Fai una copia di questa sotto:

|  |
| --- |
| **CASO D’USO:** …… |
| **ATTORI:**  ……………………………….. |
| **PRECONDIZIONE:**  **…….** |
| **SEQUENZA DEGLI EVENTI:**   1. …. 2. …. 3. … 4. … |
| **SEQUENZA ALTERNATIVA:** (se presente)   1. …. 2. …. 3. … |
| **PUNTO DI ESTENSIONE:**  **……** |
| **POSTCONDIZIONE:**  **…..** |

**Entity:**

**Boundary:**

**Control:**

|  |
| --- |
| **CASO D’USO: CHECK DATI** |
| **ATTORI:**  Utente, DBMS |
| **PRECONDIZIONE:**  L’utente deve essere autenticato |
| **SEQUENZA DEGLI EVENTI:**   1. Il caso d’uso inizia quando l’utente, dopo aver compilato il campo del codice fiscale clicca sul tasto “CHECK” 2. Il sistema legge il codice fiscale inserito e lo manda al DBMS   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.  2.2 Altrimenti il sistema compila automaticamente gli appositi campi |
| **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 |

**Entity:** Utente, DBMS

**Boundary:** checkButton, submitButton, NotificationWindow

**Control:** dbmsConnection, NotificationControl

|  |
| --- |
| **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 DBMS verifica la correttezza dei dati.    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. Se i dati sono corretti e l’utente ha i permessi da “Paziente” viene visualizzata una schermata con i tasti “Prenota” e “Gestione Visite”.    1. Altrimenti, se l’utente ha i permessi da “Medico” è presente inoltre il tasto “Nuovo Ricovero”.    2. Altrimenti, se l’utente ha i permessi da “Amministratore” è presente solamente il tasto “Gestione Utenti”. |
| **PUNTO DI ESTENSIONE:**  Nel punto 4, se cade la comunicazione il sistema invoca il caso d’uso “CadutaConnessione” |

**Entity:** Utente

**Boundary:** accediButton, loginButton, LoginWindow, UserModeWindow, MedicModeWindow, AdministratorModeWindow, NotificatonWindow

**Control:** dbmsConnection, LoginControl, UserModeControl, MedicModeControl, AdministratorModeControl, NotificationControl

|  |
| --- |
| **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”.    1. Il sistema verifica se i dati rispettano i vari campi del form.    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. |

**Entity:** Utente

**Boundary:** SignUpWindow, registratiButton, checkButton, submitButton, NotificationWindow

**Control:** SignUpControl, dbmsConnection, NotificationControl

|  |
| --- |
| **CASO D’USO:** INVIO E-MAIL E SMS DI CONFERMA REGITRAZIONE |
| **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. |
| **POSTCONDIZIONE:** |

|  |
| --- |
| **CASO D’USO:** GESTIONE VISITE |
| **ATTORI:**  DBMS, utente |
| **PRECONDIZIONE:**  L’utente è autenticato |
| **SEQUENZA DEGLI EVENTI:**   1. Il caso d'uso inizia quando l'utente non è un medico e preme il pulsante "Gestione Visite". 2. Il sistema chiede al DBMS le prenotazioni effettuate dall'utente. 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 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. 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 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. |

**Entity:** Utente

**Boundary:** manageVisitButton, ManageVisitWindow, visitTable, printButton, NotificationWindow

**Control:** ManageVisitControl, dbmsConnection, NotificationControl

|  |
| --- |
| **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. |

**Entity:** Utente

**Boundary:** printButton, ManageVisitWindow

**Control:** ManageVisitControl

|  |
| --- |
| **CASO D’USO:** CADUTA CONNESSIONE |
| **ATTORI:**  DBMS |
| **PRECONDIZIONE:**  Il sistema non riesce a connettersi al DMBS |
| **SEQUENZA DEGLI EVENTI:**   1. Il caso d’uso inizia quando il DBMS non risponde all’interazione con il sistema. 2. Il sistema mostra a video la finestra “Caduta Connessione” mostrando un messaggio di assenza di connessione e un tasto “Connettiti”. 3. L’utente preme il tasto “Connettiti” e il tasto si disabilita. 4. Se il sistema riesce a connettersi al DBMS, la finestra si chiude da solo altrimenti il tasto torna attivo. |
| **POSTCONDIZIONE:**  Il sistema riprende dal punto in cui si era interrotto. |

**Entity:**

**Boundary:** connectButton, CadutaConnessioneWindow

**Control:** dbmsConnection, CadutaConnessioneControl

|  |
| --- |
| **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 Cancella. 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 3. In caso di caduta connessione si rimanda al caso d’uso Caduta Connessione |
| **POSTCONDIZIONE:**  Il sistema torna alla schermata “Gestione Visite”. |

**Entity:** Utente

**Boundary:** deleteButton, ManageVisitWindow

**Control:** dbmsConnection, ManageVisitControl

|  |
| --- |
| **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.    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”. |

**Entity:** Utente

**Boundary:** editButton, confirmButton, confirmMsg, ManageVisitWindow

**Control:** dbmsConnection, ManageVisitControl

|  |
| --- |
| **CASO D’USO:** INVIO E-MAIL E SMS DI REMIND |
| **ATTORI:**  Tempo, DBMS |
| **PRECONDIZIONE:**  L’utente ha una prenotazione |
| **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 contente le seguenti informazioni:    1. Data e ora della prenotazione.    2. Nome dell'ambulatorio in cui avverrà la visita.    3. Eventuali documenti da portare alla visita. |
| **PUNTO DI ESTENSIONE:**  Nel punto 2 in caso di caduta connessione chiama il caso d’uso Caduta Connessione |
| **POSTCONDIZIONE:** |

|  |
| --- |
| **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:    1. Data e ora della prenotazione.    2. Nome dell'ambulatorio e del reparto in cui avverrà la visita |
| **POSTCONDIZIONE:** |

**Entity:** Tempo, emailSender, smsSender

**Boundary:**

**Control:** dbmsConnection, EmailControl, SmsControl

|  |
| --- |
| **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:   1. Nuovo data e ora della prenotazione.   2. Motivo per il quale la prenotazione è stata modificata. |

**Entity:**

**Boundary:**

**Control:**

|  |
| --- |
| **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. 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.    1. Se la ricetta non esiste il sistema genera una finestra di errore.    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:    1. Il sistema mostra in verde le date fino a 2 giorni dopo l’emissione della ricetta. 9. Se la visita ha priorità P:    1. Il sistema mostra in verde le date fino a 3 mesi dopo l’emissione della ricetta. 10. Se la visita ha priorità B:     1. Il sistema mostra in verde le date fino a 10 giorni dopo l’emissione della ricetta. 11. Se la visita ha priorità D:     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.    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:    1. Il sistema mostra in verde le date fino a 2 giorni dopo la richiesta del medico. 7. Se la visita ha priorità P:    1. Il sistema mostra in verde le date fino a 3 mesi dopo la richiesta del medico. 8. Se la visita ha priorità B:    1. Il sistema mostra in verde le date fino a 10 giorni dopo la richiesta del medico. 9. Se la visita ha priorità D:    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. |

**Entity:** Utente

**Boundary:** bookButton, BookWindow, showCalendarButton, NotificationWindow, CalendarWindow

**Control:** dbmsConnection, BookControl, CalendarControl, NotificationControl

|  |
| --- |
| **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:    1. Il sistema mostra disponibili le date fino a 2 giorni dopo la richiesta del medico. 10. Se la visita ha priorità P:     1. Il sistema mostra disponibili le date fino a 3 mesi dopo la richiesta del medico. 11. Se la visita ha priorità B:     1. Il sistema mostra disponibili le date fino a 10 giorni dopo la richiesta del medico. 12. Se la visita ha priorità D:     1. Il sistema mostra disponibili le date fino a 60 giorni dopo la richiesta del medico. 13. Viene chiamato il caso d’uso Seleziona Data ultimato 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. |

**Entity:** Utente

**Boundary:** MedicModeWindow, scheduleAsPazientButton, scheduleDoctorForm, checkButton, CalendarWindow

**Control:** dbmsConnection, MedicModeControl, scheduleControl, CalendarControl

|  |
| --- |
| **CASO D’USO:** CONTROLLO PREREQUISITI PRENOTAZIONE |
| **ATTORI:**  DBMS, Utente |
| **SEQUENZA DEGLI EVENTI:**   1. Il sistema contatta il DBMS per vedere se per la visita da prenotare serve avere della documentazione aggiuntiva. 2. Il DBMS comunica al sistema la documentazione che serve. 3. Se sono necessari dei documenti, il sistema mostra a video una finestra con l’elenco dei documenti 4. Se l’utente conferma di averli, preme sul tasto SI e prosegue il caso d’uso. 5. Se l’utente preme sul pulsante NO, non sarà possibile prenotare |
| **PUNTO DI ESTENSIONE:**  Nel punto 2, se cade la comunicazione col DBMS il sistema invoca il caso d’uso “CadutaConnessione”. |

**Entity:** Utente

**Boundary:** DocumentWindow, CalendarWindow

**Control:** dbmsConnection, DocumentControl, CalendarControl

|  |
| --- |
| **CASO D’USO:** SELEZIONA DATA |
| **ATTORI:**  Utente, DBMS |
| **PRECONDIZIONE:**  Il sistema mostra a video il calendario. |
| **SEQUENZA DEGLI EVENTI:**   1. Il caso d’uso inizia quando l’utente sceglie una data disponibile tra quelle presenti. 2. Il sistema contatta il DBMS per controllare gli orari disponibili per quella data. 3. Il sistema mostra gli orari disponibili per la data selezionata. 4. Gli orari delle visite già prenotate con priorità inferiore non vengono mostrate occupate. 5. L’utente seleziona un orario e preme sul pulsante “Salva” 6. Il sistema contatta il DBMS per verificare se per quella data e ora è già prenotata una visita con priorità inferiore.    1. Se il sistema ottiene dei risultati dal DBMS, il sistema cerca il primo appuntamento libero possibile.    2. Il sistema contatta il DBMS per spostare la prenotazione con priorità inferiore.    3. Viene invocato il caso d’uso “Invio E-Mail SMS di avviso”. 7. Il sistema contatta il DBMS per salvare la visita prenotata dall’utente. |
| **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. |

**Entity:** Utente

**Boundary:** CalendarWindow, bookButton

**Control:** dbmsConnection, CalendarControl

|  |
| --- |
| **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”.    1. L’utente preme il tasto “Ricovera”    2. Il sistema manda al DBMS i dati del paziente e la data di occupazione del posto letto. 6. Se ci sono posti letti disponibili viene mostrato a video il bottone “Metti in coda”.    1. L’utente preme il tasto “Metti in coda”    2. Il sistema manda al DBMS i dati del paziente per memorizzarli tra gli utenti in coda. |
| **PUNTO DI ESTENSIONE:**  Nel punto 2, 4 e 4.1 se vi è caduta di connessione, il sistema invoca il caso d’uso “Caduta Connessione”. |
| **POSTCONDIZIONE:**  Si viene rimandati alla pagina principale. |

**Entity:** Utente

**Boundary:** newHospitalizationButton, NewHospitalizationWindow, checkButton, confirmDataButton, hospitalizeButton, mettiInCodaButton

**Control:** NewHospitalizationControl, dbmsConnection

|  |
| --- |
| **CASO D’USO:** GESTIONE CODA RICOVERI |
| **ATTORI:**  Utente, DBMS |
| **PRECONDIZIONE:**  1.L’Utente deve essere autenticato  2.L’Utente deve avere i permessi da 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.   1. Se ci sono posti letto disponibili, il sistema abilita il tasto “Ricovera”.  1. L’utente può selezionare un paziente da una di queste due liste    1. Se l’utente seleziona un paziente dalla lista dei pazienti ricoverati:       1. Il sistema abilita il tasto “DIMETTI PAZIENTE”       2. L’utente preme sul tasto “DIMETTI PAZIENTE”       3. Il sistema contatta il DBMS per eliminare il paziente dalla lista dei pazienti ricoverati.       4. Una volta ricevuta risposta dal DBMS il sistema elimina dalla lista dei pazienti ricoverati il paziente selezionato    2. Se l’utente seleziona un paziente dalla lista dei pazienti in coda:       1. L’utente preme sul tasto “RICOVERA”       2. Il sistema contatta il DBMS per eliminare dalla coda dei pazienti il paziente selezionato e aggiungerlo alla lista dei pazienti ricoverati       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. |
| **SEQUENZA ALTERNATIVA:**  1. Nei punti 3, 4.1.4. e 4.2.2, se cade la comunicazione il sistema invoca il caso d’uso “CadutaConnessione” |
| **PUNTO DI ESTENSIONE:**  CadutaConnessione |
| **POSTCONDIZIONE:** |

**Entity:** Utente, DBMS

**Boundary:** NoficationWindow, dismissButton, ricoveraButton, ManageQueueWindow

**Control:** ManageQueueControl, NotificationControl, DbmsConnection

|  |
| --- |
| **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. |

**Entity:** Utente

**Boundary:** ManageVisitWindow, visitButton, confirmButton, ReportWindow

**Control:** ManageVisitControl, ReportControl, dbmsConnection

|  |
| --- |
| **CASO D’USO:** GESTIONE PERMESSI |
| **ATTORI:**  Utente, DBMS |
| **PRECONDIZIONE:**  L’utente ha effettuato il login 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”    1. Se l’utente è un paziente, il campo Specializzazione resta vuoto. 5. L’utente sceglie tra i seguenti permessi: “Paziente, medico e amministratore”. 6. L’utente preme il bottone “Modifica dati. 7. Il sistema contatta il DBMS per aggiornare il permesso di quell’utente 8. Il sistema mostra un messaggio di conferma |
| **SEQUENZA ALTERNATIVA:** (se presente)   1. Come sopra 2. Come sopra 3. Come sopra 4. Come sopra    1. Se l’utente è un medico il campo Specializzazione viene riempito con la qualifica del medico 5. Come sopra 6. Come sopra 7. Come sopra 8. Come sopra |
| **POSTCONDIZIONE:**  L’utente loggato come amministratore ha modificato il permesso dell’utente inserito. |

Entity: Utente

Boundary: ManagePermissionWindow, searchButton, updateButton, backButton

Control: ManagePermissionControl

|  |
| --- |
| **CASO D’USO:** GESTIONE TURNI |
| **ATTORI:**  Utente, DBMS |
| **PRECONDIZIONE:**  L’utente ha effettuato il login come amministratore |
| **SEQUENZA DEGLI EVENTI:**  L’utente preme il bottone “Modifica Turni”  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.  Cliccando su un giorno viene mostrato a video il nome dei medici di competenza.  L’utente sceglie il medico per il determinato giorno della settimana.  L’utente preme il bottone Conferma quando ha effettuato le modifiche necessarie.  Il sistema contatta il DBMS per aggiornare il turno di quell’utente.  Il sistema mostra un messaggio di conferma |
| **POSTCONDIZIONE:**  L’utente ha modificato i turni del reparto selezionato. |

Entity: Utente

Boundary: confirmButton, backButton, ManageTurnWindow

Control: ManageTurnControl