



DIPARTIMENTO DI INGEGNERIA ELETTRICA
E DELLE TECNOLOGIE DELL'INFORMAZIONE

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
CORSO DI LAUREA IN INGEGNERIA INFORMATICA
CORSO DI INGEGNERIA DEL SOFTWARE
PROF. S. RUSSO - A.A. 2020 - 21

Progetto

SISTEMA GESTIONE ASL

Gruppo: CODE BLINDERS

Studenti: Alessandro Vanacore N46004309

alessa.vanacore@studenti.unina.it

Raffaele Russo N46004653

raffaele.russo45@studenti.unina.it

Versione 1 del 18/12/2020

Indice

1. REQUISITI	3
1.1 SPECIFICHE INFORMALI D'UTENTE.....	3
1.2 REQUISITI SOFTWARE	4
2. ANALISI E SPECIFICA DEI REQUISITI	9
2.1 MODELLAZIONE DEI CASI D'USO.....	9
2.2 DIAGRAMMA DELLE CLASSI	11
2.3 DIAGRAMMI DI SEQUENZA	12
3. STIMA DEI COSTI	14
4. PIANO DI TEST FUNZIONALE	16
5. PROGETTAZIONE.....	32
5.1 DIAGRAMMA DELLE CLASSI	32
5.2 DIAGRAMMI DI SEQUENZA	33
6. IMPLEMENTAZIONE	35
7. TESTING.....	38
7.1 TEST FUNZIONALE.....	38
7.2 TEST STRUTTURALE	47
7.2.1 COMPLESSITÀ CICLOMATICA.....	47
7.2.2 TEST DI UNITÀ	49
7.3 TEST DI UNITÀ CON JUNIT	51

1. Requisiti

1.1 Specifiche informali d'utente

ID 2: Sistema ASL

Si vuole realizzare una applicazione web per una ASL. Il sistema offre al cittadino la possibilità di prenotare e/o disdire una o più visite. Per prenotare una visita, il cittadino può utilizzare il servizio di prenotazione on-line o recarsi presso l'ufficio prenotazioni dell'ASL. Per accedere al servizio di prenotazione on-line, il cittadino registrato deve effettuare il login inserendo username e password. Per effettuare la registrazione, il cittadino deve fornire i dati anagrafici: cognome, nome, codice fiscale, data di nascita, luogo di nascita, luogo di residenza, numero di telefono, e-mail, username e password. Tali dati verranno memorizzati dal sistema.

La prenotazione delle visite può anche essere effettuata dal personale all'accettazione che ha il compito di prenotare le visite a nome del cittadino che si reca allo sportello. Il personale inserisce i dati anagrafici del cittadino. In particolare, il sistema dovrà consentire di effettuare ricerche nell'archivio in base al codice fiscale nel caso i dati anagrafici siano già stati inseriti; nel caso la ricerca dia esito positivo ovviamente non è necessaria una nuova immissione e la prenotazione sarà intestata ad un cittadino che ha già effettuato una prenotazione/visita in passato. Nel caso il cittadino sia un nuovo paziente dovrà essere possibile inserirlo nell'archivio. La prenotazione può essere effettuata solo previa disponibilità. Pertanto, è necessario verificare che non siano già state effettuate prenotazioni.

Di ciascuna prenotazione, si vuole conoscere l'id, lo stato (disdetta, confermata, effettuata), la data e l'ora della visita a cui si riferisce. Ciascuna visita è caratterizzata da una descrizione, numero della stanza, numero di piano e il medico che la effettua. Inoltre il sistema memorizza anche informazioni relative ai medici che operano nella struttura. In particolare, si vuole memorizzare, il nome, il cognome, la specializzazione, l'orario e giorno di visita presso la struttura. E' compito del personale dell'accettazione inserire tali dati nel sistema ed, eventualmente, aggiornarli.

Il sistema consente: (i) ai cittadini e al personale di accettazione di ricercare i medici in base alla specializzazione (es. oculisti) per visualizzare giorni e orari di visita; (ii) ai medici di visualizzare le prenotazioni confermate.

La verifica delle disponibilità di prenotazione avviene specificando il giorno, l'orario e la descrizione della visita (es. visita oculistica). Se non ci sono prenotazioni confermate, il sistema invia all'utente un messaggio di conferma prenotazione, altrimenti invia un messaggio di non disponibilità.

1.2 Requisiti software

ID 2: Sistema ASL

Si vuole realizzare una applicazione web per una ASL. Il sistema offre al cittadino la possibilità di prenotare e/o disdire una o più visite. Per prenotare una visita, il cittadino può utilizzare il servizio di prenotazione on-line o recarsi presso l'ufficio prenotazioni dell'ASL. Per accedere al servizio di prenotazione on-line, il cittadino registrato deve effettuare il login inserendo username e password.

Per registrarsi, il cittadino deve inserire i dati anagrafici: cognome, nome, codice fiscale, data di nascita, luogo di nascita, luogo di residenza, numero di telefono, e-mail, username e password. Tali dati verranno memorizzati dal sistema.

La prenotazione delle visite può anche essere effettuata dal personale all'accettazione che ha il compito di prenotare le visite a nome del cittadino che si reca allo sportello.

In particolare, il sistema dovrà consentire al personale di effettuare ricerche nell'archivio in base al codice fiscale del cittadino.

Nel caso la ricerca dia esito positivo, cioè i dati anagrafici siano già memorizzati nel sistema, non ne è necessaria una nuova immissione. La prenotazione sarà quindi intestata ad un cittadino che ha già richiesto una prenotazione in passato.

Nel caso il cittadino sia un nuovo paziente dovrà essere possibile inserirlo nell'archivio prima che possa prenotare una visita allo sportello.

La prenotazione può essere confermata solo previa disponibilità.

La struttura offre al pubblico visite mediche nelle seguenti specialità: cardiologia, dermatologia, endocrinologia, fisiatria, gastroenterologia, neurologia, oculistica, oncologia, ortopedia, ginecologia, otorinolaringoiatria.

Ciascun tipo di visita è caratterizzata da un numero di piano, un numero di stanza e il medico che la effettua (si assuma che ogni medico sia associato ad un'unica specializzazione).

Di ciascuna prenotazione, si vuole conoscere l'id, lo stato (disdetta, confermata) e la data della prenotazione.

Pertanto, la verifica delle disponibilità di prenotazione avviene specificando il giorno, l'orario e la descrizione¹ della visita (es. visita oculistica).

Se non ci sono prenotazioni confermate, il sistema mostra all'utente un messaggio di conferma prenotazione, altrimenti un messaggio di non disponibilità.

In particolare, a seguito di prenotazione confermata, il cittadino riceverà un'e-mail contenente i dettagli della visita.

Inoltre, sistema memorizza anche informazioni relative ai medici che operano nella struttura. In particolare, si vuole memorizzare, il nome, il cognome, la specializzazione, l'orario e giorno di visita presso la struttura (si assuma che il medico possa lavorare soltanto per un giorno all'interno della struttura). È compito del personale dell'accettazione inserire tali dati nel sistema ed eventualmente aggiornarli.

Il sistema consente: (i) ai cittadini e al personale di accettazione di ricercare i medici in base alla specializzazione (es. oculisti) per visualizzare giorni e orari di visita; (ii) ai medici di visualizzare le visite a suo carico.

¹ La descrizione della visita coincide con la specializzazione del medico che la effettuerà.

Analisi dei nomi e dei verbi

ID 2: Sistema ASL

Si vuole realizzare una applicazione web per una ASL. Il sistema offre al cittadino la possibilità di prenotare e/o disdire una o più visite. Per prenotare una visita, il cittadino può utilizzare il servizio di prenotazione on-line o recarsi presso l'ufficio prenotazioni dell'ASL. Per accedere al servizio di prenotazione on-line, il cittadino registrato deve effettuare il login inserendo username e password.

Per registrarsi, il cittadino deve inserire i dati anagrafici: cognome, nome, codice fiscale, data di nascita, luogo di nascita, luogo di residenza, numero di telefono, e-mail, username e password. Tali dati verranno memorizzati dal sistema.

La prenotazione delle visite può anche essere effettuata dal personale all'accettazione che ha il compito di prenotare le visite a nome del cittadino che si reca allo sportello.

In particolare, il sistema dovrà consentire al personale di effettuare ricerche nell'archivio in base al codice fiscale del cittadino.

Nel caso la ricerca dia esito positivo, cioè i dati anagrafici siano già memorizzati nel sistema, non ne è necessaria una nuova immissione. La prenotazione sarà quindi intestata ad un cittadino che ha già richiesto una prenotazione in passato.

Nel caso il cittadino sia un nuovo paziente dovrà essere possibile inserirlo nell'archivio prima che possa prenotare una visita allo sportello.

La prenotazione può essere confermata solo previa disponibilità.

La struttura offre al pubblico visite mediche nelle seguenti specialità: cardiologia, dermatologia, endocrinologia, fisiatria, gastroenterologia, neurologia, oculistica, oncologia, ortopedia, ginecologia, otorinolaringoiatria.

Ciascun tipo di visita è caratterizzata da un numero di piano, un numero di stanza e il medico che la effettua (si assuma che ogni medico sia associato ad un'unica specializzazione).

Di ciascuna prenotazione, si vuole conoscere l'id, lo stato (disdetta, confermata) e la data della prenotazione.

Pertanto, la verifica delle disponibilità di prenotazione avviene specificando il giorno, l'orario e la descrizione¹¹ della visita (es. visita oculistica).

Se non ci sono prenotazioni confermate, il sistema mostra all'utente un messaggio di conferma prenotazione, altrimenti un messaggio di non disponibilità.

In particolare, a seguito di prenotazione confermata, il cittadino riceverà un'e-mail contenente i dettagli della visita.

Inoltre, sistema memorizza anche informazioni relative ai medici che operano nella struttura. In particolare, si vuole memorizzare, il nome, il cognome, la specializzazione, l'orario e giorno di visita presso la struttura (si assuma che il medico possa lavorare soltanto per un giorno all'interno della struttura). È compito del personale dell'accettazione inserire tali dati nel sistema ed eventualmente aggiornarli.

Il sistema consente: (i) ai cittadini e al personale di accettazione di ricercare i medici in base alla specializzazione (es. oculisti) per visualizzare giorni e orari di visita; (ii) ai medici di visualizzare le visite a suo carico.

Requisiti Funzionali e Requisiti sui Dati

Si vuole realizzare una applicazione web per una ASL. Il sistema è utilizzato da cittadini, cittadini registrati, medici e impiegati all'accettazione.

Requisiti Funzionali Cittadini

- I cittadini si registrano al sistema
- I cittadini ricercano i medici in base alla specializzazione

Requisiti Funzionali Cittadini Registrati

- I cittadini registrati prenotano una visita medica. Il sistema in seguito alla richiesta deve verificarne la disponibilità e in caso di esito positivo inviare una e-mail di conferma al cittadino
- I cittadini registrati disdicono una visita medica

Requisiti Funzionali Impiegati all'Accettazione

- Gli impiegati inseriscono i dati dei medici
- Gli impiegati aggiornano i dati dei medici
- Gli impiegati ricercano i medici in base alla specializzazione
- Gli impiegati ricercano i cittadini allo sportello per codice fiscale
- Gli impiegati assegnano una prenotazione ai cittadini allo sportello. Il sistema in seguito alla richiesta deve verificarne la disponibilità e in caso di esito positivo inviare una e-mail di conferma al cittadino
- Gli impiegati immettono i dati dei cittadini nel sistema

Requisiti Funzionali Medici

- I medici visualizzano le visite a loro carico

Requisiti sui Dati

- Ciascun cittadino è identificato da un cognome, nome, codice fiscale, data di nascita, luogo di nascita, luogo di residenza, numero di telefono e un'e-mail
- Ciascun cittadino è identificato da un cognome, nome, codice fiscale, data di nascita, luogo di nascita, luogo di residenza, numero di telefono, e-mail, username e password
- Ciascuna prenotazione è identificata da un id e una data
- Ciascuna visita è identificata da un orario, una data, un numero di stanza e di piano
- Ciascun medico è identificato da il nome, il cognome, la specializzazione, l'orario e giorno di visita presso la struttura

Attori Primari:

Cittadino

Cittadino Registrato

Impiegato all'accettazione

Medico

Attori Secondari:

ServerEmail

Casi d'Uso:

PrenotaVisita

VisualizzaVisite

AssegnaPrenotazione

DisdiciVisita

RicercaMedico

Registrazione

AggiornaDatiMedico

InserisciDatiMedico

Casi d'Uso di inclusione:

VerificaDisponibilitàPrenotazione

RicercaDatiCittadino

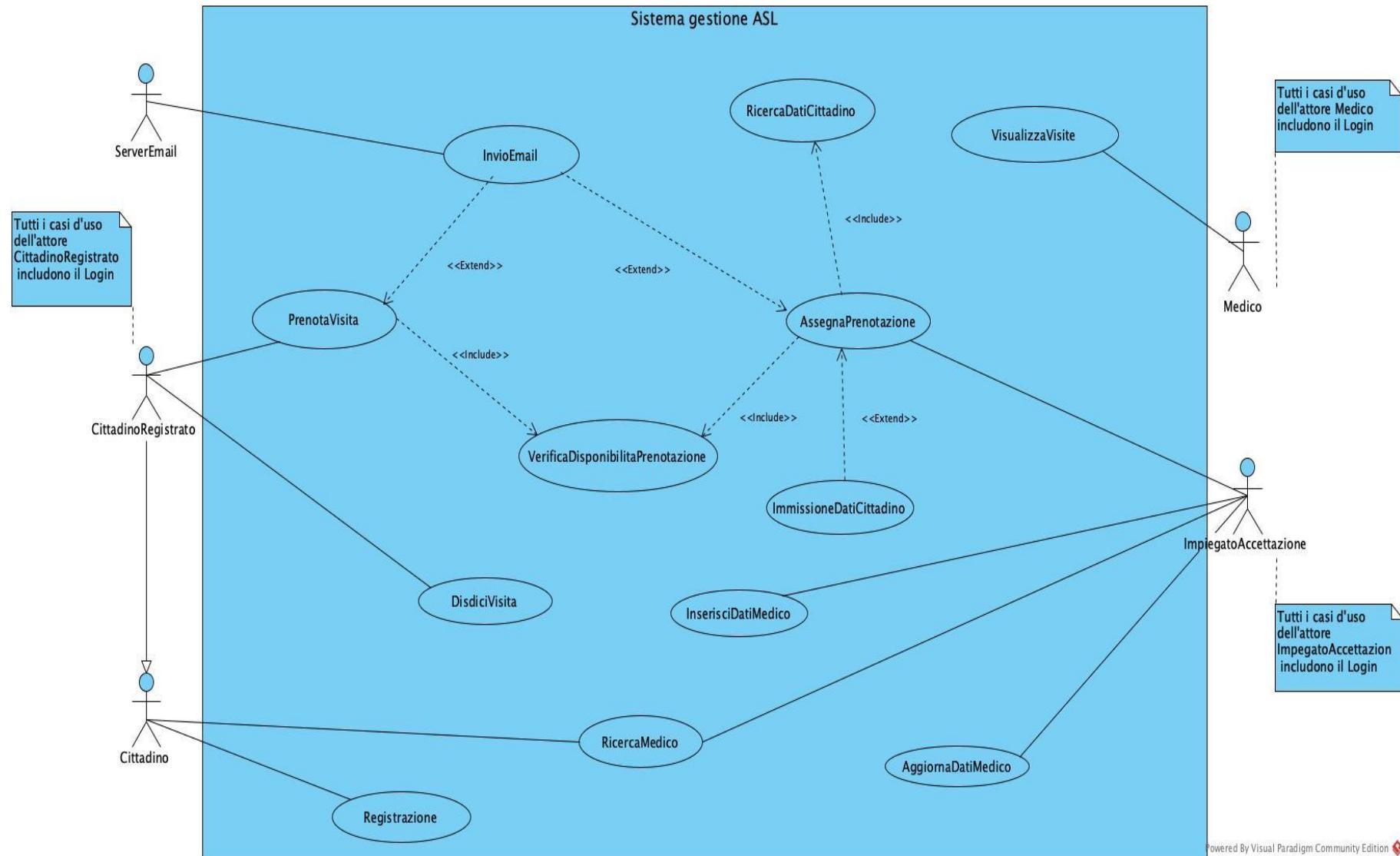
Casi d'Uso d'estensione:

InvioEmail

ImmissioneDatiCittadino

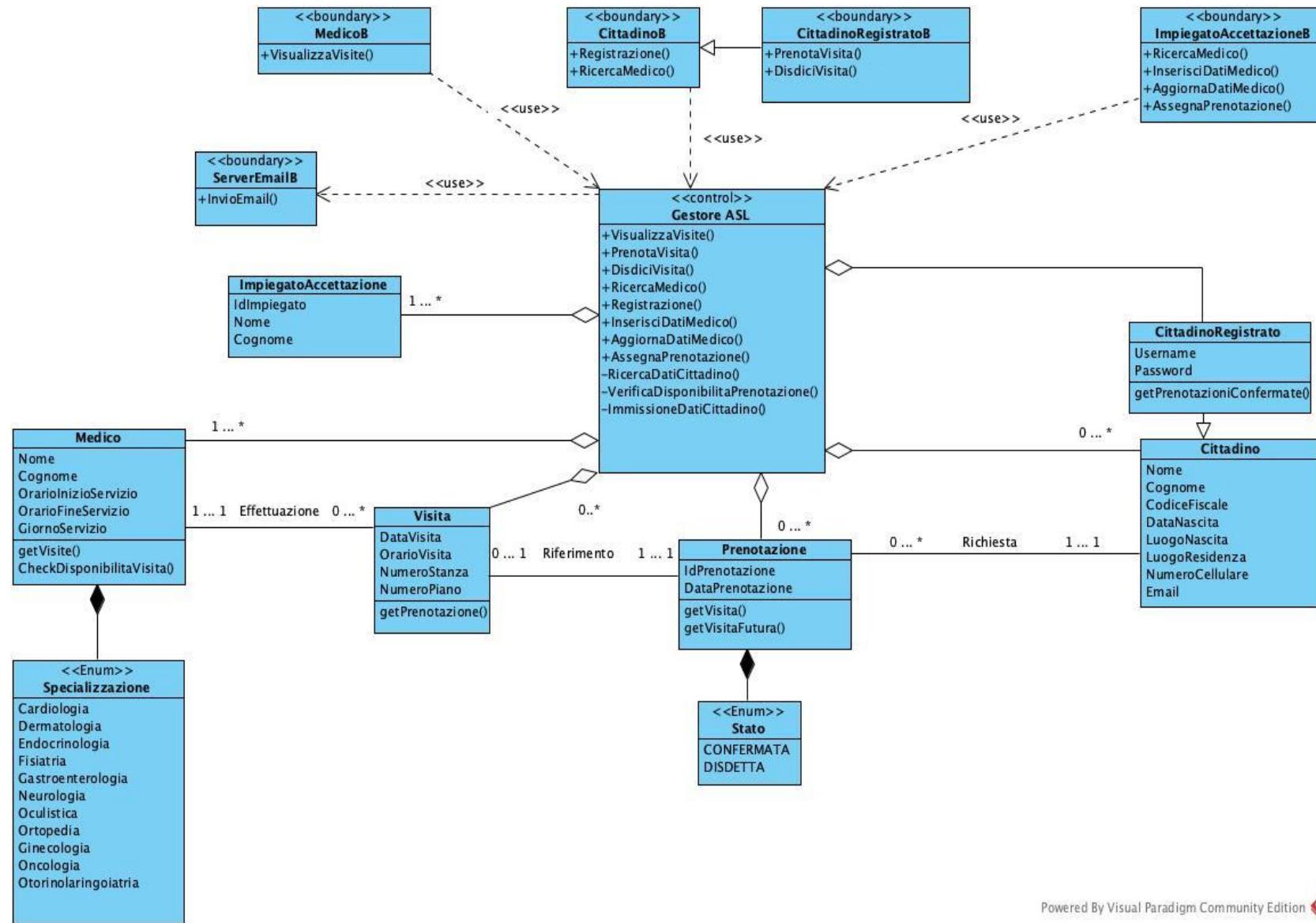
2. Analisi e specifica dei requisiti

2.1 Modellazione dei casi d'uso



Caso d'uso: Registrazione	Caso d'uso: PrenotaVisita	Caso d'uso: VerificaDisponibilitàPrenotazione	Caso d'uso: InvioEmail
<p>ID: 1</p> <p>Breve descrizione: Il cittadino si registra all'applicazione web dell'ASL</p> <p>Attori primari: Cittadino</p> <p>Attori secondari: Nessuno</p> <p>Precondizioni: Email e username non utilizzati</p> <p>Sequenza degli eventi principali: 1. Il caso d'uso inizia quando il cittadino vuole usufruire per la prima volta del sistema di prenotazione online 2. Il cittadino fornisce i suoi dati anagrafici, numero di cellulare, email, username e password 3. Il sistema comunica al cittadino che la registrazione è avvenuta con successo </p> <p>Postcondizioni: Il sistema memorizza i dati del cittadino</p>	<p>ID: 2</p> <p>Breve descrizione: Il cittadino registrato prenota una visita</p> <p>Attori primari: CittadinoRegistrato</p> <p>Attori secondari: Nessuno</p> <p>Precondizioni: Il cittadino registrato ha effettuato il login</p> <p>Sequenza degli eventi principali: 1. Il cittadino fornisce la data, l'orario e la descrizione della visita da prenotare 2. Se l'esito della verifica è positivo 2.1. Il sistema comunica al cittadino che la prenotazione è avvenuta con successo 2.2. Extend InvioEmail 3. Altrimenti 3.1. Il sistema comunica al cittadino che la richiesta di prenotazione è stata rifiutata </p> <p>Postcondizioni: Il sistema memorizza le informazioni dell'eventuale prenotazione confermata e della visita riferita</p>	<p>ID: 8</p> <p>Breve descrizione: Il sistema verifica la disponibilità di una prenotazione richiesta</p> <p>Attori primari: CittadinoRegistrato</p> <p>Attori secondari: Nessuno</p> <p>Precondizioni: Il cittadino registrato ha fornito il giorno, l'orario e la descrizione della visita da prenotare</p> <p>Sequenza degli eventi principali: 1. Il sistema verifica la disponibilità della visita richiesta </p> <p>Postcondizioni: Il sistema memorizza l'esito della verifica</p>	<p>ID: 13</p> <p>Breve descrizione: Il sistema inoltra al ServerEmail una richiesta di invio email</p> <p>Attori primari: CittadinoRegistrato</p> <p>Attori secondari: ServerEmail</p> <p>Precondizioni: L'esito di VerificaDisponibilitàPrenotazione è positivo</p> <p>Sequenza degli eventi principali: 1. Il sistema fornisce al ServerEmail l'indirizzo email del cittadino e il relativo promemoria della prenotazione confermata </p> <p>Postcondizioni: Il cittadino riceve l'email di conferma</p>

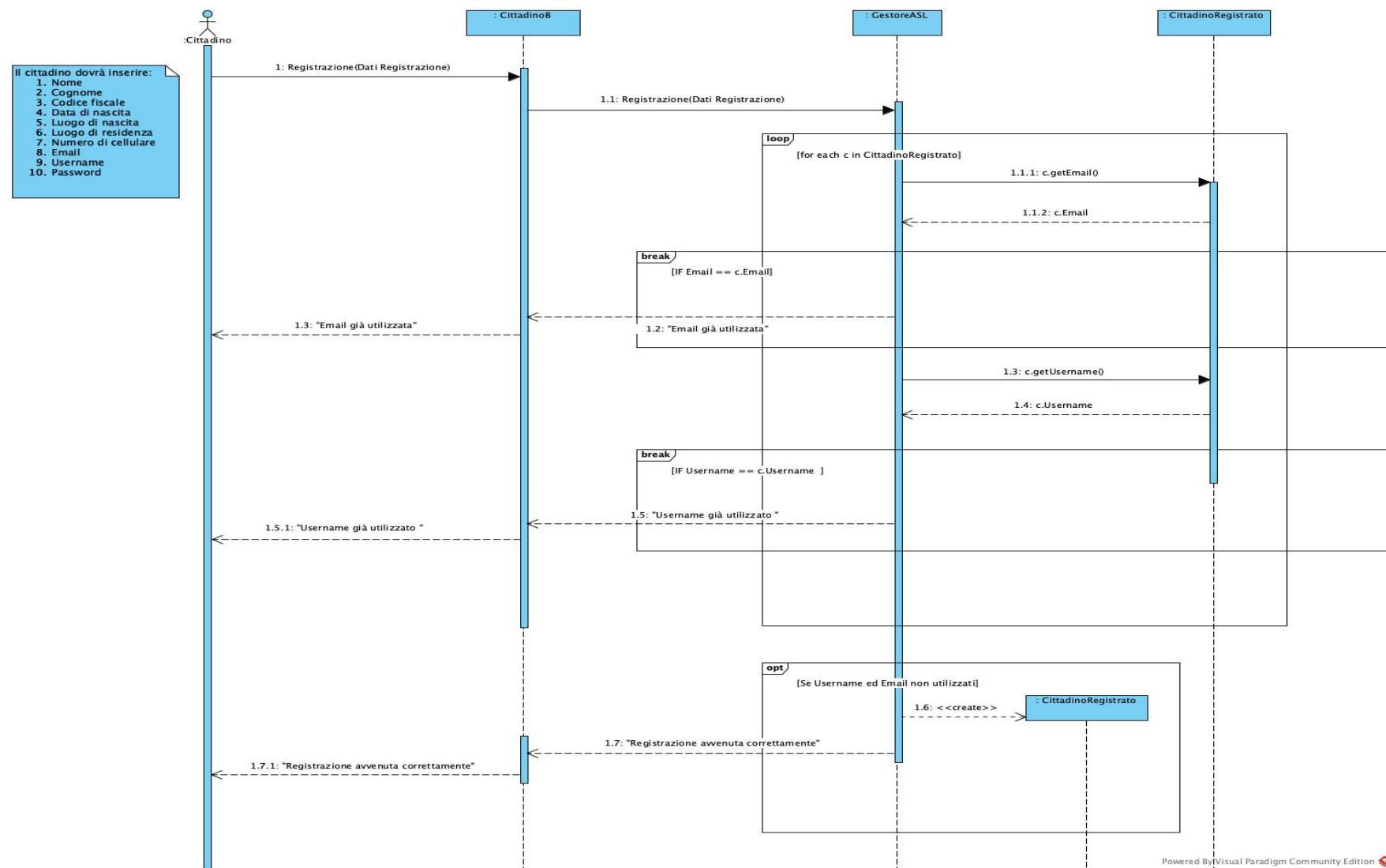
2.2 Diagramma delle classi



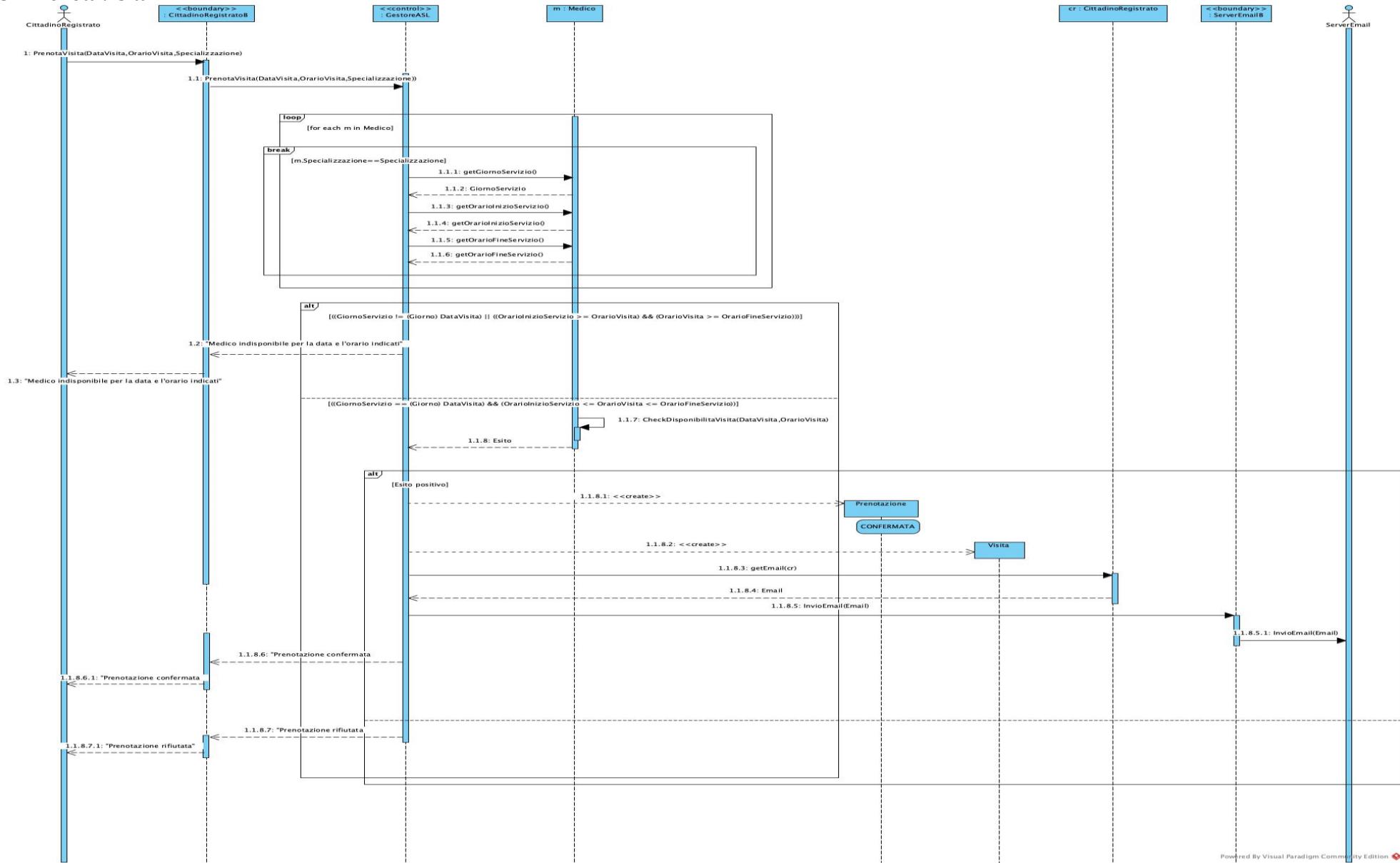
Powered By Visual Paradigm Community Edition

2.3 Diagrammi di sequenza

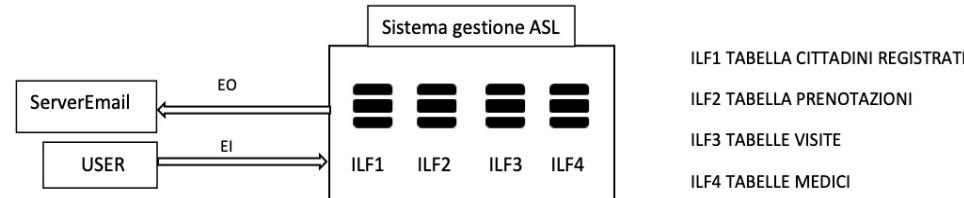
SD Registrazione



SD PrenotaVisita



3. Stima dei costi



INDICE	PESO
ILF1	7
ILF2	7
ILF3	7
ILF4	7
EI	6
EO	5

Conteggio per sviluppo di progetto

Complessità ILF1 TABELLA PRENOTAZIONI DET = 2 RET = 1

Complessità ILF2 TABELLA CITTADINI REGISTRATI DET = 10 RET = 1

Complessità ILF3 TABELLA VISITE DET = 4 RET = 1

Complessità ILF4 TABELLA MEDICI DET = 5 RET = 1

Complessità EI DET = 5 (3 campi input + messaggio conferma + comando utente) FTR = 4

Complessità EO DET = 6 (4 campi visita+ messaggio + informazioni testuali) FTR = 2

UFP = 39

CARATTERISTICHE GENERALI	VALORE
COMUNICAZIONE DATI	4
DISTRIBUZIONE DELL' ELABORAZIONE	2
PRESTAZIONI	3
FACILITA' DI GESTIONE OPERATIVA	3
UTILIZZO INTENSIVO CONFIGURAZIONE	2
FREQUENZA DELLE TRANSAZIONI	1
INSERIMENTO DATI INTERATTIVO	5
EFFICIENZA PER L'UTENTE FINALE	3
AGGIORNAMENTO INTERATTIVO	4
COMPLESSITA' ELABORATIVA	2
RIUSABILITA'	4
FACILITA' DI INSTALLAZIONE	0
MOLTEPLICITA' DI SITI	0
FACILITA' DI MODIFICA	2

$$FP = UFP \times (0.65 + 0.01 \times 35) = 39 \times 1 = 39$$

Il numero di linee di codice stimato se l'applicazione è sviluppato in JAVA è pari a :

$$FP \times LLOC/FP = 39 \times 53 = 2067$$

Function Point Analysis :Registrazione



INDICE	PESO
ILF	7
EI	3

Conteggio per sviluppo di progetto

Complessità ILF TABELLA CITTADINI REGISTRATI DET = 10 RET = 1

EI FTR = 1 DET = 11 (10 campi registrazione + 1 messaggio di conferma o errore)

UFP = 10

CARATTERISTICHE GENERALI	VALORE
COMUNICAZIONE DATI	4
DISTRIBUZIONE DELL' ELABORAZIONE	2
PRESTAZIONI	3
FACILITA DI GESTIONE OPERATIVA	3
UTILIZZO INTENSIVO CONFIGURAZIONE	2
FREQUENZA DELLE TRANSAZIONI	1
INSERIMENTO DATI INTERATTIVO	5
EFFICIENZA PER L'UTENTE FINALE	3
AGGIORNAMENTO INTERATTIVO	4
COMPLESSITA' ELABORATIVA	2
RIUSABILITA'	4
FACILITA' DI INSTALLAZIONE	0
MOLTEPLICITA' DI SITI	0
FACILITA' DI MODIFICA	2

$$FP = UFP \times (0.65 + 0.01 \times 35) = 10 \times 1 = 10$$

Il numero di linee di codice stimato se l'applicazione è sviluppato in JAVA è pari a :

$$FP \times LLOC/FP = 10 \times 53 = 530$$

4. Piano di test funzionale

PIANO DI TEST UTILIZZANDO IL METODO DEL *CATEGORY-PARTITION TESTING* PER LA FUNZIONALITÀ DI “*PrenotaVisita*”.

CPT: ‘PrenotaVisita’

L'applicazione permette al cittadino registrato di prenotare una visita inserendo i seguenti dati:

- Data e orario della visita rispettivamente nel formato gg/mm/aaaa e HH:MM definita sulle 24 ore.
Le visite sono svolte dal lunedì al venerdì dalle 07:30 alle 17:30, mentre il sabato dalle 07:30 alle 13:30. Esse hanno una durata di 30 minuti , pertanto il cittadino dovrà prenotarsi inserendo un orario di inizio visita nel seguente formato: "HH:30" o "HH:00"
- Descrizione della visita. Stringa di caratteri alfabetici presente all'interno del seguente elenco:
 - cardiologia;
 - dermatologia;
 - endocrinologia;
 - fisiatria;
 - gastroenterologia;
 - neurologia;
 - oculistica;
 - oncologia;
 - ortopedia;
 - ginecologia;
 - otorinolaringoiatria.

CATEGORIA	VALORI
Data della visita	<ul style="list-style-type: none"> • Data con formato valido, giorno: da lunedì a venerdì • Data con formato valido, giorno: sabato [SABATO] • Data con formato non valido [ERROR] • Data con formato valido, giorno: domenica [ERROR]
Orario della visita	<ul style="list-style-type: none"> • Fascia oraria 07:30-13:30 • Fascia oraria 13:30-17:30 [if not SABATO] • Formato valido e valori non validi [ERROR] • Formato invalido [ERROR] • Formato valido, valori validi, fuori fascia oraria [ERROR]
Descrizione della visita	<ul style="list-style-type: none"> • Stringa di caratteri compresa nell'elenco • Stringa di caratteri non compresa nell'elenco [ERROR]
Medico in servizio	<ul style="list-style-type: none"> • Vero • Falso [ERROR]
Visita già presente nel database	<ul style="list-style-type: none"> • Vero [ERROR] • Falso

Ci sono 3 categorie, di cui: 1 categorie con 4 classi di valori, 3 categorie con 2 classi di valori, 1 categorie con 5 classi di valori.

Numero di test senza vincoli: $4^*5^*(2^3) = 160$.

Ci son 8 vincoli di [ERROR]:

Numero di test con vincoli [ERROR]: $2^*1^*2^*1^*1 + 8 = 12$

Modalità di calcolo del numero di test con vincoli property

Escludiamo le classi di valori etichettate [ERROR] e ci riferiamo alle categorie che possiedono più di una classe di valori, perché le altre non alterano il numero dei test-case da generare comportando un fattore moltiplicativo unitario.

Restano i seguenti sottoinsiemi con cardinalità maggiore di uno:

Data della visita {Data con formato valido, giorno: da lunedì a venerdì, Data con formato valido, giorno: sabato}
 Orario della visita {Fascia oraria 07:30-13:30, Fascia oraria 13:30-17:30}

Procedimento Additivo

Calcoliamo tutte le combinazioni valide senza considerare il caso con l'etichetta if-property, cioè escludiamo le combinazioni che includono "Fascia oraria 13:30-17-30".

$$|\{\text{Data con formato valido, giorno: da lunedì a venerdì, Data con formato valido, giorno: sabato}\}| \times |\{\text{Fascia oraria 07:30-13-30}\}| = 2^*1 = 2$$

Aggiungiamo le combinazione escluse dal conteggio considerando i vincoli property:

$$|\{\text{Data con formato valido, giorno: sabato}\}| \times |\{\text{Fascia oraria 13:30-17-30}\}| = 1^*1 = 1$$

$$\text{Totale} = 2 + 1 = 3$$

Procedimento sottrattivo

Calcoliamo tutte le combinazioni valide ignorando i vincoli di property:

$$|\{\text{Data con formato valido, giorno: da lunedì a venerdì, Data con formato valido, giorno: sabato}\}| \times |\{\text{Fascia oraria 07:30-13-30, Fascia oraria 13:30-17-30}\}| = 2^*2 = 4$$

Sottraiamo dal risultato precedente le combinazioni non valide considerando i vincoli property:

$$|\{\text{Data con formato valido, giorno: sabato}\}| \times |\{\text{Fascia oraria 13:30-17-30}\}| = 1^*1 = 1$$

$$\text{Totale} = 4 - 1 = 3$$

Per entrambi i procedimenti al numero totale di test-case vanno aggiunti i 10 test-case ottenuti considerando i vincoli [ERROR].

In totale i casi di test sono $8 + 3 = 11$

TEST SUITE

PrenotaVisita

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese	Output Ottenuti	Post-condizioni Ottenute	Esito (FAIL, PASS)
1	Richiesta di prenotazione con esito positivo	Data con formato valido, giorno: da lunedì a venerdì Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data=16/03/2021 Orario = 15:00 Descrizione = cardiologia	"Prenotazione confermata"	Il sistema memorizza la visita e la prenotazione associata			
2	Richiesta di prenotazione con esito positivo	Data con formato valido, giorno: sabato Fascia oraria 07:30-13:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 17/04/2021 Orario = 07:30 Descrizione = neurologia	"Prenotazione confermata"	Il sistema memorizza la visita e la prenotazione associata			
3	Richiesta di una prenotazione con descrizione non compresa nell'elenco dell'ASL	Data con formato valido, giorno: da lunedì a venerdì Fascia oraria 13:30-17:30 Stringa di caratteri non compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 17/03/2021 Orario = 14:30 Descrizione = psichiatria	"Errore: hai inserito una descrizione di visita non prevista"	Nessuna			
4	Richiesta di una prenotazione con orario al di fuori delle 24 ore	Data con formato valido, giorno: sabato Formato valido e valori non validi Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 19/06/2021 Orario = 26:00 Descrizione = cardiologia	"Errore: l'orario inserito deve essere nel formato HH:00 o HH:30 e definito sulle 24 ore"	Nessuna			

5	Richiesta di una prenotazione con formato orario non valido	Data con formato valido, giorno: da lunedì a venerdì Formato invalido Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 19/06/2021 Orario = 9:30 Descrizione = cardiologia	"Errore: l'orario inserito deve essere nel formato HH:00 o HH:30 e definito sulle 24 ore"	Nessuna				
6	Richiesta di una prenotazione con orario al di fuori della fascia oraria consentita dall'ASL	Data con formato valido, giorno: sabato Formato valido, valori validi, fuori fascia oraria Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 17/04/2021 Orario = 19:00 Descrizione = neurologia	"Errore: orario non disponibile per il giorno inserito"	Nessuna				
7	Richiesta di una prenotazione con formato data non valido	Data con formato non valido Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 8/12/21 Orario = 16:30 Descrizione = cardiologia	"Errore: hai inserito un formato data non valido"	Nessuna				
8	Richiesta di una prenotazione per una domenica	Data con formato valido, giorno: domenica Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 18/04/2021 Orario = 16:00 Descrizione = cardiologia	"Errore: orario non disponibile per il giorno inserito"	Nessuna				
9	Richiesta di una prenotazione per un sabato in una fascia oraria non consentita per quel giorno	Data con formato valido, giorno: sabato Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 09/01/2021 Orario = 17:00 Descrizione = cardiologia	"Errore: orario non disponibile per il giorno inserito"	Nessuna	"			
10	Richiesta di una prenotazione per una visita già prenotata	Data con formato valido, giorno: domenica Fascia oraria 13:30-	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita già presente nel database	Data = 16/03/2021 Orario = 15:00	"Prenotazione rifiutata"	Nessuna				

		17:30 Stringa di caratteri compresa nell'elenco Vero Vero		Descrizione = cardiologia					
11	Richiesta di una prenotazione per un medico non in servizio	Data con formato valido, giorno: sabato Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Falso Falso	Il cittadino registrato ha effettuato con successo il login. Medico non in servizio e visita non presente database	Data = 16/12/2020 Orario = 15:00 Descrizione = cardiologia	"Medico indisponibile per la data e l'orario indicati"	Nessuna			

PIANO DI TEST UTILIZZANDO IL METODO DEL *CATEGORY-PARTITION TESTING* PER LA FUNZIONALITÀ DI “*Registrazione*”.

CPT: ‘*Registrazione*’

L’applicazione permette al cittadino di registrarsi inserendo i seguenti dati:

- Nome e cognome. Entrambi sono stringhe di soli caratteri alfabetici e non possono eccedere la lunghezza massima di 20 caratteri
- Luogo di nascita e residenza. Entrambi sono stringhe di soli caratteri alfabetici e non possono eccedere la lunghezza massima di 30 caratteri
- Codice Fiscale. Esso è una stringa alfanumerica di 16. I primi 6 sono caratteri alfabetici, seguiti da 3 caratteri alfanumerici, 2 cifre, 4 caratteri alfanumerici e terminati da una lettera.
- Data di nascita nel seguente formato gg/mm/aaaa.
- Numero di cellulare, composto da 10 cifre.
- Email. Essa è una stringa alfanumerica univoca di almeno 7 caratteri contenente una '@' seguita da un punto che non sia consecutivo ad essa. Per esempio “prova@gmail.com”.
- Username. Esso è una stringa alfanumerica univoca di lunghezza compresa tra i 6 e i 20 caratteri.
- Password. Essa è una stringa alfanumerica di lunghezza compresa tra gli 8 e i 30 caratteri, che comprendono almeno una lettera maiuscola e una cifra.

CATEGORIA	VALORI
N: Nome	<ol style="list-style-type: none"> Stringa di caratteri, lunghezza ≤ 20 Stringa di caratteri, lunghezza > 20 [ERROR] Stringa di caratteri con presenza di altri simboli [ERROR]
C: Cognome	<ol style="list-style-type: none"> Stringa di caratteri, lunghezza ≤ 20 Stringa di caratteri, lunghezza > 20 [ERROR] Stringa di caratteri con presenza di altri simboli [ERROR]
CF: Codice Fiscale	<ol style="list-style-type: none"> Stringa alfanumerica, lunghezza > 16 [ERROR] Stringa alfanumerica, lunghezza < 16 [ERROR] Stringa alfanumerica, lunghezza = 16, formato valido Stringa alfanumerica, lunghezza = 16, formato non valido [ERROR]
DN: Data di nascita	<ol style="list-style-type: none"> Data con formato valido Data con formato non valido [ERROR]
LN: Luogo di nascita	<ol style="list-style-type: none"> Stringa di caratteri, lunghezza ≤ 30 Stringa di caratteri, lunghezza > 30 [ERROR] Stringa di caratteri con presenza di altri simboli [ERROR]
LR: Luogo di residenza	<ol style="list-style-type: none"> Stringa di caratteri, lunghezza ≤ 30 Stringa di caratteri, lunghezza > 30 [ERROR] Stringa di caratteri con presenza di altri simboli [ERROR]
NC: Numero di cellulare	<ol style="list-style-type: none"> Stringa di numeri, lunghezza = 10 Stringa di numeri, lunghezza > 10 [ERROR] Stringa di numeri, lunghezza < 10 [ERROR] Stringa di numeri con presenza di altri simboli [ERROR]
E: Email	<ol style="list-style-type: none"> Stringa alfanumerica, lunghezza ≤ 7 [ERROR] Stringa alfanumerica, lunghezza > 7, formato valido Stringa alfanumerica, lunghezza > 7, formato non valido [ERROR]
U: Username	<ol style="list-style-type: none"> Stringa alfanumerica, $6 \leq$ lunghezza ≤ 20 Stringa alfanumerica, lunghezza > 20 [ERROR] Stringa alfanumerica, lunghezza < 6 [ERROR]
P: Password	<ol style="list-style-type: none"> Stringa alfanumerica, lunghezza < 8 [ERROR] Stringa alfanumerica, lunghezza > 30 [ERROR] Stringa alfanumerica, $8 \leq$ lunghezza ≤ 30, formato valido Stringa alfanumerica, $8 \leq$ lunghezza ≤ 30, formato non valido [ERROR]
EP: Email già presente nel database	<ol style="list-style-type: none"> Falso Vero [ERROR]
UP: Username già presente nel database	<ol style="list-style-type: none"> Falso Vero [ERROR]

Ci sono 12 categorie, di cui: 6 categorie con 3 classi di valori, 3 categorie con 2 classi di valori, 3 categorie con 4 classi di valori.

Numero di test senza vincoli: $(3^6) * (2^3) * (4^3) = 373248$.

Ci son 24 vincoli di [ERROR]:

Numero di test con vincoli [ERROR]: $1 * 1 \dots * 1 + 24 = 25$

TEST SUITE

Registrazione

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese	Output Ottenuati	Post-condizioni Ottenuate	Esito (FAIL, PASS)
1	Registrazione corretta di un cittadino	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Registrazione avvenuta correttamente"	I dati del cittadino registrato sono memorizzati nel database			

2	Registrazione di un cittadino con Username già utilizzato	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.2	Email non presente e Username presente nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocast@gmail.com U = checcocast P = Checco1978	"Username già utilizzato"	Nessuna			
3	Registrazione di un cittadino con Email già utilizzata	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.2 UP.1	Username non presente e Email presente nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcoNapoli P = Checco1978	"Email già utilizzata"	Nessuna			
4	Registrazione di un cittadino con formato Password non valido	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = checcocast	"Errore: la Password deve contenere almeno una cifra e un carattere maiuscolo"	Nessuna			
5	Registrazione di un cittadino con Password di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.2 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Francesco78castaldoForzaNapoli!	"Errore: la Password deve avere una lunghezza compresa tra gli 8 e i 30 caratteri"	Nessuna			

6	Registrazione di un cittadino con Password di lunghezza inferiore al limite minimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.1 EP.1 UP.1	E-mail e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1	"Errore: la Password deve avere una lunghezza compresa tra gli 8 e i 30 caratteri"	Nessuna		
7	Registrazione di un cittadino con Username di lunghezza inferiore al limite minimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.3 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = kekko P = Checco1978	"Errore: l'Username deve avere una lunghezza compresa tra i 6 e i 20 caratteri"	Nessuna		
8	Registrazione di un cittadino con Username di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.2 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = francescocastaldo1978 P = Checco1978	"Errore: l'Username deve avere una lunghezza compresa tra i 6 e i 20 caratteri"	Nessuna		
9	Registrazione di un cittadino con formato Email non valido	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.3 U.1 P.3 EP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldogmail.com U = checcocast P = Checco1978	"Errore: l'Email deve essere nel seguente formato prova@gmail.com "	Nessuna		

		UP.1							
10	Registrazione di un cittadino con Email di lunghezza inferiore al limite minimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.1 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = fr@a.it U = checcocast P = Checco1978	"Errore: l'Email deve essere lunga almeno 8 caratteri"	Nessuna			
11	Registrazione di un cittadino con numero di cellulare che include simboli non permessi	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.4 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 33356789*? E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il numero di cellulare inserito contiene simboli non permessi"	Nessuna			
12	Registrazione di un cittadino con numero di cellulare con più di 10 cifre	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.2 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 333567890145 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il numero di cellulare inserito deve essere di 10 cifre"	Nessuna			
13	Registrazione di un cittadino con numero di cellulare con meno di 10 cifre	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.3 E.2 U.1 P.3	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 33356789 E = francescocastaldo@gmail.com	"Errore: il numero di cellulare inserito deve essere di 10 cifre"	Nessuna			

		EP.1 UP.1		U = checcocast P = Checco1978				
14	Registrazione di un cittadino con luogo di residenza di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.1 LR.2 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Primiero San Martino di Castrozza NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il luogo di residenza inserito deve avere meno di 30 caratteri"	Nessuna		
15	Registrazione di un cittadino con luogo di residenza che include simboli non permessi	N.1 C.1 CF.3 DN.1 LN.1 LR.3 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli! NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il luogo di residenza inserito contiene simboli non permessi"	Nessuna		
16	Registrazione di un cittadino con luogo di nascita di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.2 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Primiero San Martino di Castrozza LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il luogo di nascita inserito deve avere meno di 30 caratteri"	Nessuna		
17	Registrazione di un cittadino con luogo di nascita che include simboli non permessi	N.1 C.1 CF.3 DN.1 LN.3	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = NapOli** LR = Napoli NC = 3335678901	"Errore: il luogo di nascita inserito contiene simboli non permessi"	Nessuna		

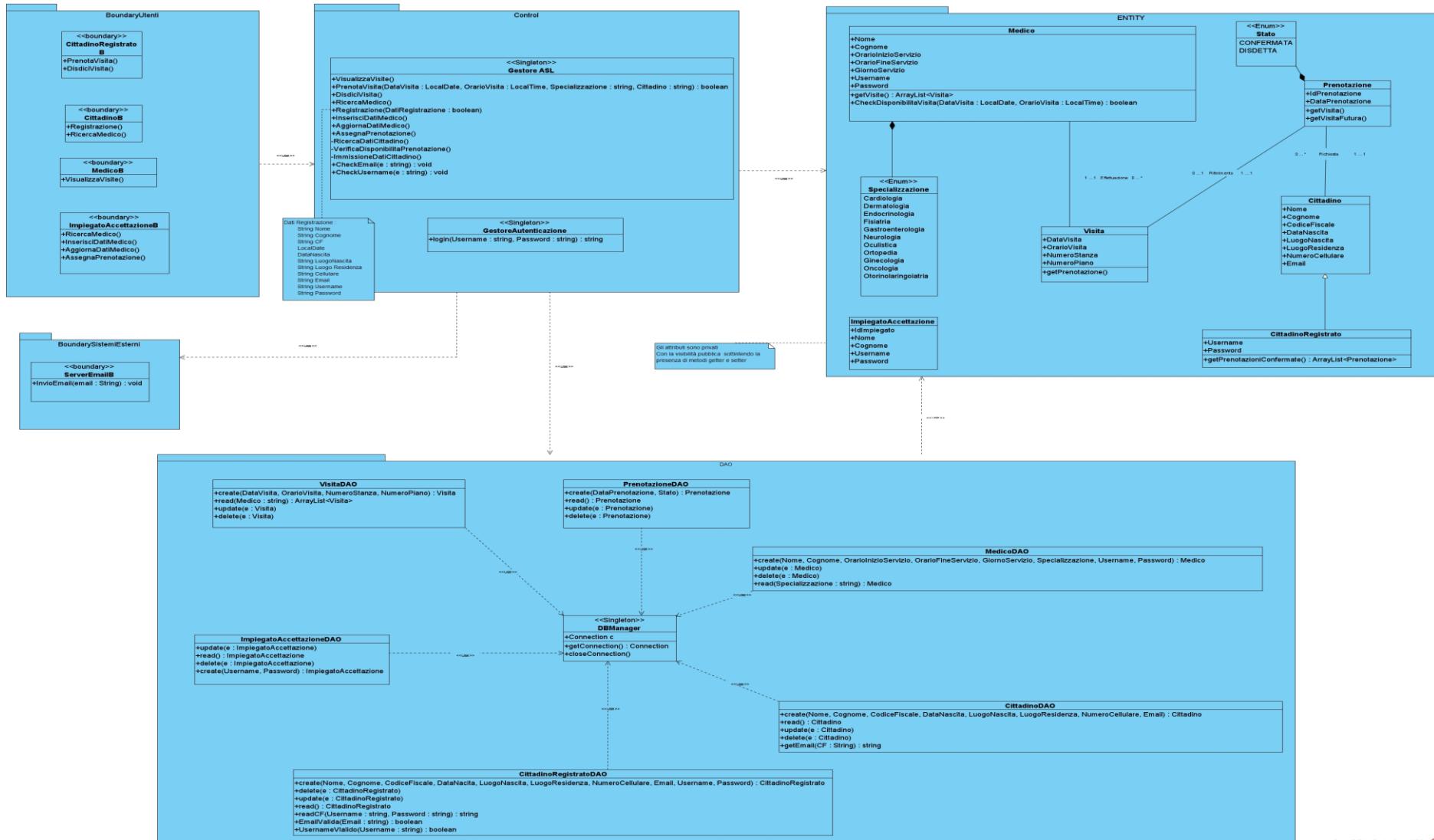
		LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1		E = francescocastaldo@gmail.com U = checcocast P = Checco1978					
18	Registrazione di un cittadino con formato Codice fiscale non valido	N.1 C.1 CF.4 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = C6TFN978B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il Codice Fiscale inserito non è nel formato standard"	Nessuna			
19	Registrazione di un cittadino con Codice fiscale di lunghezza minore di 16	N.1 C.1 CF.2 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839 DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il Codice Fiscale inserito deve avere 16 caratteri "	Nessuna			
20	Registrazione di un cittadino con Codice fiscale di lunghezza maggiore di 16	N.1 C.1 CF.1 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839CXX DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il Codice Fiscale inserito deve avere 16 caratteri"	Nessuna			

21	Registrazione di un cittadino con cognome di lunghezza superiore al massimo	N.1 C.2 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo del Conte di Savoia CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il cognome inserito deve avere meno di 20 caratteri"	Nessuna			
22	Registrazione di un cittadino con cognome che include simboli non permessi	N.1 C.3 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo0 CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il cognome inserito contiene simboli non permessi"	Nessuna			
23	Registrazione di un cittadino con nome di lunghezza superiore al massimo	N.2 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco Antonio Maria C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il nome inserito deve avere meno di 20 caratteri"	Nessuna			
24	Registrazione di un cittadino con nome che include simboli non permessi	N.3 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1	Email e Username non presenti nel database	N = Francesc0 C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il nome inserito contiene simboli non permessi"	Nessuna			

		UP.1						
25	Registrazione di un cittadino con formato data non valido	N.1 C.1 CF.3 DN.2 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/78 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: la data inserita deve essere nel seguente formato: gg/mm/aaaa "	Nessuna		

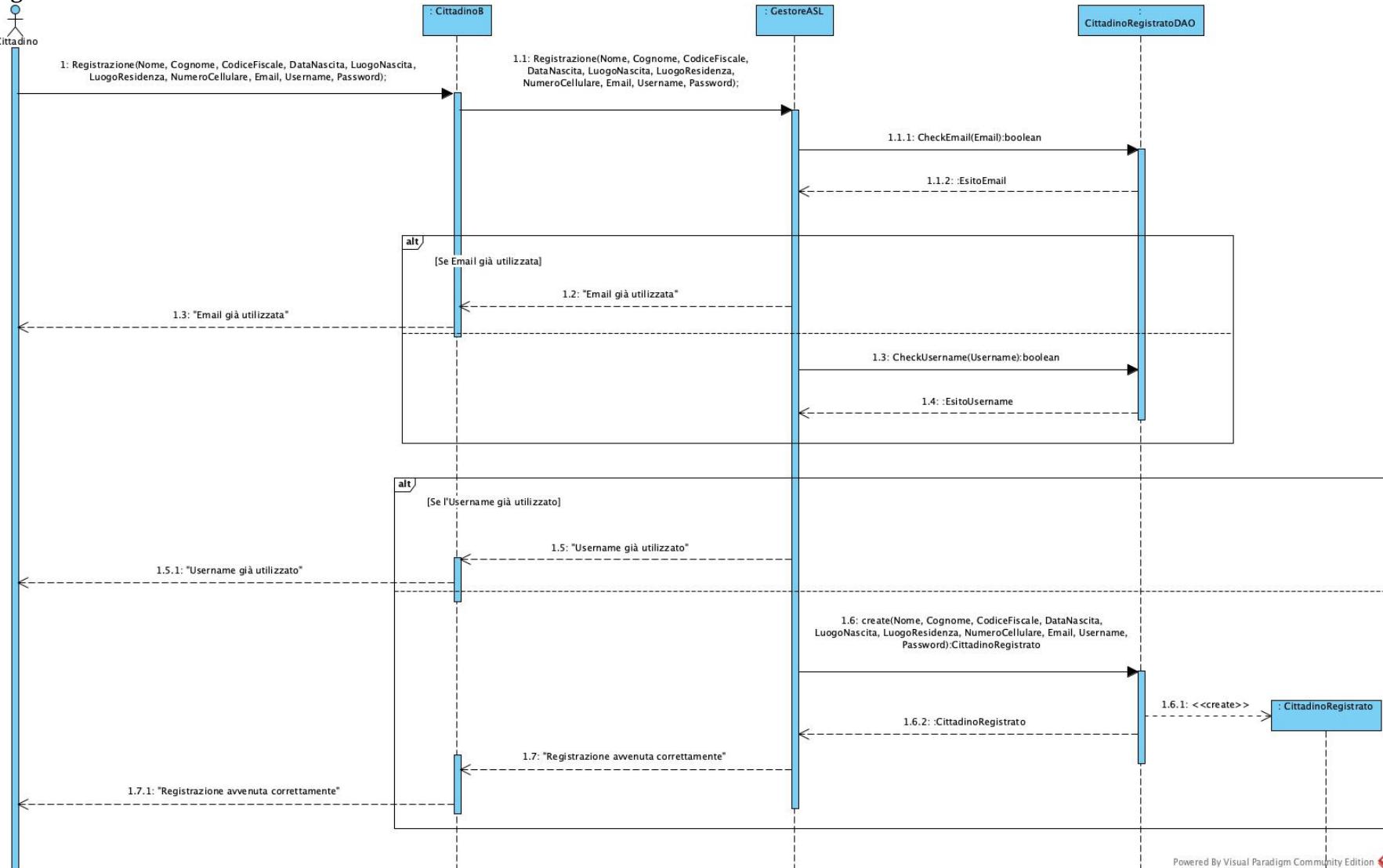
5. Progettazione

5.1 Diagramma delle classi

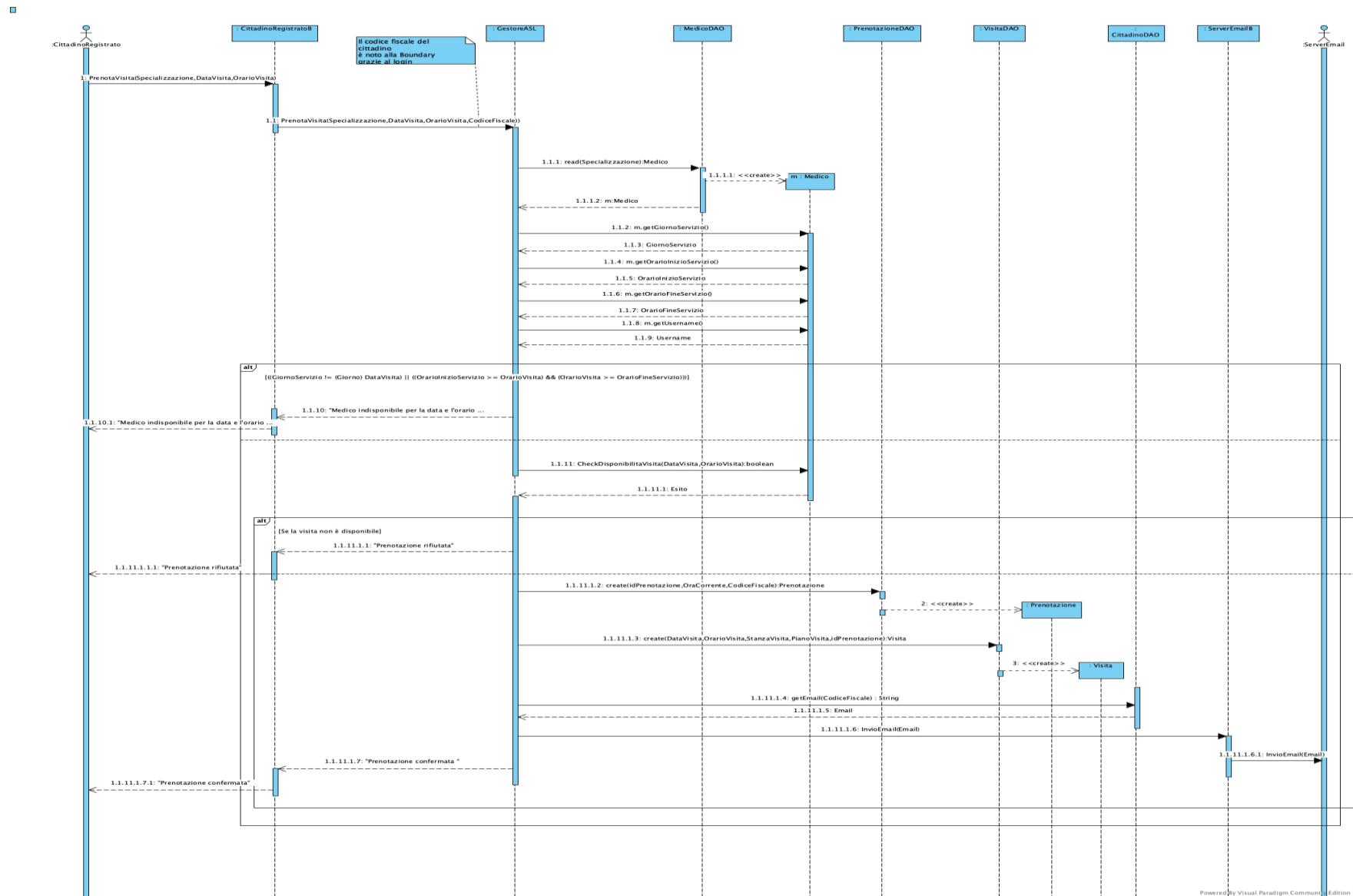


5.2 Diagrammi di sequenza

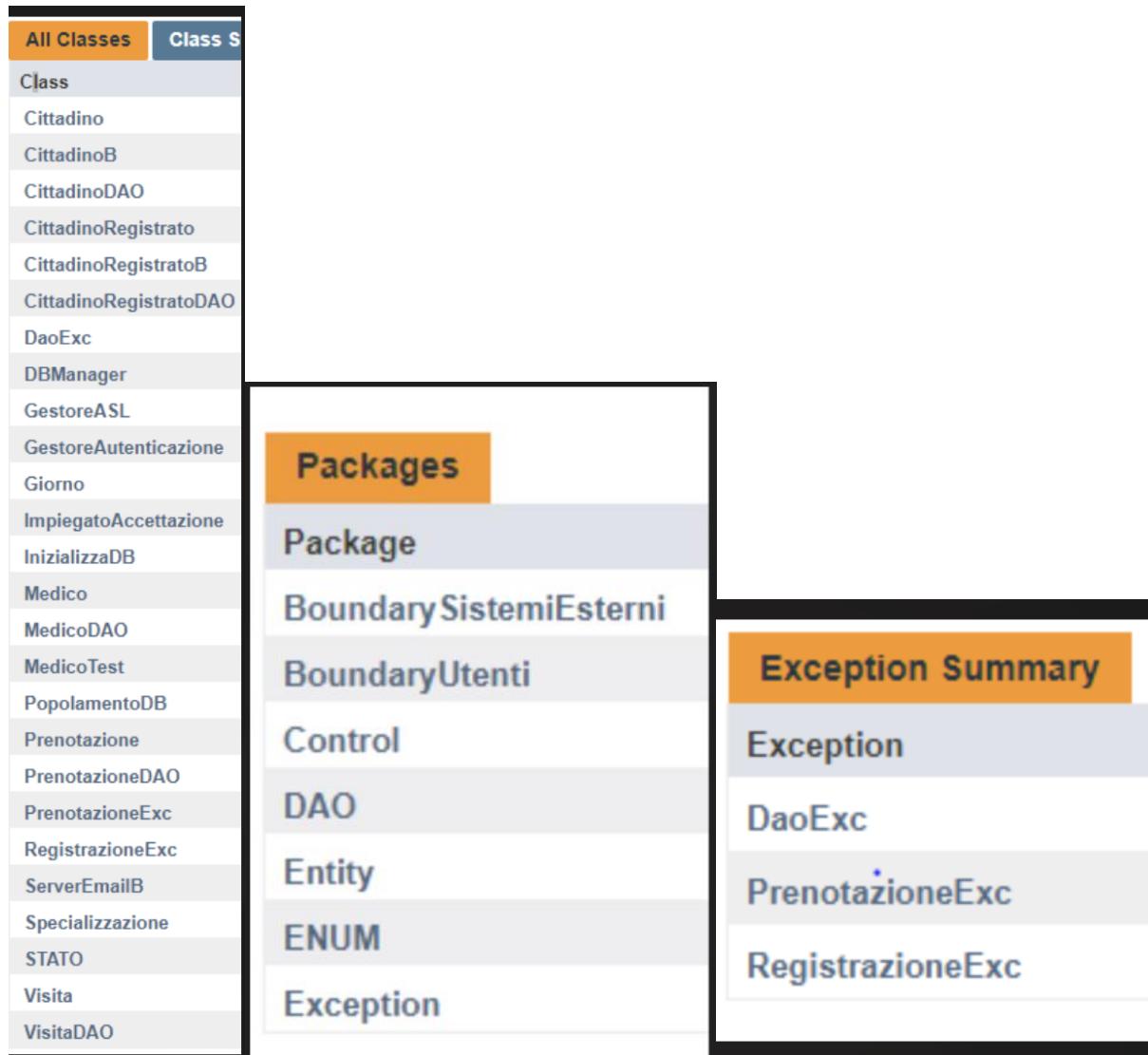
Registrazione



PrenotaVisita



6. Implementazione



Class GestoreASL

java.lang.Object
Control.GestoreASL

```
public class GestoreASL  
extends java.lang.Object
```

Field Summary

Fields

Modifier and Type	Field	Description
static GestoreASL	gA	

Constructor Summary

Constructors

Modifier	Constructor	Description
protected	GestoreASL()	

Method Summary

All Methods Static Methods Instance Methods Concrete Methods

Modifier and Type	Method	Description
private static void	CheckEmail(java.lang.String Email)	
private static void	CheckUsername(java.lang.String Username)	
private static java.lang.String	getGiorno(java.time.LocalDate Date)	
static GestoreASL	getInstance()	
boolean	PrenotaVisita(java.time.LocalDate DataVisita, java.time.Localtime OrarioVisita, java.lang.String Spec, java.lang.String Cittadino)	
boolean	Registrazione(java.lang.String Nome, java.lang.String Cognome, java.lang.String CodiceFiscale, java.time.LocalDate DataNascita, java.lang.String LuogoNascita, java.lang.String LuogoResidenza, java.lang.String NumeroCellulare, java.lang.String Email, java.lang.String Username, java.lang.String Password)	

Method Details

getInstance

```
public static GestoreASL getInstance()
```

getGiorno

```
private static java.lang.String getGiorno(java.time.LocalDate Date)
```

PrenotaVisita

```
public boolean PrenotaVisita(java.time.LocalDate DataVisita,
                           java.time.LocalTime OrarioVisita,
                           java.lang.String Spec,
                           java.lang.String Cittadino)
                           throws PrenotazioneExc
```

Throws:
PrenotazioneExc

Registrazione

```
public boolean Registrazione(java.lang.String Nome,
                           java.lang.String Cognome,
                           java.lang.String CodiceFiscale,
                           java.time.LocalDate DataNascita,
                           java.lang.String LuogoNascita,
                           java.lang.String LuogoResidenza,
                           java.lang.String NumeroCellulare,
                           java.lang.String Email,
                           java.lang.String Username,
                           java.lang.String Password)
                           throws RegistrazioneExc
```

Throws:
RegistrazioneExc

CheckUsername

```
private static void CheckUsername(java.lang.String Username)
                           throws RegistrazioneExc,
                           DaoExc
```

Throws:
RegistrazioneExc
DaoExc

CheckEmail

```
private static void CheckEmail(java.lang.String Email)
                           throws RegistrazioneExc,
                           DaoExc
```

Throws:

RegistrazioneExc

DaoExc

NOTA:

Prima di eseguire i casi di test è necessario inizializzare il DB e popolarlo con le apposite classi inserite nel default package. Inoltre, aggiornare il path del file h2-1.4.196.jar presente nella cartella JAVA.

7. Testing

7.1 Test funzionale

PrenotaVisita

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese	Output Ottenuti	Post-condizioni Ottenute	Esito (FAIL, PASS)
1	Richiesta di prenotazione con esito positivo	Data con formato valido, giorno: da lunedì a venerdì Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data=16/03/2021 Orario = 15:00 Descrizione = cardiologia	"Prenotazione confermata"	Il sistema memorizza la visita e la prenotazione associata	"Prenotazione confermata "	Il sistema memorizza la visita e la prenotazione associata	PASS
2	Richiesta di prenotazione con esito positivo	Data con formato valido, giorno: sabato Fascia oraria 07:30-13:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 17/04/2021 Orario = 07:30 Descrizione = neurologia	"Prenotazione confermata"	Il sistema memorizza la visita e la prenotazione associata	"Prenotazione confermata"	Il sistema memorizza la visita e la prenotazione associata	PASS
3	Richiesta di una prenotazione con descrizione non compresa nell'elenco dell'ASL	Data con formato valido, giorno: da lunedì a venerdì Fascia oraria 13:30-17:30 Stringa di caratteri non compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 17/03/2021 Orario = 14:30 Descrizione = psichiatria	"Errore: hai inserito una descrizione di visita non prevista"	Nessuna	"Errore: hai inserito una descrizione di visita non prevista"	Nessuna	PASS
4	Richiesta di una prenotazione con orario al di fuori delle 24 ore	Data con formato valido, giorno: sabato Formato valido e valori non validi Stringa di caratteri compresa nell'elenco	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 19/06/2021 Orario = 26:00 Descrizione = cardiologia	"Errore: l'orario inserito deve essere nel formato HH:00 o HH:30 e definito sulle 24 ore"	Nessuna	"Errore: l'orario inserito deve essere nel formato HH:00 o HH:30 e definito sulle 24 ore"	Nessuna	PASS

		Vero Falso							
5	Richiesta di una prenotazione con formato orario non valido	Data con formato valido, giorno: da lunedì e venerdì Formato invalido Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 13/01/2021 Orario = 9:30 Descrizione = cardiologia	"Errore: l'orario inserito deve essere nel formato HH:00 o HH:30 e definito sulle 24 ore"	Nessuna	"Errore: l'orario inserito deve essere nel formato HH:00 o HH:30 e definito sulle 24 ore"	Nessuna	PASS
6	Richiesta di una prenotazione con orario al di fuori della fascia oraria consentita dall'ASL	Data con formato valido, giorno: sabato Formato valido, valori validi, fuori fascia oraria Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 17/04/2021 Orario = 19:00 Descrizione = neurologia	"Errore: orario non disponibile per il giorno inserito"	Nessuna	"Errore: orario non disponibile per il giorno inserito"	Nessuna	PASS
7	Richiesta di una prenotazione con formato data non valido	Data con formato non valido Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 8/12/21 Orario = 16:30 Descrizione = cardiologia	"Errore: hai inserito un formato data non valido"	Nessuna	"Errore: hai inserito un formato data non valido"	Nessuna	PASS
8	Richiesta di una prenotazione per una domenica	Data con formato valido, giorno: domenica Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 18/04/2021 Orario = 16:00 Descrizione = cardiologia	"Errore: orario non disponibile per il giorno inserito"	Nessuna	"Errore: orario non disponibile per il giorno inserito"	Nessuna	PASS
9	Richiesta di una prenotazione per un sabato in una fascia oraria non consentita per quel giorno	Data con formato valido, giorno: sabato Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Falso	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita non presente nel database	Data = 09/01/2021 Orario = 17:00 Descrizione = cardiologia	"Errore: orario non disponibile per il giorno inserito"	Nessuna	"Errore: orario non disponibile per il giorno inserito"	Nessuna	PASS
10	Richiesta di una prenotazione per una visita già prenotata	Data con formato valido, giorno: domenica Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Vero Vero	Il cittadino registrato ha effettuato con successo il login. Medico in servizio e visita già presente nel database	Data = 16/03/2021 Orario = 15:00 Descrizione = cardiologia	"Prenotazione rifiutata"	Nessuna	"Prenotazione rifiutata"	Nessuna	PASS

11	Richiesta di una prenotazione per un medico non in servizio	Data con formato valido, giorno: sabato Fascia oraria 13:30-17:30 Stringa di caratteri compresa nell'elenco Falso Falso	Il cittadino registrato ha effettuato con successo il login. Medico non in servizio e visita non presente database	Data = 16/12/2020 Orario = 15:00 Descrizione = cardiologia	"Medico indisponibile per la data e l'orario indicati"	Nessuna	"Medico indisponibile per la data e l'orario indicati"	Nessuna PASS

Registrazione

Test Case ID	Descrizione	Classi di equivalenza coperte	Pre-condizioni	Input	Output Attesi	Post-condizioni Attese	Output Otttenuti	Post-condizioni Otttenute	Esito (FAIL, PASS)
1	Registrazione corretta di un cittadino	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Registrazione avvenuta correttamente"	I dati del cittadino registrato sono memorizzati nel database	"Registrazione avvenuta correttamente"	I dati del cittadino registrato sono memorizzati nel database	PASS
2	Registrazione di un cittadino con Username già utilizzato	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.1 UP.2	Email non presente e Username presente nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocast@gmail.com U = checcocast P = Checco1978	"Username già utilizzato"	Nessuna	"Username già utilizzato"	Nessuna	PASS
3	Registrazione di un cittadino con Email già utilizzata	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.3 EP.2	Username non presente e Email presente nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com	"Email già utilizzata"	Nessuna	"Email già utilizzata"	Nessuna	PASS

		UP.1		U = checcoNapoli P = Checco1978					
4	Registrazione di un cittadino con formato Password non valido	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = checcocast	"Errore: la Password deve contenere almeno una cifra e un carattere maiuscolo"	Nessuna	"Errore: la Password deve contenere almeno una cifra e un carattere maiuscolo"	Nessuna	PASS
5	Registrazione di un cittadino con Password di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.2 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Francesco78castaldoForzaNapoli!	"Errore: la Password deve avere una lunghezza compresa tra gli 8 e i 30 caratteri"	Nessuna	"Errore: la Password deve avere una lunghezza compresa tra gli 8 e i 30 caratteri"	Nessuna	PASS
6	Registrazione di un cittadino con Password di lunghezza inferiore al limite minimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.1 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1	"Errore: la Password deve avere una lunghezza compresa tra gli 8 e i 30 caratteri"	Nessuna	"Errore: la Password deve avere una lunghezza compresa tra gli 8 e i 30 caratteri"	Nessuna	PASS

7	Registrazione di un cittadino con Username di lunghezza inferiore al limite minimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.3 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = kekko P = Checco1978	"Errore: l'Username deve avere una lunghezza compresa tra i 6 e i 20 caratteri"	Nessuna	"Errore: l'Username deve avere una lunghezza compresa tra i 6 e i 20 caratteri"	Nessuna	PASS
8	Registrazione di un cittadino con Username di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.2 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = francescocastaldo1978 P = Checco1978	"Errore: l'Username deve avere una lunghezza compresa tra i 6 e i 20 caratteri"	Nessuna	"Errore: l'Username deve avere una lunghezza compresa tra i 6 e i 20 caratteri"	Nessuna	PASS
9	Registrazione di un cittadino con formato Email non valido	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.3 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldogmail.com U = checcocast P = Checco1978	"Errore: l'Email deve essere nel seguente formato prova@gmail.com "	Nessuna	"Errore: l'Email deve essere nel seguente formato prova@gmail.com "	Nessuna	PASS
10	Registrazione di un cittadino con Email di lunghezza inferiore al limite minimo	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.1 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = fr@a.it U = checcocast P = Checco1978	"Errore: l'Email deve essere lunga almeno 8 caratteri"	Nessuna	"Errore: l'Email deve essere lunga almeno 8 caratteri"	Nessuna	PASS

11	Registrazione di un cittadino con numero di cellulare che include simboli non permessi	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.4 E.2 U.1 P.3 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 33356789*? E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il numero di cellulare inserito contiene simboli non permessi"	Nessuna	"Errore: il numero di cellulare inserito contiene simboli non permessi"	Nessuna	PASS
12	Registrazione di un cittadino con numero di cellulare con più di 10 cifre	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.2 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 333567890145 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il numero di cellulare inserito deve essere di 10 cifre"	Nessuna	"Errore: il numero di cellulare inserito deve essere di 10 cifre"	Nessuna	PASS
13	Registrazione di un cittadino con numero di cellulare con meno di 10 cifre	N.1 C.1 CF.3 DN.1 LN.1 LR.1 NC.3 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 33356789 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il numero di cellulare inserito deve essere di 10 cifre"	Nessuna	"Errore: il numero di cellulare inserito deve essere di 10 cifre"	Nessuna	PASS
14	Registrazione di un cittadino con luogo di residenza di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.1 LR.2 NC.1 E.2	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Primiero San Martino di Castrozza NC = 3335678901 E = francescocastaldo@gmail.com	"Errore: il luogo di residenza inserito deve avere meno di 30 caratteri"	Nessuna	"Errore: il luogo di residenza inserito deve avere meno di 30 caratteri"	Nessuna	PASS

		U.1 P.4 EP.1 UP.1		U = checcocast P = Checco1978					
15	Registrazione di un cittadino con luogo di residenza che include simboli non permessi	N.1 C.1 CF.3 DN.1 LN.1 LR.3 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli! NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il luogo di residenza inserito contiene simboli non permessi"	Nessuna	"Errore: il luogo di residenza inserito contiene simboli non permessi"	Nessuna	PASS
16	Registrazione di un cittadino con luogo di nascita di lunghezza superiore al limite massimo	N.1 C.1 CF.3 DN.1 LN.2 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Primiero San Martino di Castrozza LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il luogo di nascita inserito deve avere meno di 30 caratteri"	Nessuna	"Errore: il luogo di nascita inserito deve avere meno di 30 caratteri"	Nessuna	PASS
17	Registrazione di un cittadino con luogo di nascita che include simboli non permessi	N.1 C.1 CF.3 DN.1 LN.3 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = NapOli** LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il luogo di nascita inserito contiene simboli non permessi"	Nessuna	"Errore: il luogo di nascita inserito contiene simboli non permessi"	Nessuna	PASS
18	Registrazione di un cittadino con formato Codice fiscale non valido	N.1 C.1 CF.4 DN.1 LN.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = C6TFN978B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901	"Errore: il Codice Fiscale inserito non è nel formato standard"	Nessuna	"Errore: il Codice Fiscale inserito non è nel formato standard"	Nessuna	PASS

		LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1		E = francescocastaldo@gmail.com U = checcocast P = Checco1978					
19	Registrazione di un cittadino con Codice fiscale di lunghezza minore di 16	N.1 C.1 CF.2 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839 DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il Codice Fiscale inserito deve avere 16 caratteri"	Nessuna	"Errore: il Codice Fiscale inserito deve avere 16 caratteri"	Nessuna	PASS
20	Registrazione di un cittadino con Codice fiscale di lunghezza maggiore di 16	N.1 C.1 CF.1 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839CXX DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il Codice Fiscale inserito deve avere 16 caratteri"	Nessuna	"Errore: il Codice Fiscale inserito deve avere 16 caratteri"	Nessuna	PASS
21	Registrazione di un cittadino con cognome di lunghezza superiore al massimo	N.1 C.2 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo del Conte di Savoia CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il cognome inserito deve avere meno di 20 caratteri"	Nessuna	"Errore: il cognome inserito deve avere meno di 20 caratteri"	Nessuna	PASS
22	Registrazione di un cittadino con cognome che include simboli non permessi	N.1 C.3 CF.3 DN.1 LN.1	Email e Username non presenti nel database	N = Francesco C = Castaldo0 CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli	"Errore: il cognome inserito contiene simboli non permessi"	Nessuna	"Errore: il cognome inserito contiene simboli non permessi"	Nessuna	PASS

		LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1		NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978					
23	Registrazione di un cittadino con nome di lunghezza superiore al massimo	N.2 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco Antonio Maria C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il nome inserito deve avere meno di 20 caratteri"	Nessuna	"Errore: il nome inserito deve avere meno di 20 caratteri"	Nessuna	PASS
24	Registrazione di un cittadino con nome che include simboli non permessi	N.3 C.1 CF.3 DN.1 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesc0 C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/1978 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: il nome inserito contiene simboli non permessi"	Nessuna	"Errore: il nome inserito contiene simboli non permessi"	Nessuna	PASS
25	Registrazione di un cittadino con formato data non valido	N.1 C.1 CF.3 DN.2 LN.1 LR.1 NC.1 E.2 U.1 P.4 EP.1 UP.1	Email e Username non presenti nel database	N = Francesco C = Castaldo CF = CSTFNC78B09F839C DN = 18/05/78 LN = Napoli LR = Napoli NC = 3335678901 E = francescocastaldo@gmail.com U = checcocast P = Checco1978	"Errore: la data inserita deve essere nel seguente formato: gg/mm/aaaa "	Nessuna	"Errore: la data inserita deve essere nel seguente formato: gg/mm/aaaa "	Nessuna	PASS

7.2 Test strutturale

7.2.1 Complessità ciclomatica

CFG CheckOrario_Data

Numero Ciclomatico = 5

Percorsi indipendenti:

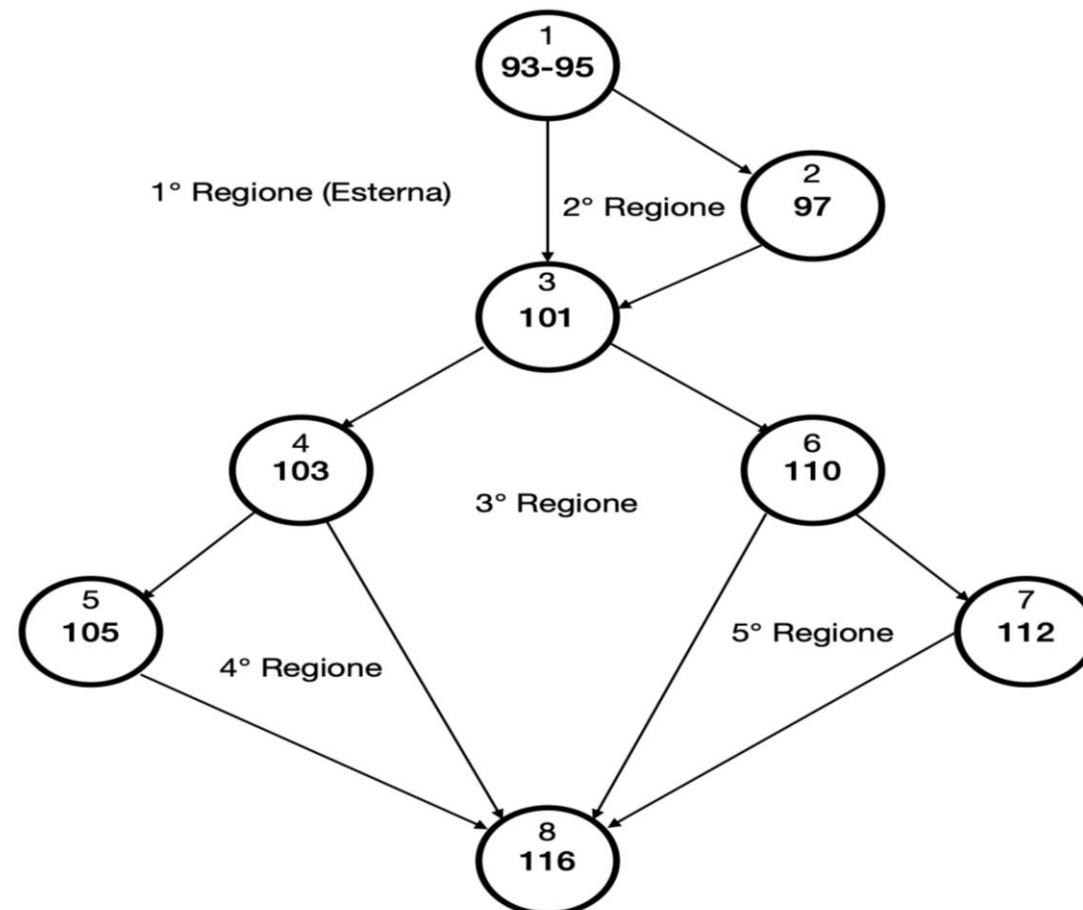
C1: 1-3-4-5-8

C2: 1-3-4-8

C3: 1-3-6-8

C4: 1-3-6-7-8

C5: 1-2-3-6-8

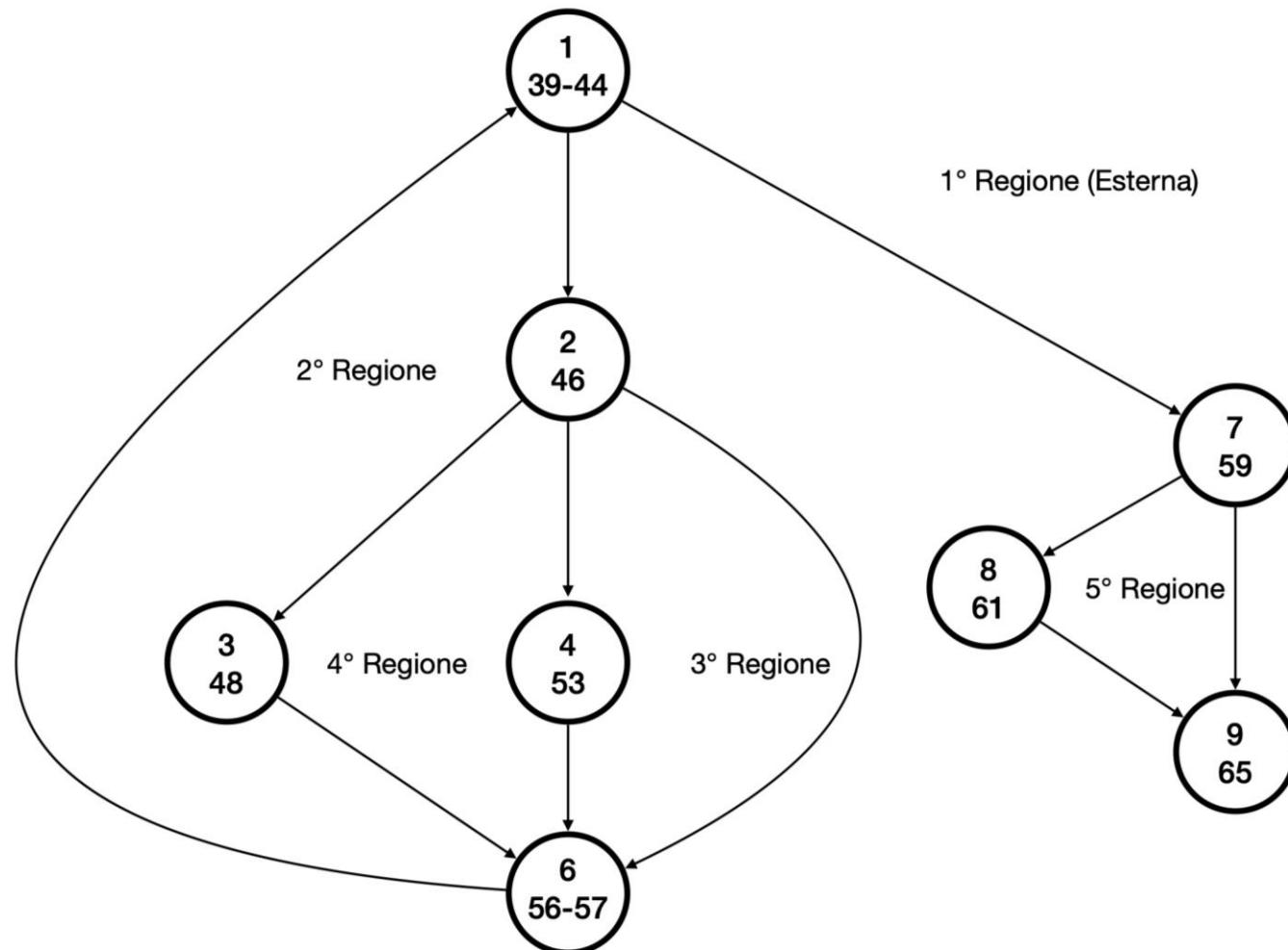


CFG CheckFormatoPassword

Numero Ciclomatico = 5

Percorsi indipendenti:

- C1: 1-2-4-6-1-7-8-9
- C2: 1-2-3-6-1-7-8-9
- C3: 1-2-6-1-7-8-9
- C4: 1-7-8-9
- C5: 1-7-9



7.2.2 Test di unità

Test-Suite CFG CheckFormatoPassword

Test-Case per C1:

 Password = “A”

 Risultato atteso: Trovato = false.

Test-Case per C2:

 Password = “5”

 Risultato atteso: Trovato = false.

Test-Case per C3:

 Password = “a”

 Risultato atteso: Trovato = false.

Test-Case per C4:

 Password = “”.

 Risultato atteso: Trovato = false.

Test-Case per C5:

 Non realizzabile per nessun valore di input.

Test-Suite CFG CheckOrario_Data

Test-Case per C1:

OrarioVisita = 18:30,
DataVisita = 19/12/2020(SABATO).
Risultato atteso: Trovato = false.

Test-Case per C2:

OrarioVisita = 12:30,
DataVisita = 19/12/2020(SABATO).
Risultato atteso: Trovato = true.

Test-Case per C3:

OrarioVisita = 12:30,
DataVisita = 18/12/2020(!SABATO, !DOMENICA).
Risultato atteso: Trovato = true.

Test-Case per C4:

OrarioVisita = 18:30,
DataVisita = 18/12/2020(!SABATO, !DOMENICA).
Risultato atteso: Trovato = false.

Test-Case per C5:

OrarioVisita = 12:30,
DataVisita = 20/12/2020(DOMENICA).
Risultato atteso: Trovato = false.

7.3 Test di unità con JUnit

```
java.lang.Object
Entity.MedicoTest

public class MedicoTest
extends java.lang.Object
```

Constructor Summary

Constructors	Description
Constructor MedicoTest()	

Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	
void	testAssociaVisita()	
void	testCheckDisponibilitaVisita()	
void	testGetCognome()	
void	testGetGiornoServizio()	
void	testGetNome()	
void	testGetOrarioFineServizio()	
void	testGetOrarioInizioServizio()	
void	testGetPassword()	
void	testGetSpecializzazione()	
void	testGetUsername()	
void	testGetVisite()	
void	testRimuoviVisita()	
void	testSetCognome()	
void	testSetGiornoServizio()	
void	testSetName()	
void	testSetOrarioFineServizio()	
void	testSetOrarioInizioServizio()	
void	testSetPassword()	
void	testSetSpecializzazione()	
void	testSetUsername()	
void	testSetVisitaRef()	

Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
↳ GestioneASL	28,5 %	980	2.460	3.440
↳ src	28,5 %	980	2.460	3.440
> ENUM	88,7 %	188	24	212
↳ Entity	71,9 %	792	309	1.101
> Medico.java	100,0 %	138	0	138
> MedicoTest.java	100,0 %	633	0	633
> Visita.java	33,9 %	21	41	62
> Cittadino.java	0,0 %	0	93	93
> CittadinoRegistrato.java	0,0 %	0	62	62
> ImpiegatoAccettazione.java	0,0 %	0	50	50
> Prenotazione.java	0,0 %	0	63	63
> (default package)	0,0 %	0	166	166
> BoundarySistemiEsterni	0,0 %	0	13	13
> BoundaryUtenti	0,0 %	0	966	966
> Control	0,0 %	0	232	232
> DAO	0,0 %	0	739	739
> Exception	0,0 %	0	11	11