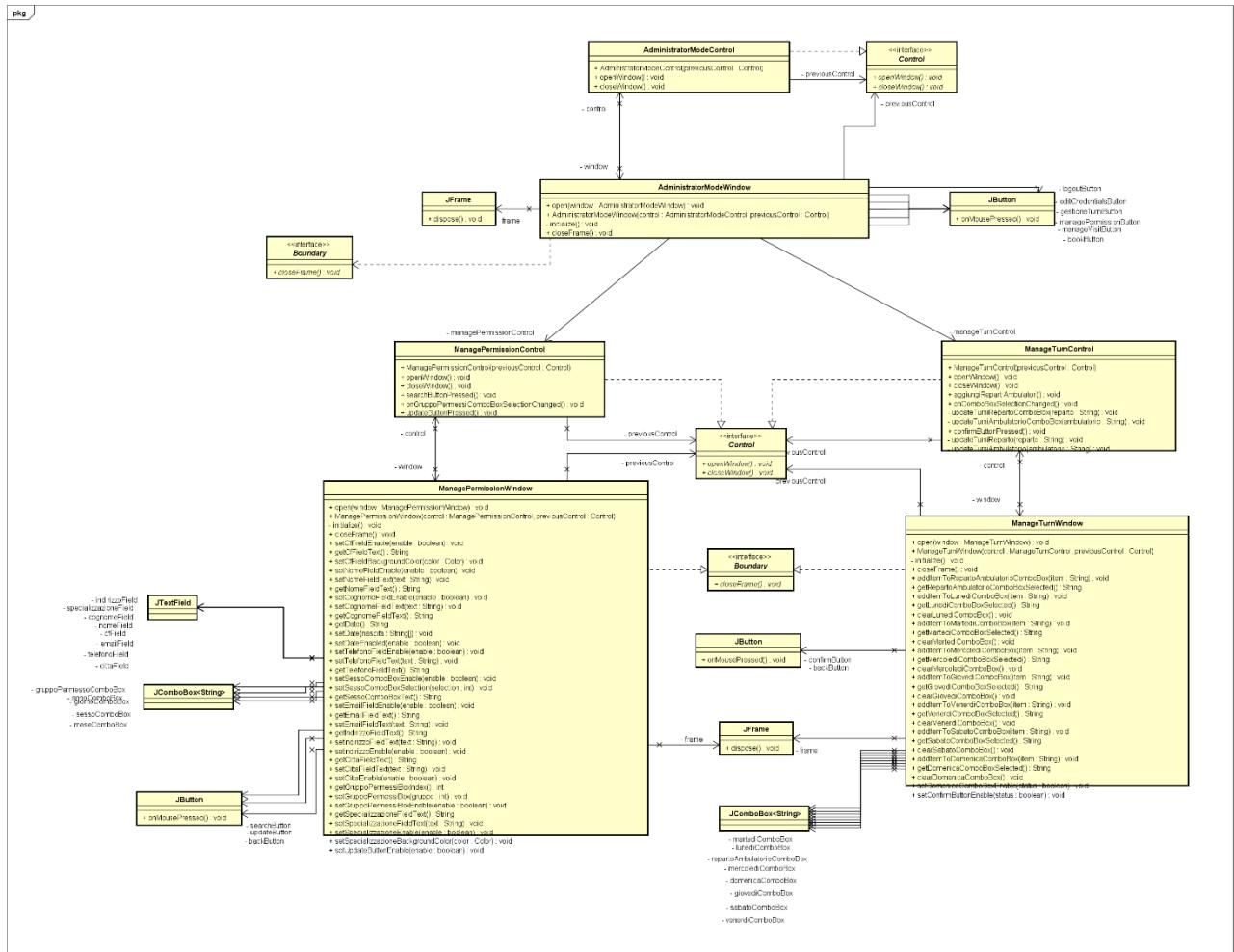
# **Indice**

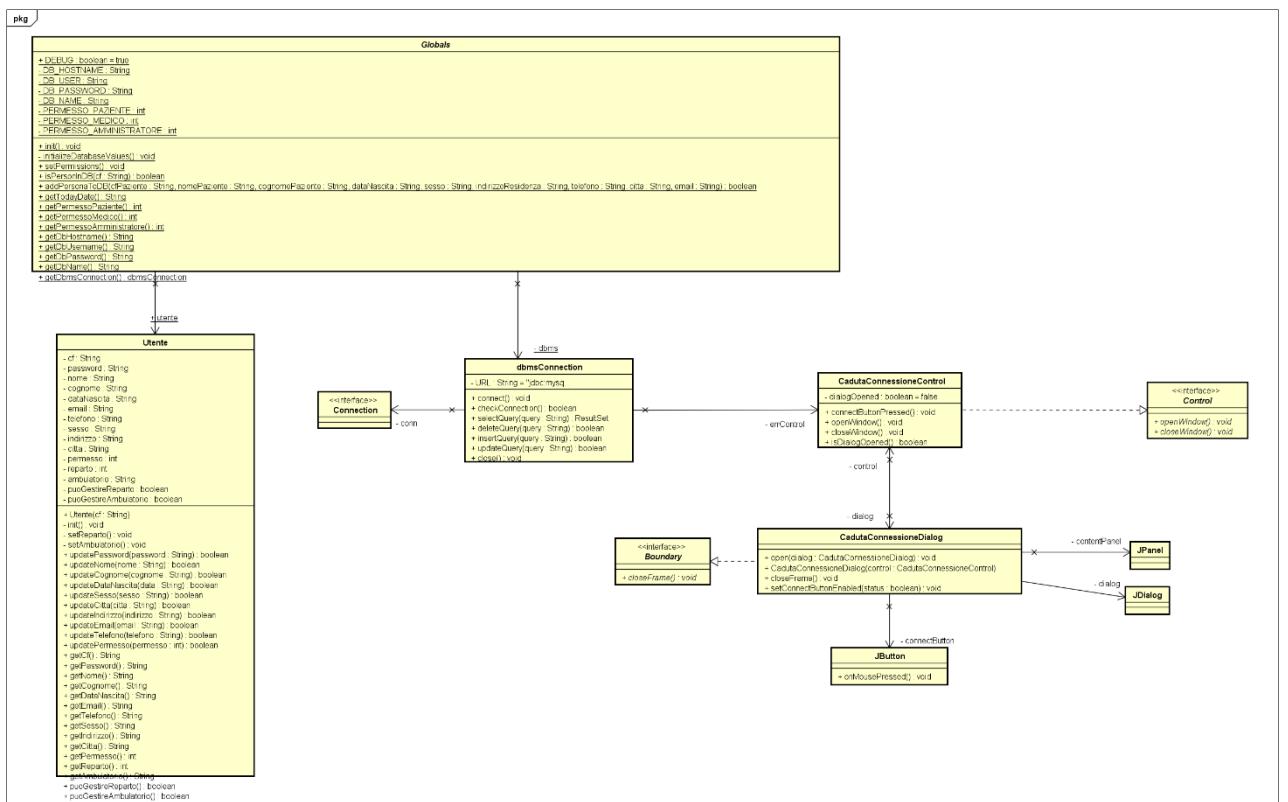
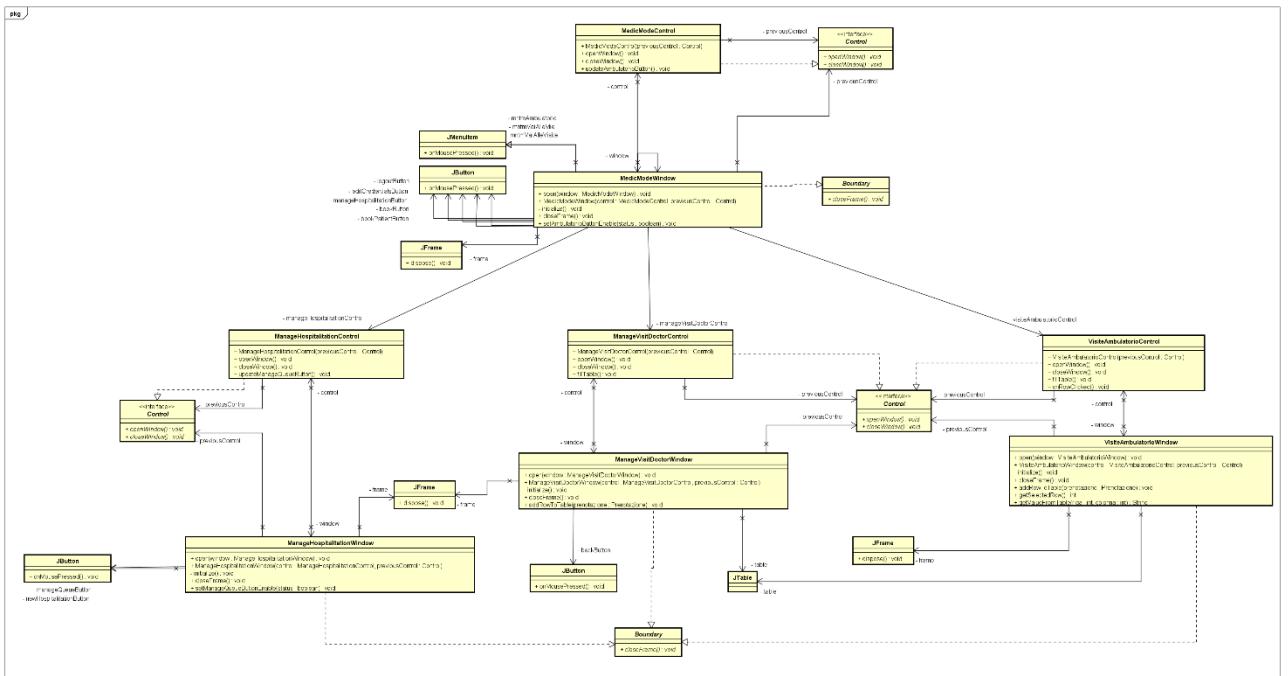
- 1. Introduzione**
- 2. Diagrammi delle classi**
- 3. Procedura di installazione**

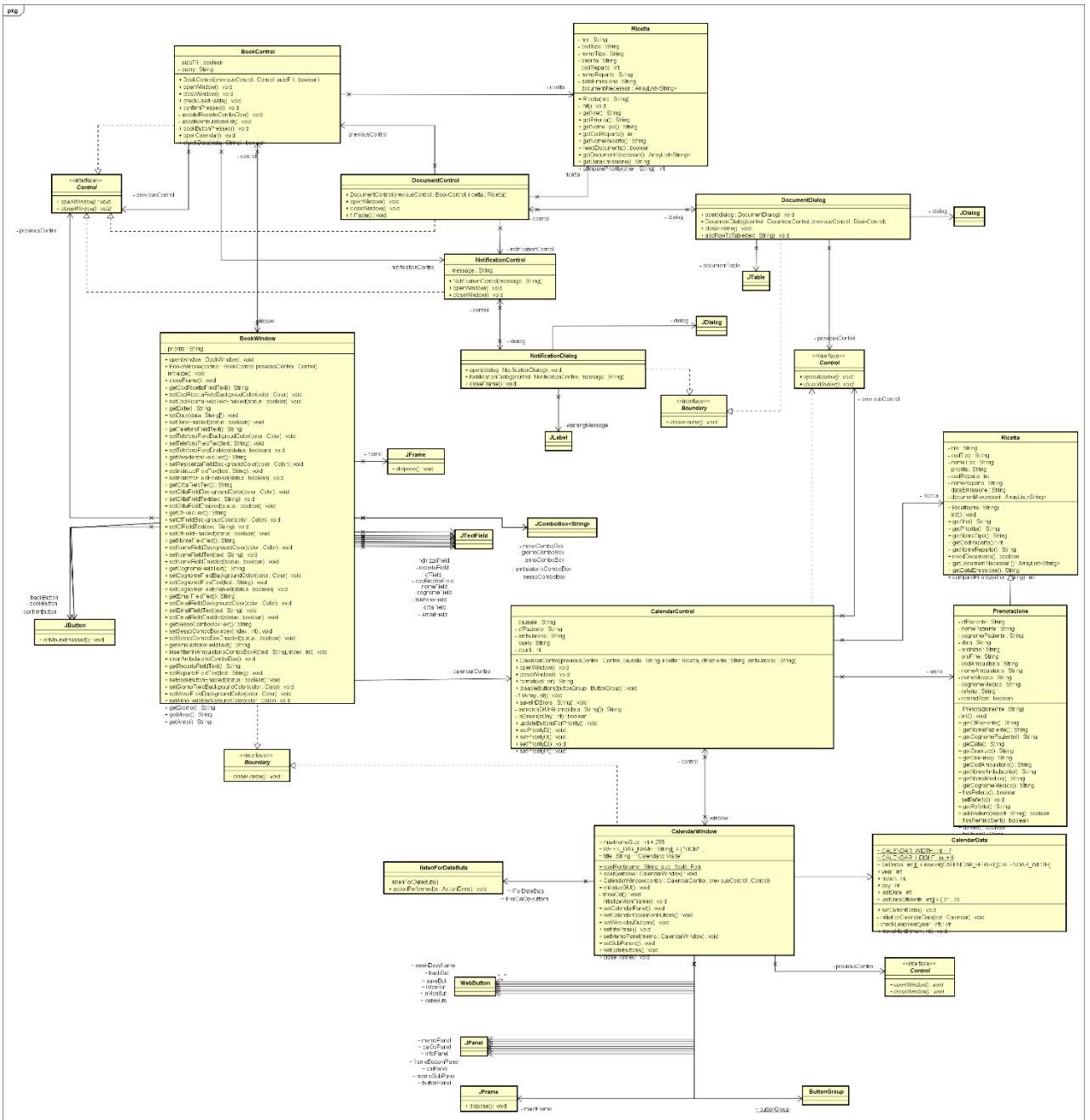
# 1. Introduzione

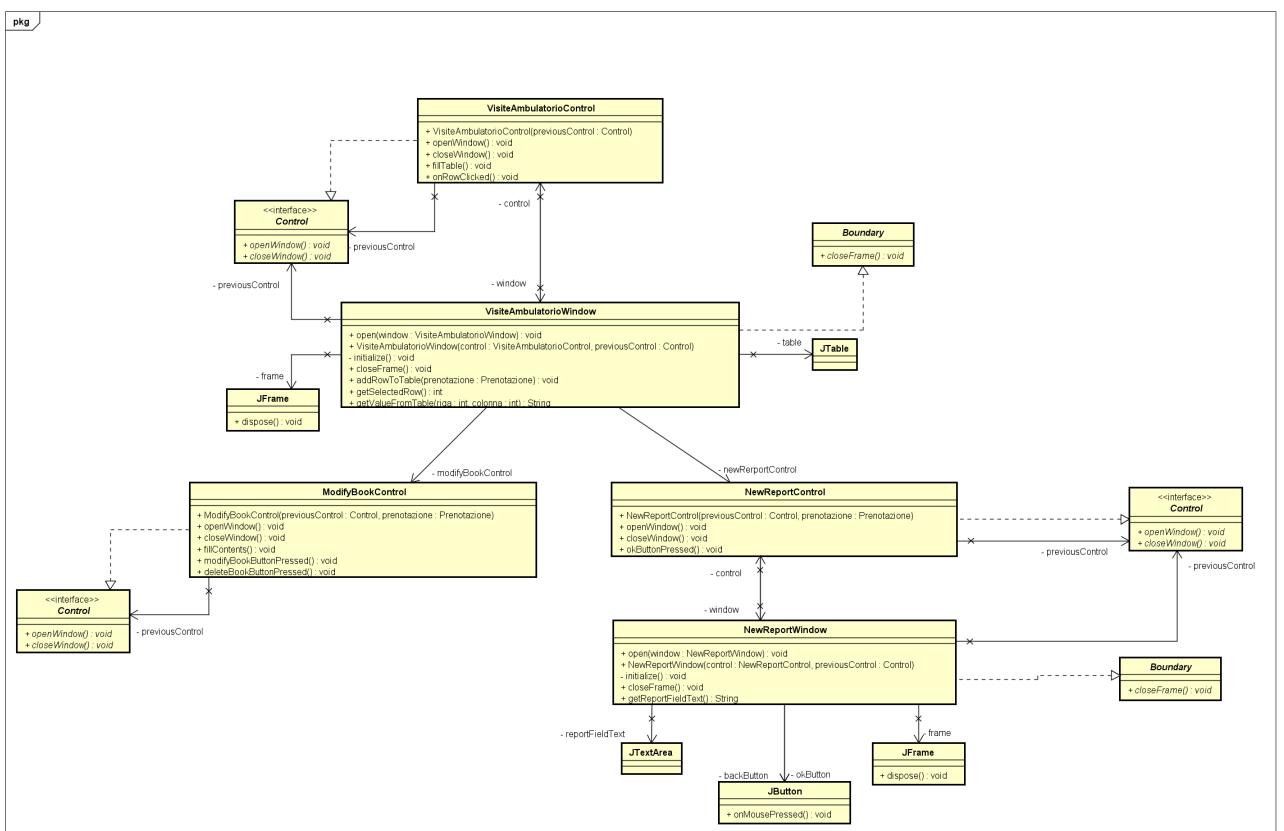
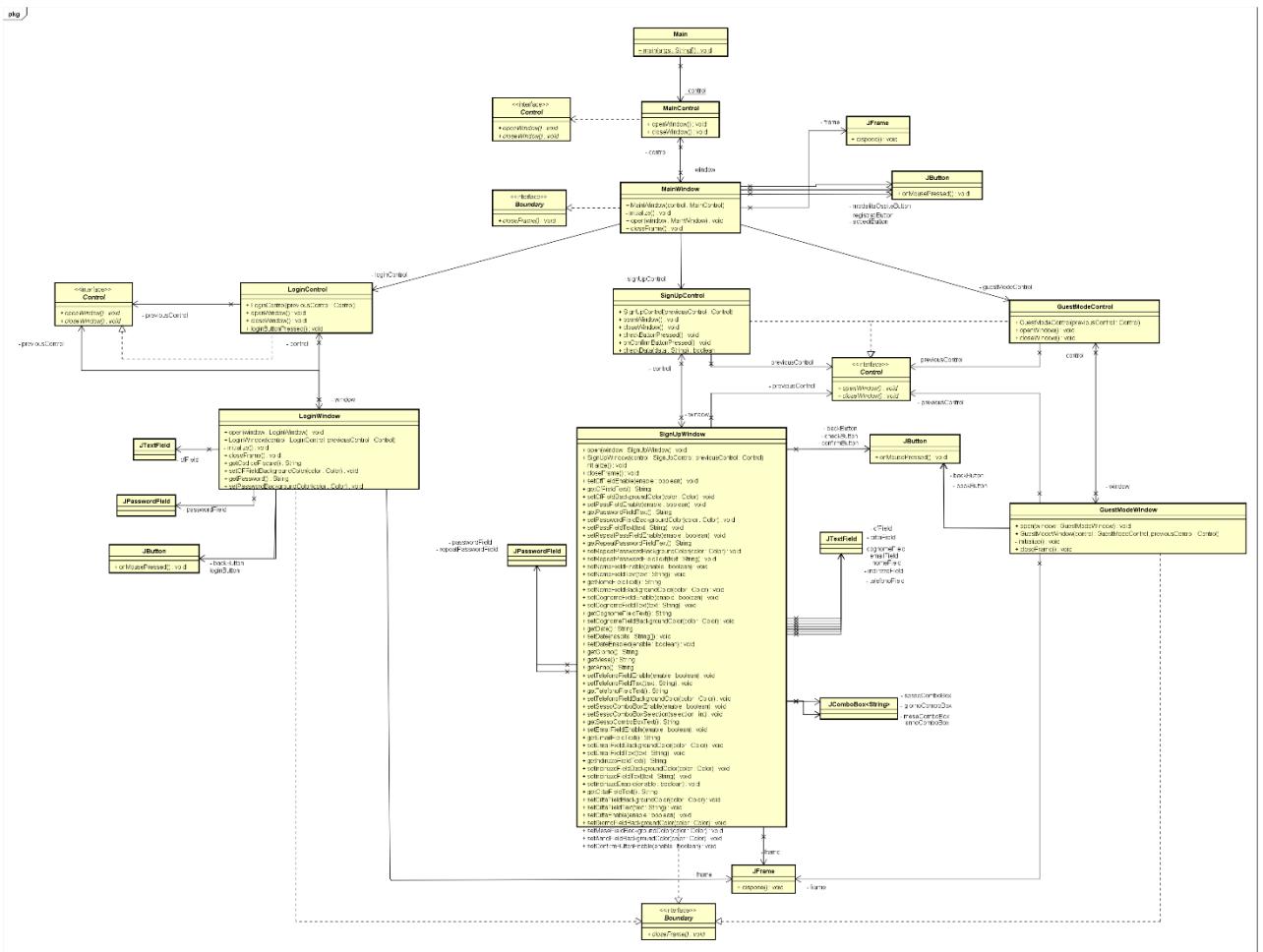
In questo documento vengono riportati i diagrammi delle classi.

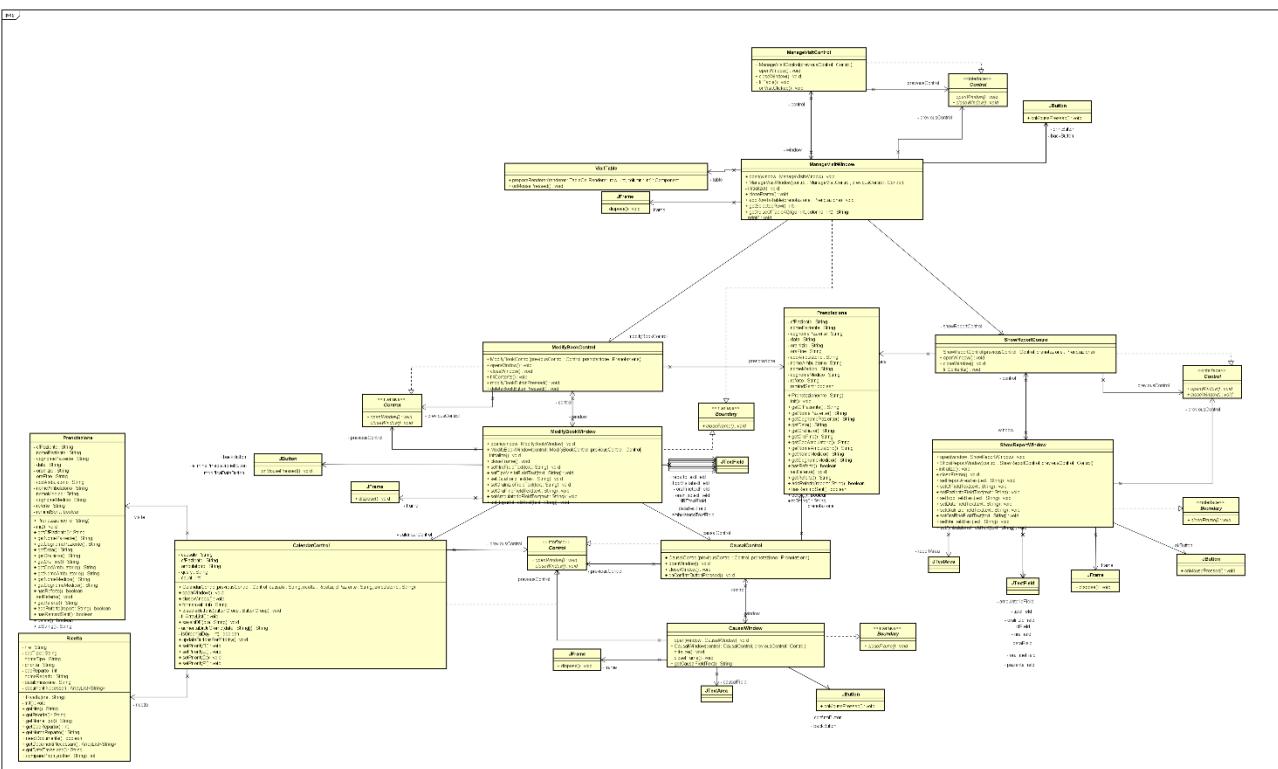
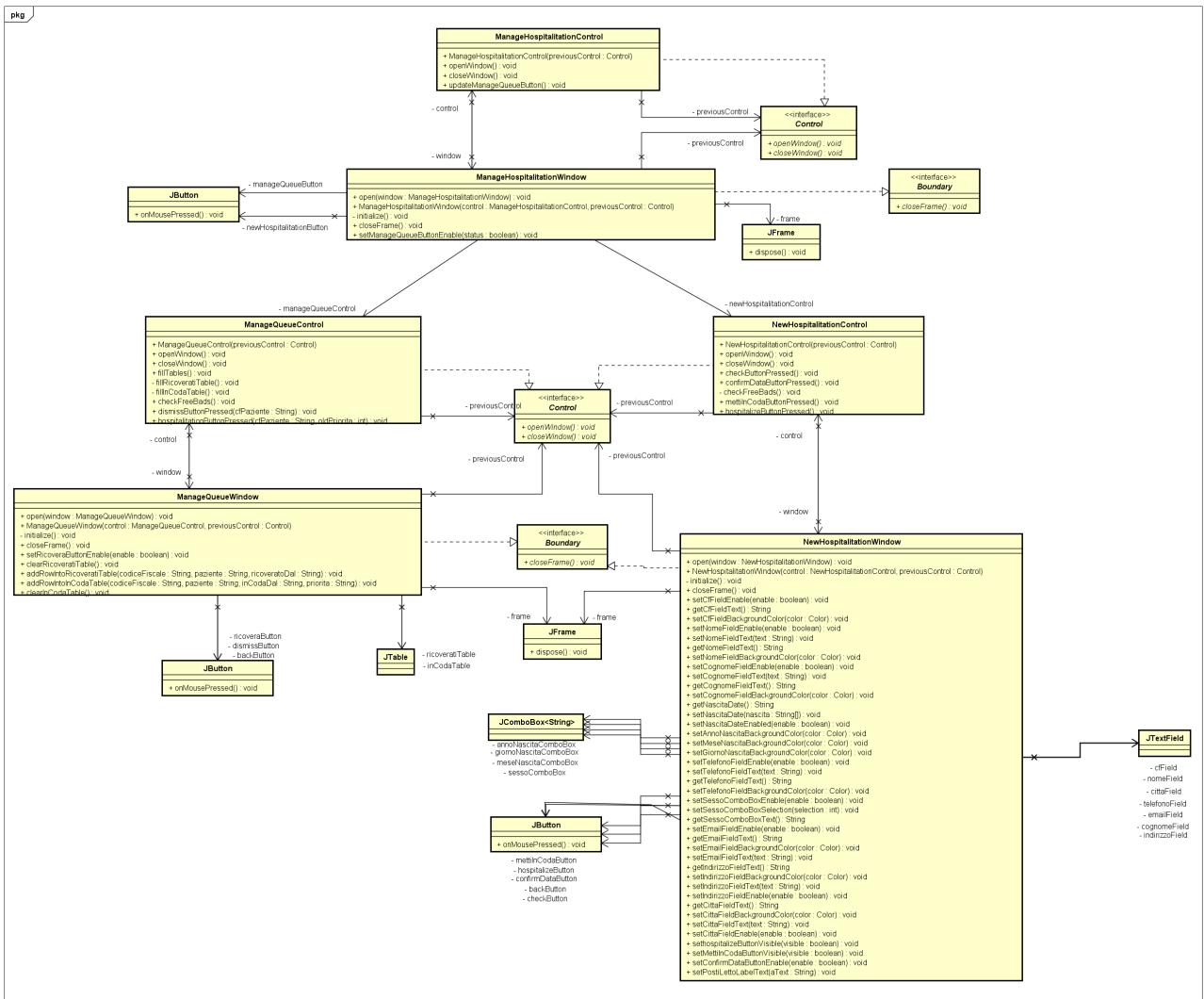
## 2. Diagrammi delle classi

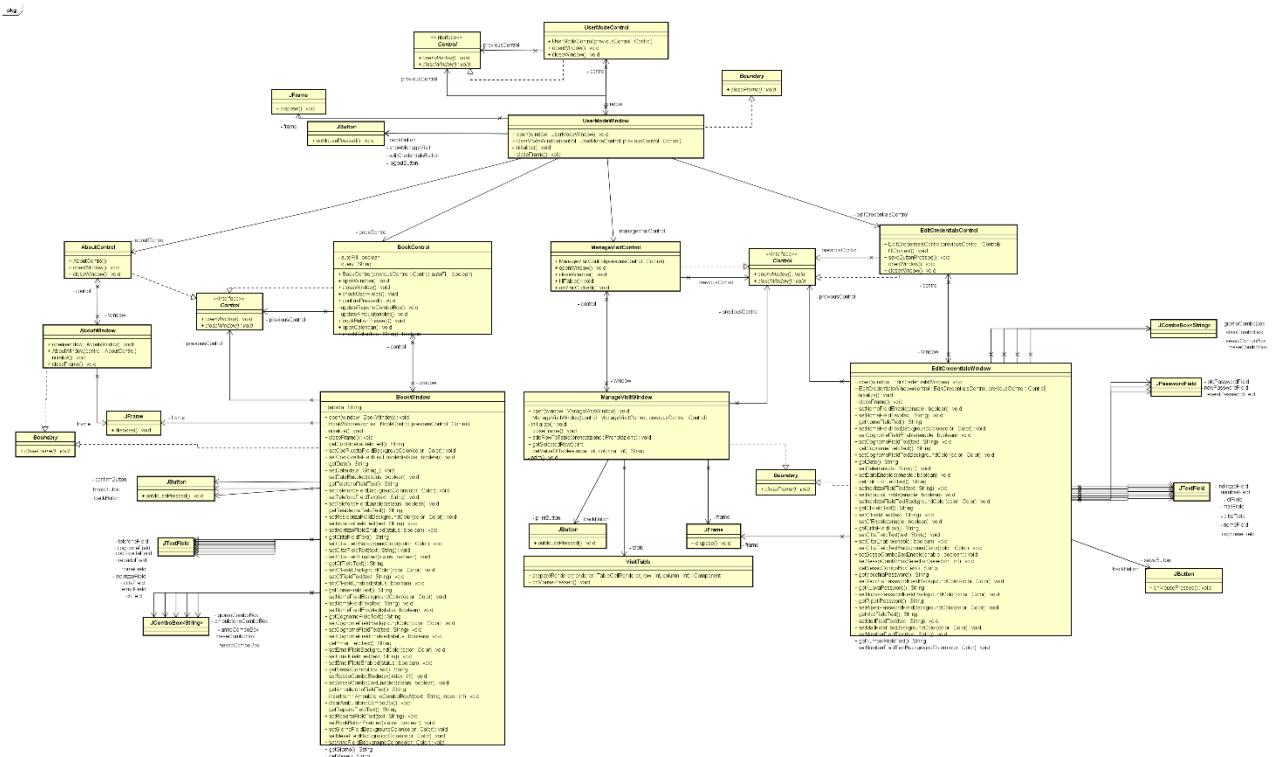
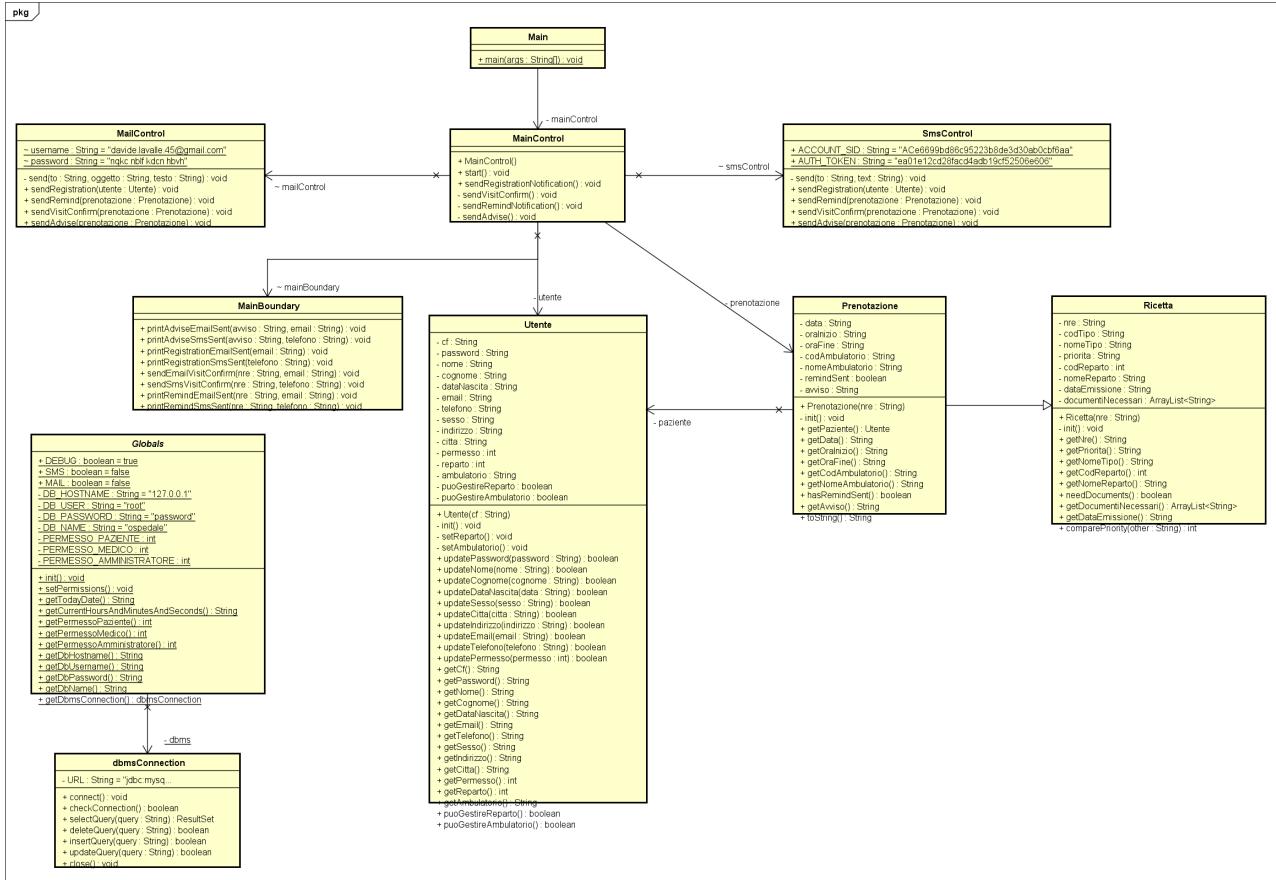












### 3. Procedura di installazione

La procedura da seguire per l'installazione del software è così composta:

- a) Installazione di "MySQL Server" e importazione dei file SQL per la creazione del DB (solo nel Server)
- b) Installazione del servizio di invio E-Mail/SMS (solo nel Server)
- c) Installazione del software (solo nel client)

#### 3.1. Installazione di MySQL Server e importazione dei file SQL per la creazione del DB (Solo nel server)

Prerequisiti:

MySQL Server for macOS, reperibile presso il seguente URL:

<https://dev.mysql.com/downloads/file/?id=484914>

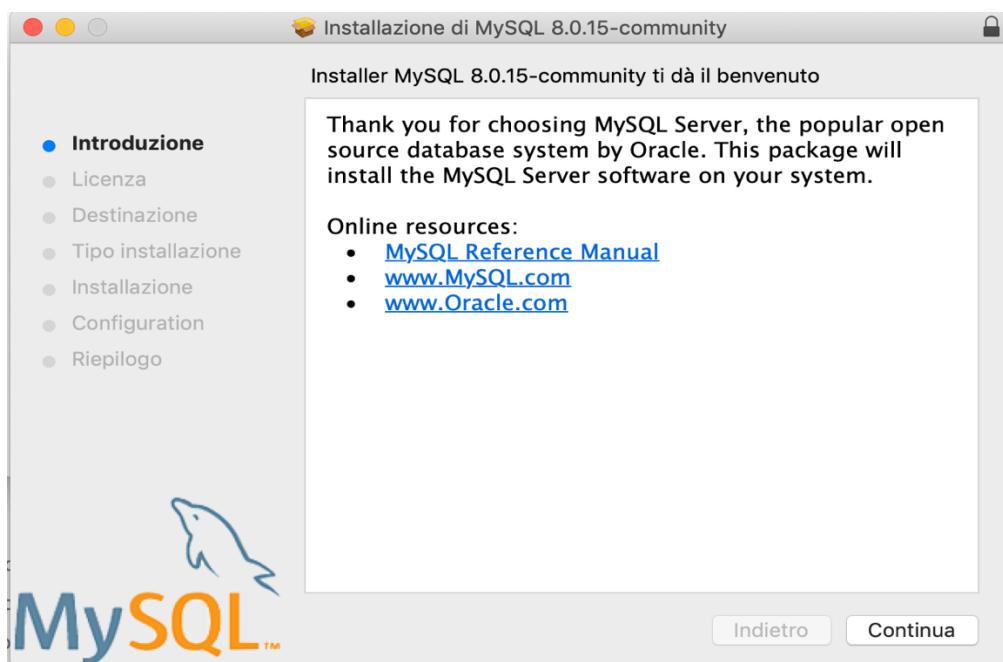
MySQL Workbench for macOS, reperibile presso il seguente URL:

<https://dev.mysql.com/downloads/file/?id=484391>

##### 3.1.1. Installazione di MySQL Server

Si procede con l'apertura del file dmg, scaricato dal precedente link.

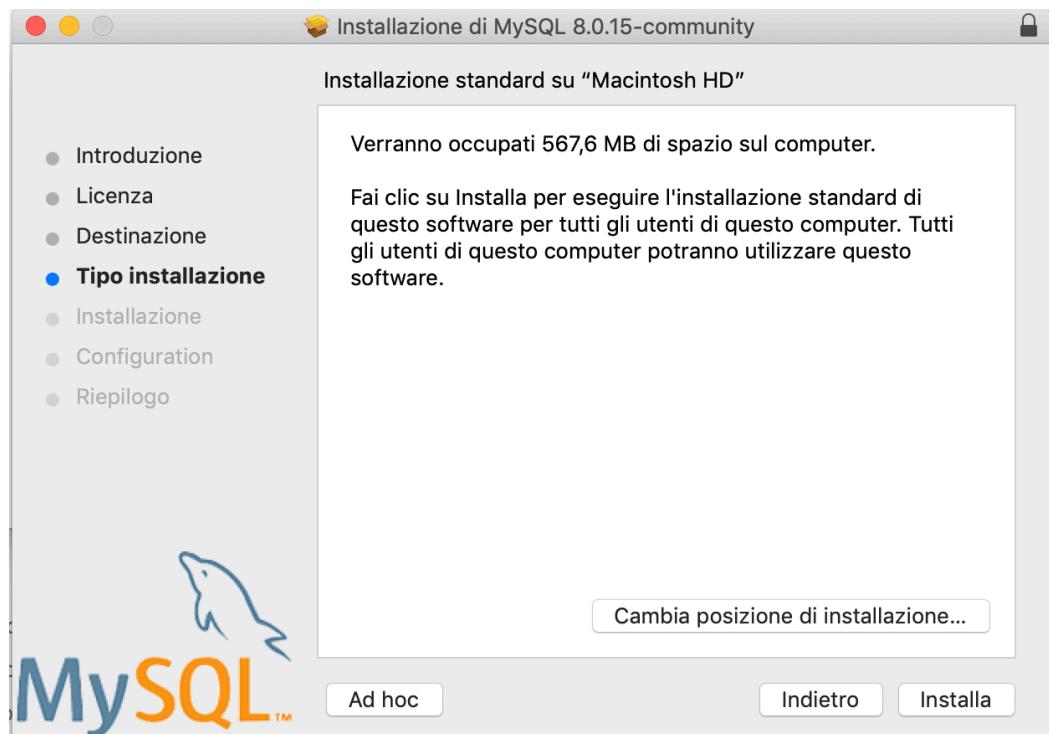
L'interfaccia che si presenterà è la seguente:



Dopo aver cliccato su “Continua” e nuovamente su “Continua”, l’interfaccia che si presenterà è la seguente:



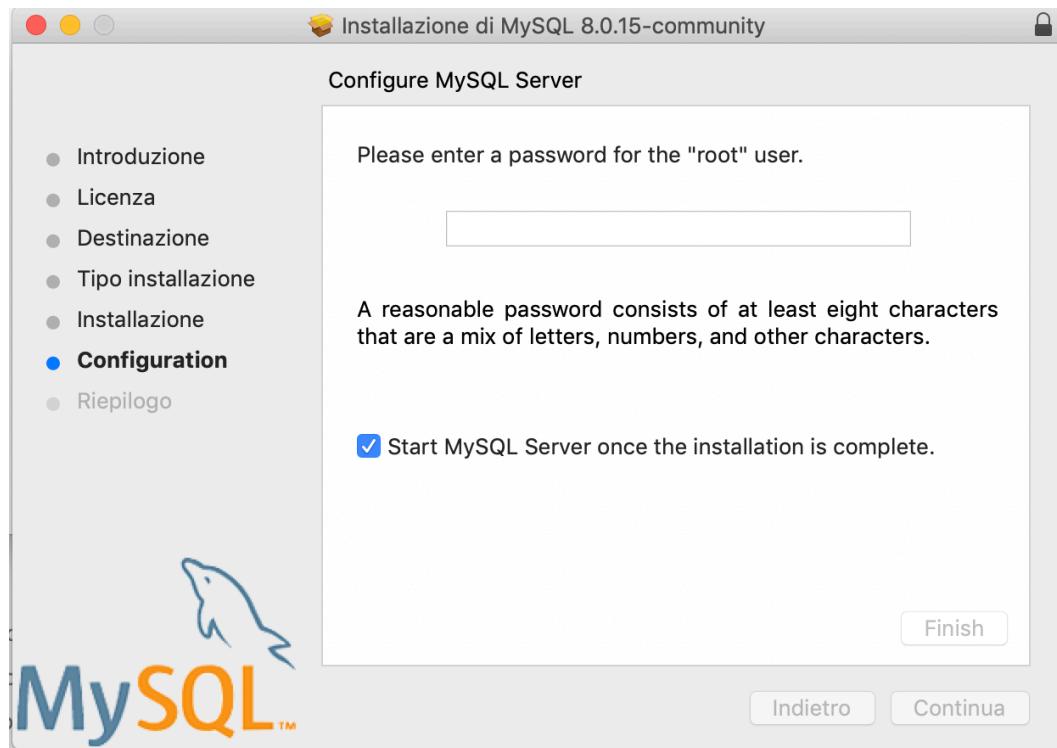
Cliccare su “Accetta”, quindi su “Installa”:



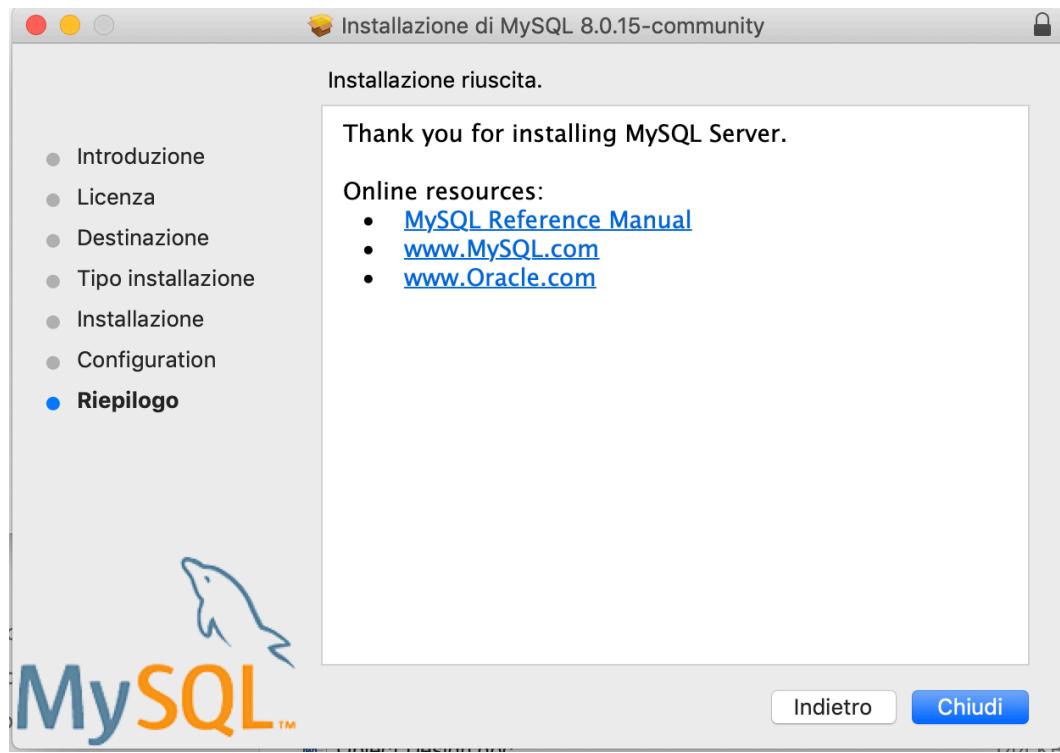
Selezionare la voce “Use Strong Password Encryption”, come da immagine, e premere su “Next”:



Nella schermata seguente, bisognerà inserire la password per l’account “root”, scegliendola arbitrariamente:

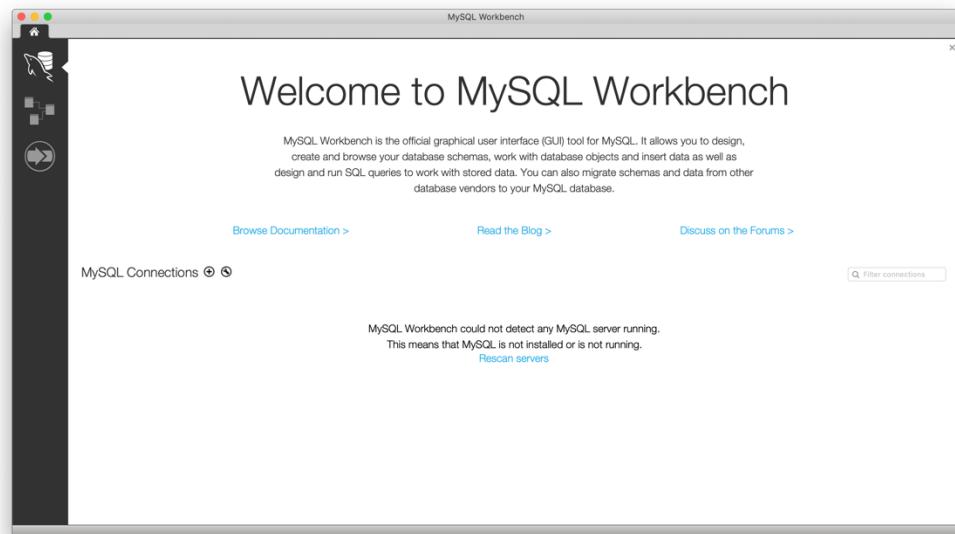


Una volta scritta la password, premere su “Continua” ed infine su “Chiudi”:

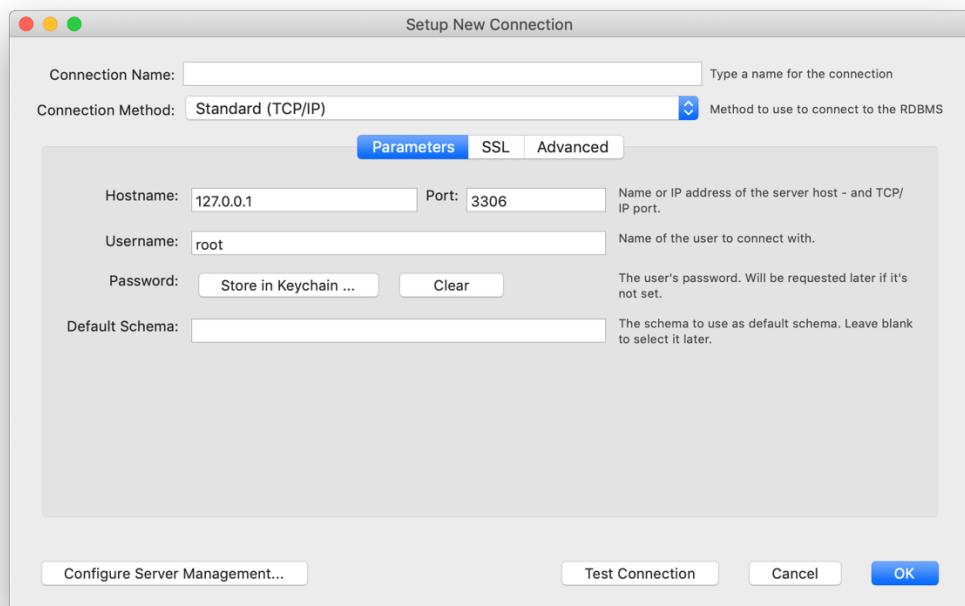


### 3.1.2. Importazione dei file SQL per la creazione del DB

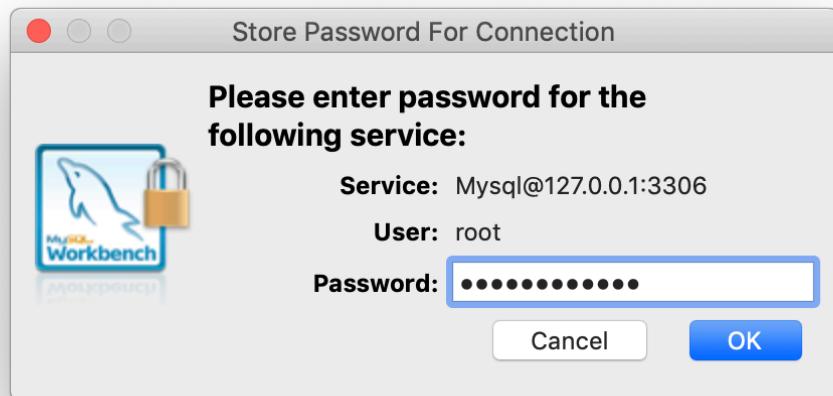
Dopo aver installato “MySQL Workbench”, bisogna aprirlo e l’interfaccia che si presenterà è la seguente:



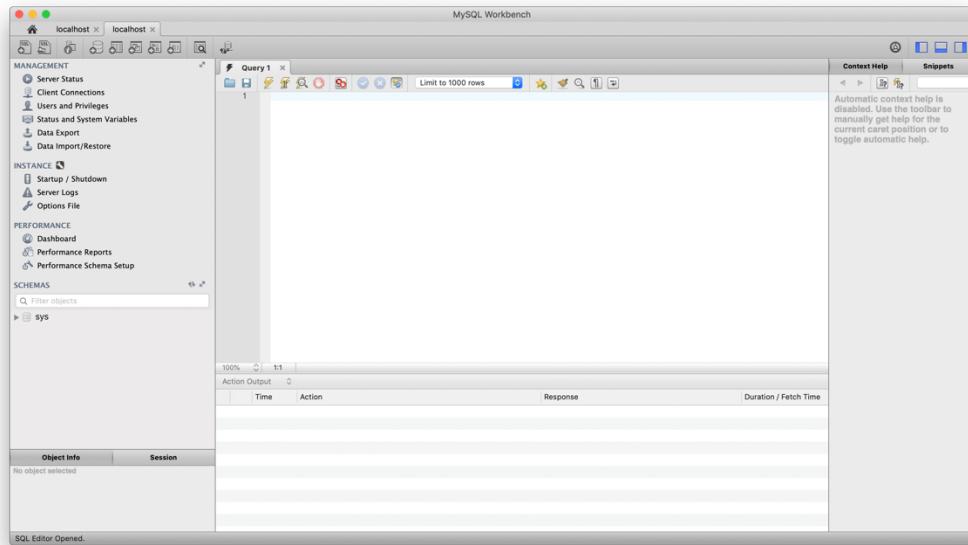
Premere sul “+” vicino la voce “MySQL Connections” e l’interfaccia che si presenterà è la seguente:



Riempire il campo “Connection Name” arbitrariamente e cliccare su “Store in keychain”:



Inserire la password usata per l'account root creato durante la fase d'installazione di "MySQL Server" e cliccare su "Ok"; quindi cliccare nuovamente su "Ok". A questo punto, comparirà la connessione appena creata sotto la voce "MySQL Connections"; cliccare su essa e l'interfaccia che comparirà è la seguente:



Cliccare su "File" e successivamente sulla voce "Run SQL Script", quindi selezionare il file "CreateDB.sql". Dopo aver cliccato su "Open", cliccare su "Run". Ripetere la procedura importando il file "InsertDBServer.sql". Una volta che la procedura è completa, cliccare su "Close".

### 3.2. Installazione del servizio di invio E-Mail/SMS (solo nel Server).

Decomprimere l'archivio "Server" e aprire il file "database.txt".

Modificare la terza riga con la password dell'account creato durante la fase d'installazione di "MySQL Server".

Avviare il file "Server.jar" da terminale, mediante il seguente comando:

"Java -jar Server.jar"

### 3.3. Installazione del programma (solo nel client).

Decomprimere l'archivio "Ospedale" e aprire il file "database.txt".

Modificare la prima riga con l'indirizzo IP o l'hostname dell'host, in cui si trova "MySQL Server". Avviare il file "Ospedale.jar", facendo doppio click su esso.

Tesina di Ingegneria del Software, anno accademico 2018/2019

Gruppo D.S.P.D.

Formato da: Davide Lavalle, Giuseppe Peri, Salvatore Pisciotta, Domenico Catalano