



UNIVERSITÀ DEGLI STUDI DI MILANO - BICOCCA

Scuola di Scienze

Dipartimento di Informatica, Sistemistica e Comunicazione

Corso di laurea in Informatica

Cartella Cardiologica Virtuale

*Progettazione e Sviluppo di un'applicazione web
mediante i framework Spring e Hibernate*

Relatore: Prof.ssa Elisabetta Fersini

Correlatore: Dott.ssa Annalisa Marra (Sync Lab Srl)

Relazione della prova finale di:

Manuel Nicoletta

Matricola 806237

Anno Accademico 2019-2020

Sommario

Il progetto "Cartella Cardiologica Virtuale" è stato realizzato durante uno stage curricolare presso l'azienda Sync Lab Srl e ha lo scopo di virtualizzare la cartella cardiologica di un paziente.

Questo elaborato introduce i concetti teorici necessari per l'implementazione del software realizzato, inoltre presenta le fasi di progettazione e sviluppo della Cartella Cardiologica Virtuale.

Desidero dedicare questo spazio alle persone che mi hanno sostenuto e hanno contribuito alla conclusione di questo percorso di studi.

Ringrazio la mia famiglia, che mi ha dato la possibilità di intraprendere la carriera universitaria e, sostenendomi durante le difficoltà, mi ha aiutato a terminarla serenamente.

Un ringraziamento speciale alla mia fidanzata Debora, che crede in me, ed è sempre pronta ad ascoltarmi, supportarmi e darmi un consiglio durante i momenti di incertezza e di criticità.

Grazie a te ho sempre ritrovato la forza e la motivazione per raggiungere i miei obiettivi nonostante i diversi ostacoli che si sono presentati. Grazie di essere sempre al mio fianco.

Desidero infine ringraziare tutti i miei amici, i miei colleghi di corso e tutte le persone che, grazie ai loro consigli e alla loro positività, hanno contribuito a rendere meravigliosa e costruttiva la mia esperienza universitaria.

Indice

1	Introduzione	2
1.1	Cartella Cardiologica Virtuale	2
1.1.1	Cartella Clinica	2
1.1.2	Rischio Cardiologico	2
1.2	Stato dell'arte	5
1.3	Analisi e Progettazione Del Software	14
1.3.1	UML - Unified Modeling Language	14
1.4	Design Pattern	16
1.4.1	MVC - Model View Controller	16
2	Tecnologie Utilizzate	18
2.1	IDE	18
2.1.1	Eclipse JEE	18
2.2	Framework	18
2.2.1	Spring	19
2.2.2	Hibernate	21
2.2.3	AngularJS	22
3	Analisi Dei Requisiti	23
3.1	Specifiche Generali	23
3.2	Requisiti Funzionali	24
3.3	Requisiti non Funzionali	26
3.4	Casi d'uso	27
3.4.1	Login	28
3.4.2	Visualizza Visita	29
3.4.3	Compilazione Form Inserimento Medico	30
3.4.4	Compilazione Form Visita	32
3.4.5	Inserimento Referto	33
4	Progettazione	35
4.1	Client	35
4.1.1	Interfaccia	35
4.2	Server	39
4.2.1	Diagramma delle Classi	39
4.3	Base Di Dati	40
5	Conclusioni e Sviluppi Futuri	43
	Bibliografia	45

Elenco delle figure

1.1	Carta Cardiologica delle Donne Diabetiche	3
1.2	Carta Cardiologica delle Donne Non Diabetiche	3
1.3	Carta Cardiologica degli Uomini Diabetici	4
1.4	Carta Cardiologica degli Uomini Non Diabetici	4
1.5	Indice del Rischio Cardiologico	4
1.6	Rappresentazione grafica del Model View Controller	16
2.1	Architettura del framework Spring	20
2.2	Frammento di codice dell'applicazione Cartella Cardiologica Virtuale . .	21
2.3	Script di AngularJS utilizzato nella Cartella Cardiologica Virtuale	22
3.1	Diagramma Dei Casi D'Uso	27
3.2	Diagramma Di Sequenza del Login	28
3.3	Diagramma Di Sequenza del caso d'uso: Visualizza Visita	29
3.4	Diagramma delle Attività del caso d'uso: Visualizza Visita	30
3.5	Diagramma Di Sequenza del caso d'uso: Compilazione Form Inserimento Medico	31
3.6	Diagramma delle Attività del caso d'uso: Compilazione Form Inserimento Medico	31
3.7	Diagramma Di Sequenza del caso d'uso: Compilazione Form Visita	32
3.8	Diagramma delle Attività del caso d'uso: Compilazione Form Visita	33
3.9	Diagramma Di Sequenza del caso d'uso: Inserimento Referto	34
3.10	Diagramma delle Attività del caso d'uso: Inserimento Referto	34
4.1	Homepage	36
4.2	Area privata del Medico - Lista dei Pazienti	36
4.3	Cartella Clinica Del Paziente	37
4.4	Area privata del Paziente	37
4.5	Rischio Cardiologico del Paziente	38
4.6	Diagramma Delle Classi	39
4.7	Diagramma E-R	40
4.8	Modello Relazionale	41

Capitolo 1

Introduzione

L'elaborato presenta le fasi di progettazione e sviluppo dell'applicazione realizzata durante l'attività di stage svolta presso l'azienda Sync Lab Srl.

1.1 Cartella Cardiologica Virtuale

Il progetto in questione ha come scopo l'informatizzazione della cartella clinica ponendo particolare rilevanza al rischio cardiologico.

1.1.1 Cartella Clinica

La cartella clinica è un documento sanitario il cui fine è quello di verbalizzare l'attività del reparto ospedaliero.

In particolare essa rappresenta il mezzo mediante il quale viene documentato il decorso clinico di ogni degente e contiene l'intera documentazione del ricovero, ovvero tutti i referti medici e tutti i report delle visite, che vengono effettuate al paziente durante la degenza.

1.1.2 Rischio Cardiologico

Il progetto in questione pone particolare attenzione al rischio cardiologico e al suo calcolo.

Il *rischio cardiovascolare*, o *rischio cardiologico*, esprime la probabilità che una persona vada incontro ad una malattia inerente il cuore o i vasi sanguigni (ad esempio un ictus o un infarto del miocardio).

Tale rischio viene calcolato mediante l'utilizzo di strumento denominato Carta Del Rischio.

Carte del Rischio Cardiovascolare

Le Carte del rischio cardiovascolare sono classi implementate in base ad alcuni fattori:

- *Sesso*: definito in Donne e Uomini;
- *Età*: definita negli intervalli: 40-49, 50-59, 60-69 anni di età;
- *Pressione Arteriosa Sistolica*: suddivisa negli intervalli: 90 mmHg - 130 mmHg, 130 mmHg - 150 mmHg, 150 mmHg - 170 mmHg, 170 mmHg - 200 mmHg.

- *Colesterolemia Totale*: suddivisa negli intervalli: 130 mg/dl - 174 mg/dl, 174 mg/dl - 213 mg/dl, 213 mg/dl - 252 mg/dl, 252 mg/dl - 291 mg/dl, 291 mg/dl - 320 mg/dl.
- *Abitudine al Fumo*: si considera non fumatore colui che non fuma da almeno 12 mesi;
- *Diabete*: definito nelle due categorie: Paziente Diabetico e Paziente Non Diabetico

È importante precisare che qualora i fattori del paziente non fossero compresi negli intervalli di Età, Colesterolemia Totale e Pressione Sistolica Arteriosa specificati, allora non sarà possibile calcolare il rischio cardiologico.

Le Carte del Rischio Cardiologico vengono suddivise in quattro categorie in base al sesso e alla presenza o meno del diabete:

- Donna Non Diabetica
- Donna Diabetica
- Uomo Non Diabetico
- Uomo Diabetico

Ognuna di queste categorie viene ulteriormente suddivisa in fumatori e non fumatori.

Di seguito vengono riportate le carte del rischio cardiovascolare e il livello di rischio ad esse associato.

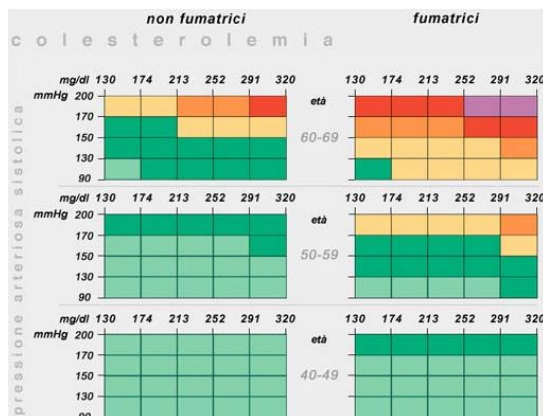


Figura 1.1: Carta Cardiologica delle Donne Diabetiche

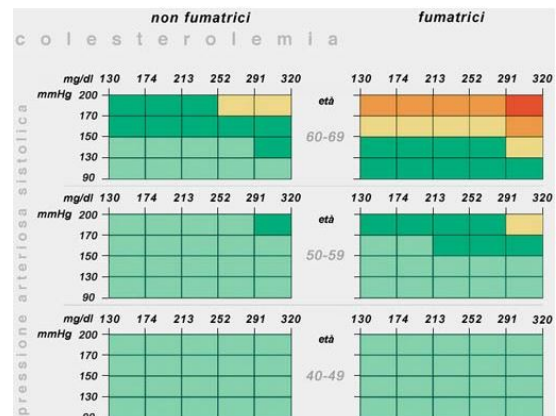


Figura 1.2: Carta Cardiologica delle Donne Non Diabetiche

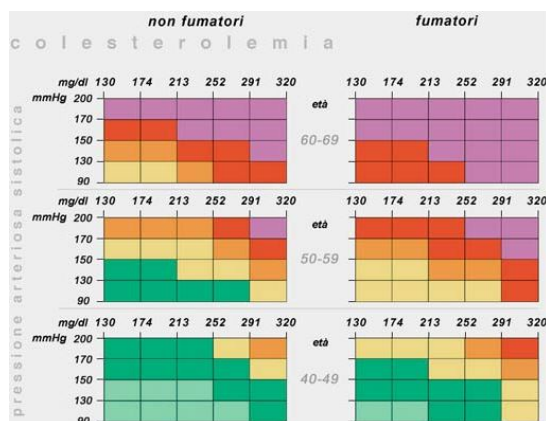


Figura 1.3: Carta Cardiologica degli Uomini Diabetici

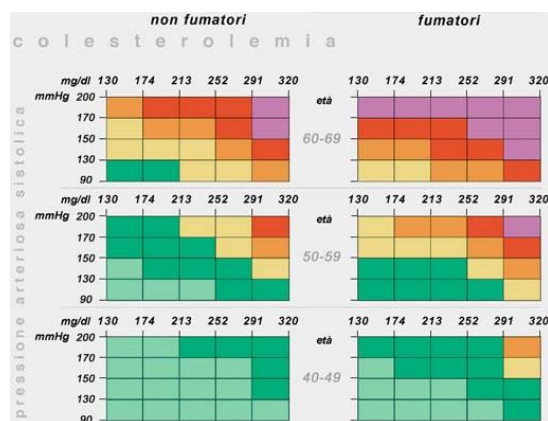


Figura 1.4: Carta Cardiologica degli Uomini Non Diabetici

rischio MCV VI		oltre 30%
rischio MCV V		20% - 30%
rischio MCV IV		15% - 20%
rischio MCV III		10% - 15%
rischio MCV II		5% - 10%
rischio MCV I		meno 5%

Figura 1.5: Indice del Rischio Cardiologico

In conclusione, la Cartella Cardiologica Virtuale permette di informatizzare la cartella clinica ed offre la possibilità di calcolare la percentuale di rischio cardiologico del paziente in base alle Carte del Rischio Cardiologico sopra descritte.

⁰Le Figure 1.1, 1.2, 1.3, 1.4, 1.5 sono tratte dal sito web "www.cuore.iss.it"

1.2 Stato dell'arte

Introduzione

Al giorno d'oggi la tecnologia informatica influenza ogni aspetto della nostra vita, imponendo dei cambiamenti nelle strutture organizzative di ogni tipologia, tra cui l'ambito sanitario.

Negli ultimi anni ha iniziato ad assumere particolare importanza la *Sanità digitale*, ovvero l'utilizzo degli strumenti informatici e di telecomunicazione a vantaggio della salute umana e del sistema sanitario.

Tutto ciò è confermato dalle aziende sanitarie che attualmente hanno come principale area di investimento il settore delle innovazioni tecnologiche, tra le quali troviamo la *cartella clinica elettronica*.

La medicina è basata su un alto numero di informazioni, in particolare quelle riguardanti i pazienti che devono essere costantemente memorizzate e comparate tra di loro affinché si possano migliorare le cure per le diverse patologie.

Le informazioni inerenti i degenti vengono inserite all'interno della cartella clinica cosicché più professionisti sanitari possano accedere a tali dati con lo scopo di studiarli.

La cartella clinica elettronica è uno strumento innovativo che offre la possibilità di dematerializzare la versione cartacea e facilita l'accesso alle informazioni sia da parte dei pazienti, sia da parte degli esperti che si occupano di analizzarle.

Negli ultimi anni questa tecnologia ha assunto un'importanza rilevante ed esistono numerosi software che svolgono la funzione di *EHR, Electronic Health Records*.

EHR - Electronic Health Records

L' EHR, Electronic Health Records, è una raccolta digitale dei documenti sanitari di un paziente e include informazioni come il sesso, l'età, la storia clinica, il risultato di visite e test di laboratorio effettuati sul degente.

In sintesi la Electronic Health Records è uno strumento che ha la funzione di digitalizzare la cartella clinica.

Tale tecnologia offre un beneficio molto importante, infatti permette agli operatori sanitari di condividere questi documenti digitali all'interno di uno stesso ospedale o sistema sanitario.

In tal modo i medici hanno la possibilità di conoscere le eventuali terapie seguite dal paziente ed evitare di prescrivere nuovi farmaci che potrebbero causare problemi se assunti insieme ad altri medicinali.

La condivisione dei dati favorita dall'EHR semplifica la collaborazione tra gli operatori sanitari e, avendo accesso a tutte le informazioni del paziente, possono stabilire più velocemente un piano di cura con un rischio minore di errori.

L'EHR inoltre garantisce la sicurezza dei dati sia in termini di consultazione, in quanto è possibile accedervi solamente se si è in possesso delle credenziali appropriate, sia in termini di memorizzazione, in quanto i documenti in formato elettronico hanno una probabilità quasi nulla di essere smarriti.

Implementazione dell'EHR

La tecnologia Electronic Health Records nacque in America durante gli anni 70 all'interno degli ospedali accademici, i quali sperimentavano nuove modalità per memorizzare le informazioni sanitarie.

L'EHR inizia ad essere sviluppata ed implementata durante gli anni 90, con l'avvento di HL7, Health Level Seven, ovvero un'associazione internazionale che si occupa di gestire gli standard per la sanità.

HL7 viene fondata nel 1987 con lo scopo di fornire un quadro completo e standard per lo scambio, la condivisione e il recupero delle informazioni sanitarie elettroniche.

A partire da quel periodo ci furono numerose strutture sanitarie che iniziarono ad adottare la tecnologia EHR per la memorizzazione delle informazioni riguardanti i degen- ti e tra le prime vi è la Cleveland Clinic, un centro medico situato in Ohio negli USA, che utilizza tutt'ora un software con la funzione di Electronic Health Records sviluppato dalla Epic System Corporation per digitalizzare le cartelle cliniche dei propri pazienti.

Tra le principali strutture sanitarie che hanno adottato la tecnologia in questione vi è anche la Mayo Clinic, un centro di ricerca medica con sede a Rochester situata nel Minnesota, negli USA.

Questa clinica utilizza il software HealthKit che offre la funzionalità di EHR ed è stato sviluppato dalla collaborazione tra Apple e Epic Systems Corporation.

Questa applicazione, oltre a garantire la digitalizzazione della cartella clinica, permette di memorizzare le informazioni raccolte da dispositivi indossabili come i braccialetti elettronici.

Tali dati interagiscono con il sistema EHR della cartella clinica e nel caso di letture anoma- le vengono attivate una serie di raccomandazioni presenti all'interno dell'app HealthKit utili per il paziente.

Studi effettuati sull'EHR

La Electronic Health Records è utilizzata da diversi anni nell'ambito sanitario e il suo impiego ha dato la possibilità di testare a fondo tale strumento, facendo emergere sia gli aspetti positivi, già descritti nelle sezioni precedenti, sia quelli negativi.

In questa sezione verranno presentati degli studi in merito ad alcune caratteristiche osservate in seguito all'utilizzo della tecnologia EHR e delle cartelle cliniche elettroniche in generale.

Usabilità

Il termine usabilità indica il grado in cui un prodotto può essere usato da particolari utenti per raggiungere certi obiettivi con efficacia, efficienza e soddisfazione in uno specifico contesto d'uso ¹.

In seguito vengono presentati alcuni studi inerenti l'usabilità della cartella clinica elettronica.

La tecnologia EHR viene utilizzata quotidianamente nelle strutture sanitarie e, in questi luoghi emergono maggiormente le limitazioni di tale strumento.

Uno studio [11] recente mette in risalto il malcontento tra i medici di base riguardo questo mezzo di archiviazione delle informazioni.

Essi segnalano la presenza di un'eccessiva quantità di dati inutili e presentati in un formato antiquato all'interno della cartella clinica elettronica.

I medici utilizzano l'Electronic Health Records per acquisire le informazioni inerenti la situazione clinica di un paziente, infatti gli consente di conoscere in anticipo le problematiche della persona affinché formulino una diagnosi nel minor tempo possibile.

Il personale sanitario esige quindi dei dati diretti e mirati alla cura del degente.

Un primo passo per migliorare la presentazione delle informazioni è quello di determinare il materiale della cartella clinica del paziente di cui i dottori hanno bisogno.

A tal proposito in questo studio sono stati intervistati 16 medici di base durante la preparazione necessaria per effettuare le visite presso il proprio ufficio, con lo scopo di comprendere meglio le loro problematiche nella visualizzazione delle informazioni all'interno dell'EHR.

Al termine delle interviste è emerso che i dottori evidenziano spesso la mancata corrispondenza delle informazioni presenti rispetto alle loro esigenze.

Nell'Electronic Health Records sono presenti troppi dati, spesso ridondanti, che creano confusione causando il rallentamento del flusso di lavoro.

In conclusione, sarebbe meglio riprogettare la cartella clinica elettronica inserendo delle interfacce che mostrano le informazioni necessarie riguardo il paziente, semplificando e velocizzando il lavoro dei medici.

L'utilizzo delle cartelle cliniche elettroniche è stato analizzato anche durante uno studio, presentato nell'articolo *"Electronic medical record systems: Decision support examination framework for individual, security and privacy concerns using multi-perspective analysis"* [5], effettuato presso cinque ospedali pubblici malesi.

Sebbene le strutture sanitarie utilizzano quotidianamente questo strumento, gli operatori hanno difficoltà a fidarsi di tale sistema in quanto vi sono delle incertezze riguardo le questioni di privacy e sicurezza inerenti le informazioni.

Lo studio in questione ha l'obiettivo di indagare con precisione sui fattori che influenzano l'utilizzo delle cartelle cliniche elettroniche all'interno degli ospedali pubblici.

A tal fine sono stati intervistati quattro gruppi di professionisti sanitari che hanno lavorato presso cinque strutture ospedaliere pubbliche malesi ovvero medici, farmacisti, infermieri e dipendenti di laboratorio.

¹Definizione tratta dal dizionario Treccani

Per condurre questa indagine è stato utilizzato il modello UTAUT (Unified Theory of Acceptance and Use of Technology) [3], uno strumento per la valutazione dell'accettazione di una tecnologia da parte degli utenti.

È stato inoltre utilizzato l'approccio MCDM (Multiple Criteria Decision Making) [20], ovvero un metodo che permette di classificare i fattori che influenzano l'utilizzo dello strumento in esame.

Dopo aver raccolto e analizzato tutti i dati sono emersi i seguenti risultati:

Esistono quattro fattori positivi riguardo l'utilizzo delle cartelle cliniche elettroniche, che sono: l'integrità dei dati, la riservatezza, il non ripudio e la facilità dell'accesso ai dati.

Sono altresì emersi alcuni fattori negativi riguardo la tecnologia implementata, ad esempio l'utilizzo non autorizzato delle informazioni e la presenza di errori all'interno dei dati.

Questo studio contribuisce alla Sanità Digitale infatti i risultati assumono particolare rilevanza in quanto offrono un'ulteriore conoscenza per la realizzazione delle cartelle cliniche elettroniche, mettendo in evidenza le difficoltà riscontrate dagli utilizzatori dell'attuale tecnologia.

Alcune problematiche inerenti l'EHR vengono riscontrate anche in Danimarca dove è stata condotta una ricerca [9] riguardo l'utilizzo delle cartelle cliniche per l'assistenza a domicilio.

Al termine dell'indagine è emerso che l'uso delle cartelle cliniche elettroniche nell'assistenza a lungo termine è rimasto indietro rispetto all'impiego di questo strumento negli ospedali.

Tale studio ha evidenziato la necessità di aggiornare la documentazione presente all'interno dell'Electronic Health Records in quanto « Il rischio predominante per la sicurezza dei pazienti è la documentazione incompleta » comportando un'assistenza inadeguata sia per le case di cura sia per i pazienti assistiti a domicilio.

Nell'articolo *"Impact of an electronic health record on task time distribution in a neonatal intensive care unit"* [4] viene descritta un'indagine condotta in merito all'utilizzo dell'Electronic Health Records presso quattro unità di maternità irlandesi.

Come evidenziato nello scritto « gli operatori sanitari sono preoccupati che le cartelle cliniche elettroniche possano aumentare il tempo dedicato alla documentazione e alle attività correlate ai farmaci ».

L'obiettivo era quello di determinare l'effettivo impatto della cartella clinica elettronica sul tempo in una terapia intensiva neonatale.

I risultati di tale ricerca, condotta su 63 medici e infermieri, hanno determinato che non ci sono state modifiche significative alla distribuzione dei tempi nello svolgimento dei normali compiti del personale sanitario.

Condivisione e sicurezza delle informazioni

Un aspetto importante riguardante le cartelle cliniche elettroniche interessa l'ambito scientifico, ovvero la condivisione dei dati.

Uno studio recente, trattato nell'articolo *"Ensuring electronic medical record simulation through better training, modeling, and evaluation"* [21], ha confermato l'enorme utilità che questa tecnologia assume nel campo della medicina, in quanto permette la condivi-

sione dei dati sia per scopi clinici, sia per scopi di studio scientifico, ma i regolamenti riguardanti il rispetto per la privacy limitano la condivisione di tali informazioni creando quindi un ostacolo per l'accesso ai documenti necessari agli studiosi.

Nell'indagine in questione è stato analizzato questo problema mediante la simulazione della condivisione dei dati tra alcune cartelle cliniche elettroniche.

Lo scopo era quello di dimostrare la possibilità di condividere un maggior quantitativo di informazioni rispettando le regole riguardanti la privacy.

È stata quindi effettuata una simulazione mediante l'utilizzo di alcune reti generative avversarie, note come GAN [10], adottando degli approcci direttamente sui dati condivisi, ovvero sono state implementate delle modalità che permettono la condivisione delle informazioni senza violare la privacy.

Durante lo studio sono emerse però alcune difficoltà riguardo l'utilizzo delle GAN, in quanto non consentono di effettuare un approccio corretto per il trattamento dei dati senza infrangere le regole di privacy.

La soluzione a questo inconveniente è stata l'introduzione di un sistema di filtraggio appositamente realizzato per le reti generative avversarie, in modo da poter simulare lo scambio dei dati senza intaccare la loro riservatezza.

Il modello proposto ha migliorato la facilità di acquisizione dei dati della cartella clinica elettronica senza violare la privacy di essi e, ha dimostrato che la simulazione dello scambio di informazioni contenute all'interno delle cartelle, realizzata tramite l'ausilio delle reti GAN, può essere migliorata adottando dei criteri di modellazione maggiormente appropriati.

L' utilizzo delle cartelle cliniche elettroniche comporta la necessità di conoscere le modalità con cui devono essere trattati i dati in esse contenuti.

L'Electronic Health Records è soggetta a questioni legali inerenti le informazioni, ad esempio i regolamenti per la privacy, e la proprietà dei dati contenuti.

Vi è inoltre la mancanza di uno standard per le cartelle cliniche elettroniche, il che crea ulteriore disinformazione riguardo tale strumento.

In Grecia è stato condotto uno studio [6] con lo scopo di comprendere le opinioni, la conoscenza e le preoccupazioni etiche delle persone in merito all'uso delle cartelle cliniche elettroniche.

Questa indagine è stata effettuata tra numerosi studenti e ben 36 associazioni mediche presenti nella Penisola Ellenica.

Agli intervistati è stato chiesto di compilare un questionario contenente delle domande relative alle cartelle cliniche elettroniche e quindi di indicare le conoscenze personali riguardo i vantaggi, i rischi e le questioni etiche in merito a tale tecnologia.

In seguito all'analisi delle informazioni acquisite è emerso che la maggior parte degli intervistati, sia il personale medico sia gli studenti di altre facoltà, riconoscono i numerosi vantaggi offerti dall'EHR ma reputano necessario che vengano fornite sufficienti garanzie riguardo la privacy e la sicurezza delle informazioni in esse contenute.

Il tema della riservatezza dei dati contenuti all'interno delle cartelle cliniche elettroniche è stato trattato nello scritto *"Secure and Decentralized Management of Health Records"* [17], secondo il quale « le cartelle cliniche memorizzate in un archivio centralizzato potrebbero essere accessibili da varie agenzie autorizzate ».

La questione importante è stabilire le modalità con cui le informazioni presenti all'interno di esse possano essere condivise rispettando l'integrità e la privacy.

Tale studio ha proposto l'archiviazione delle cartelle cliniche crittografate all'interno di un IPFS per garantire la riservatezza, la disponibilità, il rispetto della privacy e la sicurezza contro possibili attacchi da parte di hacker, sia interni che esterni al sistema, nei confronti dei dati.

L'IPFS [12] , InterPlanetary File System, è un modello di archiviazione distribuito progettato per memorizzare le informazioni e garantire delle transazioni condivise. Questo sistema protegge le informazioni in esso contenute applicando delle tecniche di crittografia.

Quantità dei dati

L'Electronic Health Records consente di memorizzare tutti i dati dei pazienti, tuttavia è importante stimare la quantità di informazioni che vengono archiviate quotidianamente.

Recentemente, nel periodo compreso tra l'1 Febbraio 2020 e il 15 Febbraio 2020, è stato condotto uno studio con l'obiettivo di quantificare i dati sanitari generati dai pazienti.

L'indagine, illustrata nell'articolo "*Quantifying Electronic Health Record Data: A Potential Risk for Cognitive Overload*" [7], è stata svolta presso il reparto di terapia intensiva cardiovascolare pediatrica su trenta pazienti.

I dati sono stati acquisiti dalla cartella clinica elettronica dei degenti e raggruppati in base all'orario di acquisizione generando quindi due gruppi: uno costituito da tutte le informazioni registrate durante il turno di giorno e uno costituito da tutte le informazioni acquisite durante il turno di notte.

In seguito a delle opportune analisi è emerso che i pazienti generano mediamente 4380 nuovi dati durante il giorno e 16060 durante la notte.

Tali risultati evidenziano una problematica importante: il sovraccarico cognitivo.

Questo termine indica la presenza di una quantità elevata di informazioni che potrebbe mettere in seria difficoltà i medici, infatti l'enorme carico di dati rischierebbe di non permettergli di focalizzare l'attenzione su un problema e di conseguenza non sarebbero in grado di prendere decisioni opportune per la cura dei pazienti.

In merito ai dati conservati nelle cartelle cliniche elettroniche, nell'articolo "*Security and Privacy of E-health Data*" [1] viene stimato che la quantità di dati sanitari generati a livello globale sia stata di 500 petabyte nel 2012 e sono previsti 25.000 petabyte nel 2020.

Questa crescita consistente di informazioni è dovuta alla digitalizzazione dei referti e di tutti i dati multimediali che devono essere acquisiti e archiviati.

Lo scritto riporta inoltre che « Queste informazioni sono in continua crescita con conseguente accumulo di grandi volumi di dati ».

Rischio Cardiologico nell'EHR

Le cartelle cliniche elettroniche contengono grandi quantità di informazioni inerenti i pazienti e tali dati potrebbero essere utilizzati per automatizzare il calcolo del rischio cardiologico.

Questo indice offre un'informazione importante riguardo la salute del paziente ed è fondamentale per il medico durante la scelta delle terapie da seguire.

Il calcolo automatico del rischio cardiologico di tutti i pazienti è essenziale in quanto permette ai medici di dare la precedenza a coloro che sono in maggior pericolo.

Tuttavia le informazioni contenute nella cartella clinica elettronica potrebbero non essere precise e di conseguenza sarebbe necessario confrontarle con i dati presenti sugli esami cartacei.

A tal proposito è stato svolto uno studio con l'obiettivo di valutare l'accuratezza del rischio cardiologico calcolato automaticamente.

L'indagine, presentata nell'articolo "*Electronic health record-based cardiac risk assessment and identification of unmet preventive needs*" [14], è stata effettuata su 23111 persone nella fascia di età compresa tra 20 e 79 anni.

I candidati sono stati suddivisi in 4 gruppi di rischio e dall'EHR di ognuno sono state acquisite informazioni riguardanti il sesso, l'età, la pressione sistolica, la colesterolemia e l'utilizzo di farmaci.

Per ogni candidato è stato calcolato il rischio cardiologico sia automaticamente che manualmente.

In seguito all'analisi dei risultati è emerso che tra il calcolo automatico e quello manuale vi è un bassa percentuale di errore.

Ciò dimostra che i dati presenti all'interno dell'EHR possono essere utilizzati per calcolare correttamente il rischio cardiologico permettendo quindi di individuare più velocemente i pazienti con una maggiore probabilità di andare incontro ad una malattia cardiologica e intervenire tempestivamente.

L'importanza di questa funzionalità viene confermata anche da uno studio [19] recente, il cui obiettivo era quello di dimostrare la validità dell'FRS, Framingham Risk Score. Quest'ultimo è un algoritmo utilizzato per calcolare il rischio cardiovascolare.

È stato determinato l'FRS a partire dai dati presenti all'interno della cartella clinica elettronica di 84116 pazienti.

Al termine dell'indagine è emerso che il Framingham Risk Score è tutt'ora uno strumento valido per calcolare il rischio cardiologico.

L'aspetto importante è che tale algoritmo può essere implementato all'interno dell'EHR per il calcolo automatico dell'indice evidenziando ulteriormente la convenienza di inserire tale automatismo all'interno della cartella clinica elettronica.

Si ritiene necessario evidenziare l'utilità del Framingham Risk Score.

Recentemente in Canada è stata condotta un'analisi, descritta nell'articolo "*Early and surgical menopause associated with higher Framingham Risk Scores for cardiovascular disease in the Canadian Longitudinal Study on Aging*" [16], in merito all'FRS in quanto nelle donne il rischio di malattie cardiovascolari è più alto nel periodo successivo alla menopausa.

Lo scopo di tale studio era quello di valutare l'associazione tra la menopausa e il Framingham Risk Score nelle donne.

Al termine dell'indagine è stata confermata tale relazione evidenziando la necessità per le donne in post-menopausa di evitare interventi chirurgici.

Ulteriori utilizzi dell'EHR

L'utilizzo della Electronic Health Records assume particolare rilevanza anche nell'analisi di altre tipologie di rischi.

La cartella clinica elettronica ad esempio è molto utile in un momento di difficoltà come quello attuale, in cui ci si trova ad affrontare la pandemia globale di COVID-19.

Uno studio recente, illustrato nell'articolo *"COVID-19 risk and outcomes in patients with substance use disorders: analyses from electronic health records in the United States"* [18], ha trattato le problematiche causate dall'uso di oppioidi in relazione ai disturbi polmonari provocati dal virus.

Nello scritto viene esplicitato « Questo è uno studio retrospettivo sui dati delle cartelle cliniche elettroniche (EHR) di 73.099.850 pazienti » evidenziando l'utilità di tale tecnologia per l'analisi di varie tipologie di rischio.

Un settore che beneficia delle funzionalità offerte dalla cartella clinica elettronica è l'oncologia.

Secondo l'articolo *"Opportunities and challenges in leveraging electronic health record data in oncology"* [2] l'utilizzo diffuso della tecnologia EHR e la quantità crescente di informazioni riguardo i pazienti potrebbe rivoluzionare la ricerca clinica in oncologia.

Nello scritto si afferma che « la ricerca sarà probabilmente più efficiente e potenzialmente più accurata » e ciò viene confermato dai risultati ottenuti durante uno studio [8] il cui scopo era quello di descrivere la completezza della documentazione e la soddisfazione dell'utente riguardo le cartelle cliniche elettroniche prima e dopo l'introduzione del sistema EHR.

A tal proposito è stato condotto un sondaggio tra 90 pazienti oncologici oltre che medici e infermieri.

Al termine dell'indagine è emerso che l'uso dell'Electronic Health Records migliora la completezza della documentazione medica e dell'ordine chemioterapico e aumenta la soddisfazione degli utenti che utilizzano il sistema di cartelle cliniche elettroniche.

L'EHR risulta essere utile anche per la riabilitazione polmonare.

È stato effettuato uno studio in merito a questo argomento, descritto nell'articolo *"Effects of pulmonary rehabilitation on exacerbation number and severity in people with COPD: an historical cohort study using electronic health records"* [13], secondo il quale la riabilitazione polmonare ha dimostrato di ridurre i ricoveri ospedalieri.

L'aspetto rilevante è che i dati adoperati durante l'indagine sono stati acquisiti dalle cartelle cliniche elettroniche di 69089 pazienti evidenziando l'utilità di tale tecnologia.

Dai diversi studi presentati in questo capitolo si evince l'importanza che assume la cartella clinica elettronica e tale concetto viene ribadito nell'articolo *"Electronic Health Records Should Support Clinical Research"* [15].

In seguito all'analisi della Electronic Health Records e agli aspetti sia positivi che negativi offerti da questo strumento, si è scelto di realizzare la "Cartella Cardiologica Virtuale".

La web-app presentata in questo elaborato nasce con l'obiettivo di implementare le funzionalità della tecnologia EHR, ovvero garantire la digitalizzazione delle cartelle cliniche dei pazienti, con l'obiettivo di favorire il lavoro del personale medico, permettendo quindi un accesso semplice e veloce ai dati dei degenti mediante un'interfaccia chiara e intuitiva. L'applicazione realizzata introduce una nuova funzionalità oltre a quelle dell'EHR precedentemente trattate, ovvero offre la possibilità per il medico di calcolare il rischio cardiologico.

L'implementazione di tale funzione è importante in quanto, una volta effettuato l'accesso alla cartella del paziente, il personale sanitario può visionare non solo i documenti presenti ma anche l'indice che riporta la probabilità di sviluppo di malattie cardiovascolari nel degente e ciò influisce positivamente sulla velocità e la precisione nello stabilire un piano terapeutico per la persona.

1.3 Analisi e Progettazione Del Software

Il ciclo di vita del software si compone di diverse fasi, tra le quali assume una particolare importanza l'Analisi Del Software.

Il fine di quest'ultima è quello di definire le funzioni che il programma deve garantire e di fissare un modello di riferimento per l'elaborazione dell'applicazione.

In particolare l'Analisi Del Software presenta le fasi di:

- **Analisi dei requisiti**

Per *requisiti* si intendono le funzionalità che il software deve garantire. Essi si dividono in *Requisiti Funzionali*, ovvero i servizi offerti dal sistema e *Requisiti Non Funzionali*, ovvero i vincoli sui servizi offerti dal sistema.

- **Progettazione**

Per *Progettazione* si intende lo sviluppo di un modello di riferimento per l'implementazione del software.

In questo capitolo viene presentato il linguaggio UML, adottato per la realizzazione del modello.

- **Implementazione e realizzazione del software**

Avendo come riferimento le specifiche del progetto, i requisiti e il modello di riferimento, si procede alla *realizzazione* vera e propria del software.

In questo elaborato verranno presentate le fasi di Analisi e Progettazione del software sopra descritte.

1.3.1 UML - Unified Modeling Language

L' UML (Unified Modeling Language) è un linguaggio di progettazione realizzato per facilitare la modellazione dell'architettura dei software basati sulla *Programmazione ad oggetti*, come ad esempio le applicazioni implementate mediante il linguaggio di programmazione Java.

Questo strumento permette di descrivere, analizzare e modellare l'architettura di un software mediante l'utilizzo dei seguenti diagrammi:

- *Diagramma dei Casi D'uso*, ovvero la rappresentazione grafica degli scenari di utilizzo del sistema
- *Diagramma di sequenza*, ovvero la rappresentazione grafica della sequenza di interazioni che avvengono tra l'attore e il sistema in un determinato caso d'uso
- *Diagramma delle attività*, ovvero la modellazione del flusso di operazioni necessarie per implementare un determinato servizio offerto dall'applicazione

- *Diagramma delle Classi*, ovvero la rappresentazione grafica delle relazioni esistenti tra le diverse entità che compongono l'applicazione

In seguito verranno presentati tutti i diagrammi sopra elencati relativi ai diversi casi d'uso.

1.4 Design Pattern

I Design Pattern sono delle soluzioni progettuali generali applicabili a contesti simili.

Come definito dall'architetto Christopher Alexander:

“Ogni pattern descrive un problema che si ripete più e più volte nel nostro ambiente, descrive poi il nucleo della soluzione del problema, in modo tale che si possa riusare la soluzione un milione di volte, senza mai applicarla alla stessa maniera.”

L'architettura del software presentato in questo elaborato è definita mediante l'utilizzo del design pattern MVC.

1.4.1 MVC - Model View Controller

Il Model View Controller è un pattern architetturale molto diffuso che viene utilizzato principalmente nell'ambito della Programmazione orientata agli oggetti e nelle applicazioni web. L'MVC permette di realizzare l'indipendenza tra i principali livelli in cui è composto un sistema.

La figura 1.6 illustra una generica rappresentazione del funzionamento di quanto appena descritto.

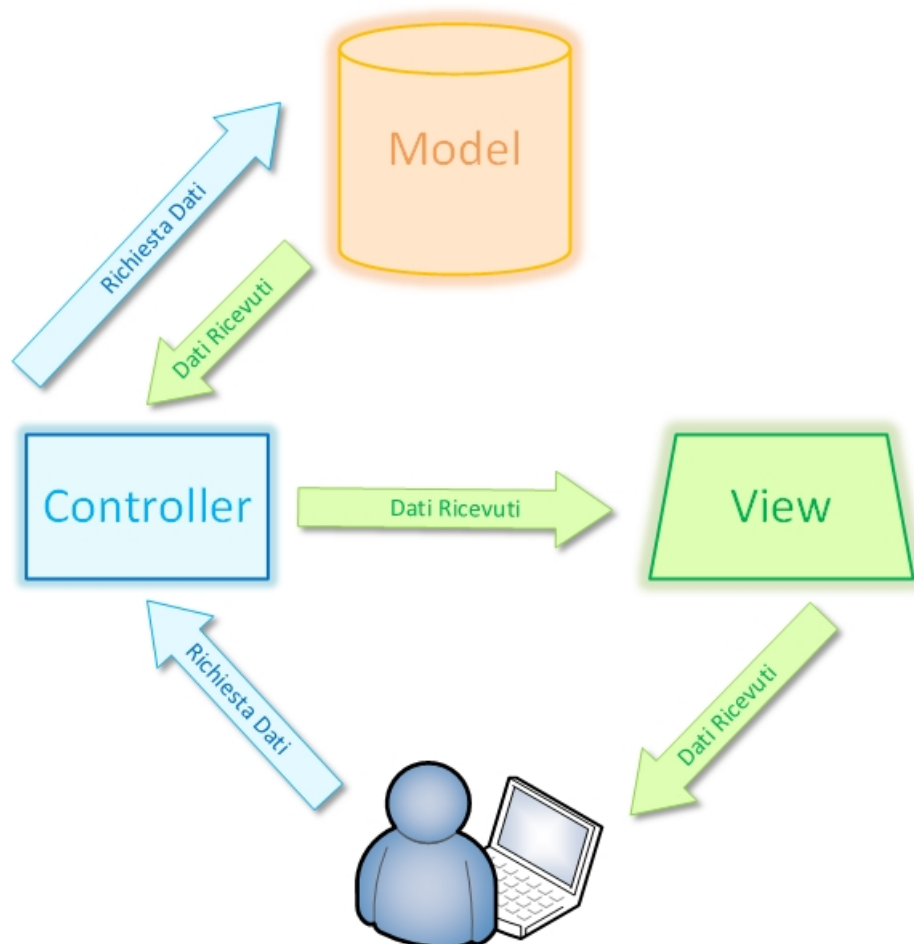


Figura 1.6: Rappresentazione grafica del Model View Controller

Come mostrato in figura 1.6 il Model View Controller è costituito da tre componenti principali:

- **Model**

Il Model rappresenta il modello principale dell'intera architettura in quanto ad esso è attribuita la gestione dei dati

- **View**

Il View, definito anche Livello di presentazione, è costituito da tutte le funzioni che interagiscono direttamente con l'utente, ovvero permette di definire le modalità con cui l'utilizzatore vede e interagisce con l'applicazione.

- **Controller**

Il Controller è il livello di connessione tra il Model e il View ed ha la funzione di ricevere l'input dall'utente e di restituire i dati da lui richiesti, attraverso il livello di presentazione, in base alle "regole" definite nel Model

Capitolo 2

Tecnologie Utilizzate

In questo capitolo vengono riportate le tecnologie utilizzate per la realizzazione del progetto presentato in tale elaborato.

2.1 IDE

Un IDE (Integrated Development Environment) è un ambiente di sviluppo integrato progettato per la realizzazione di applicazioni.

Generalmente un IDE è caratterizzato da un *editor* che agevola la scrittura di codice software evidenziando eventuali errori di sintassi, e da un *debugger* che permette di testare l'applicazione realizzata.

2.1.1 Eclipse JEE

L'IDE utilizzato per la realizzazione dell'applicazione descritta in questo elaborato è Eclipse JEE, particolare la versione Eclipse JEE - 2019.

Tale IDE è open source e può essere adattato alle diverse esigenze di programmazione aggiungendo dei plug-in.

2.2 Framework

Un Framework è una struttura logica il cui scopo è quello di fornire degli strumenti per velocizzare il lavoro di programmazione.

Più semplicemente un framework, comunemente detto "piattaforma", è costituito da una serie di classi che vengono "richiamate" dal programmatore durante la stesura del codice. Tali classi implementano delle funzioni che possono essere utilizzate dallo sviluppatore nella costruzione dell'applicazione.

L'applicazione presentata in questo elaborato è stata sviluppata mediante il linguaggio di programmazione Java, con l'implementazione di due framework principali: Spring e Hibernate.

La realizzazione dell'aspetto grafico di tale applicazione ha inoltre richiesto l'utilizzo di un ulteriore framework: AngularJS, che viene illustrato nelle pagine successive insieme alle due piattaforme sopra nominate.

2.2.1 Spring

Spring è un modello di programmazione e configurazione per le applicazioni basate sul linguaggio Java.

Questo framework viene rilasciato per la prima volta nel 2003 con licenza Apache, e nasce come alternativa a JavaBeans.



Prima di continuare nella descrizione del framework Spring è opportuno fornire una breve descrizione di JavaBeans.

JavaBeans

Nel linguaggio di programmazione Java, per "*componente*" si intende una classe standard utilizzabile in più applicazioni.

Ciò permette alle componenti di essere definite una sola volta e di essere riutilizzate in contesti differenti, migliorando quindi la produttività.

Un *Java Bean* è una componente di Java, ovvero è una classe che ha delle caratteristiche fondamentali:

- tutti gli attributi sono di tipo *private*
 - ha un metodo costruttore privo di argomenti
 - per ciascun attributo vengono implementati i metodi **get** e **set** di tipo *public*.
-

Spring è caratterizzato da una struttura modulare, ovvero è costituito principalmente da cinque livelli:

- il livello *AOP* (*Aspect Oriented Programming*): garantisce la Separation of Concerns, ovvero ha l'obiettivo di separare le funzionalità trasversali dell'applicazione dalla logica di business. Questo livello comprende a sua volta i livelli DAO e ORM:
 - livello *DAO* (*Data Access Object*): è un pattern architetturale che ha lo scopo di separare le logiche di business da quelle di accesso ai dati
 - livello *ORM* (*Object Relational Mapping*): permette la persistenza dei dati, nello specifico, tramite strumenti software (ad esempio Hibernate), offre la possibilità di associare le tabelle presenti nel database ai corrispondenti oggetti in Java.
- livello *JEE* (*Java Platform Enterprise Edition*): insieme delle tecnologie che permettono di estendere le funzionalità del linguaggio Java alla programmazione web.
- livello *WEB*: implementa il pattern MVC (Model View Controller), presentato nel capitolo introduttivo, mediante il framework Spring MVC.
In particolare il framework Spring MVC permette di realizzare applicazioni web basate sul pattern MVC sfruttando l'AOP e l'Inversion Of Control.

Tali livelli sono costruiti a partire dalla componente "base" di Spring denominata Core Container. Questa componente è costituita dai moduli:

- *Core*: il modulo fondamentale che implementa la Dependency Injection
- *Bean*: il modulo che fornisce le classi factory per l'inizializzazione dei bean;
- *Context*: il modulo che fornisce l'accesso a tutti i bean;
- *SpEL*: il modulo che permette la manipolazione dei bean a runtime

La Figura 2.1 offre una rappresentazione grafica dell'architettura del framework Spring.

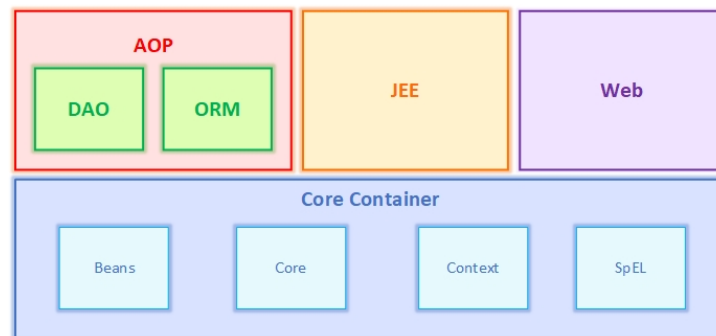


Figura 2.1: Architettura del framework Spring

Spring permette l'implementazione della Dependency Injection che è uno degli aspetti fondamentali dell' Inversion Of Control(IoC);

L'*Inversion Of Control* è una caratteristica dei framework che permette di spostare il controllo del flusso di sistema dal programmatore al framework.

La *Dependency Injection* è un'implementazione del Inversion Of Control la cui idea di base è quella di utilizzare un componente esterno per creare gli oggetti e le loro dipendenze e assemblarli mediante l'Injection.

2.2.2 Hibernate

Hibernate è un framework che offre un servizio di ORM (Object Relational Mapping).

Quest'ultima è una tecnica di programmazione che permette di effettuare il mapping di un qualsiasi oggetto, ad esempio un oggetto Java, su un database relazionale, comunemente denominato RDBMS (Relational Database Management System).

Hibernate si occupa quindi di gestire tutte le operazioni CRUD (Create, Update, Read, Delete) sul database autonomamente, semplificando di molto il lavoro del programmatore.



A livello di codice, il mapping è realizzato mediante le annotazioni `@Entity` e `@Table`.

In Figura 2.2 viene riportato un frammento di codice che mostra l'utilizzo delle annotazioni sopra indicate.

```
@Entity
@Table(name = "configurazioneRischio")
public class ConfigurazioneRischio {

    @Id
    @Column(name="idRischio")
    private String idRischio;
    @Column(name="sesso")
    private String sesso;
    @Column(name="etaMin")
    private int etaMin;
    @Column(name="etaMax")
    private int etaMax;
    @Column(name="diabete")
    private int diabete;
    @Column(name="fumo")
    private int fumo;
    @Column(name="pressioneMin")
    private int pressioneMin;
    @Column(name="pressioneMax")
    private int pressioneMax;
    @Column(name="colesteroloMin")
    private int colesteroloMin;
    @Column(name="colesteroloMax")
    private int colesteroloMax;
    @Column(name="percentuale")
    private int percentuale;
```

Figura 2.2: Frammento di codice dell'applicazione Cartella Cardiologica Virtuale

Con l'annotazione `@Entity` si indica che la classe definita (in questo caso "ConfigurazioneRischio") deve essere trattata come un bean, mentre `@Table` specifica che quella determinata entità deve essere trattata come una tabella del database.

In Figura 2.2 sono inoltre presenti le annotazioni `@Id` e `@Column`. Con `@Id` si specifica che l'attributo (in questo caso "idRischio") è una chiave primaria.

Mentre con l'annotazione `@Column` si indicano gli attributi della tabella.

2.2.3 AngularJS

AngularJS è un framework Javascript sviluppato da Google che fa la sua prima comparsa nel 2010.



Tale piattaforma è un framework open

source per le applicazioni web e adotta il pattern architetturale MVC.

AngularJS è stato realizzato con lo scopo di semplificare lo sviluppo del front-end delle web-app e ha la particolarità di estendere le funzionalità delle singole pagine HTML mediante l'utilizzo di determinate direttive.

Il framework descritto in questo capitolo è facilmente implementabile, infatti è sufficiente inserire lo script presentato in Figura 2.3 all'interno della pagina web.

```
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.6.9/angular.min.js"></script>
```

Figura 2.3: Script di AngularJS utilizzato nella Cartella Cardiologica Virtuale

Successivamente all'inserimento dello script è possibile usufruire di tutte le funzionalità offerte da questo framework mediante le diverse direttive. Alcune direttive di AngularJS sono:

- *ng-model*: direttiva che associa il valore di controlli HTML per i dati delle applicazioni;
- *ng-app*: direttiva che inizializza un'applicazione AngularJS;
- *ng-init*: direttiva che inizializza i dati delle applicazioni;
- *ng-show* e *ng-hide*: due direttive che consentono rispettivamente di visualizzare o nascondere elementi sulla pagina web in seguito alla valutazione di una espressione;

La direttiva *ng-show*, ad esempio, è stata utilizzata nella web-app descritta in questo elaborato.

Le seguenti figure 2.4 e 2.5, tratte dal codice dell'applicazione in questione, illustrano l'implementazione di tale direttiva.

```
<a href="#" ng-click="login = !login" class="btn btn-outline-primary">Login</a>
```

Figura 2.4

```
<div ng-show="login" class="container">
```

Figura 2.5

In particolare nella figura 2.4 viene mostrato il codice che permette di visualizzare il "contenitore" (definito mediante il codice mostrato in figura 2.5) in seguito al click sul link "Login".

Capitolo 3

Analisi Dei Requisiti

La Cartella Cardiologica Virtuale ha lo scopo di virtualizzare una cartella clinica permettendo ad un paziente di accedere alla propria area personale per visualizzare tutte le visite effettuate.

Tale applicazione fornisce ad un medico la possibilità di accedere in qualsiasi momento alla cartella clinica dei propri pazienti per visualizzare le loro visite, per calcolare il loro rischio cardiologico e caricare eventuali referti medici.

Di seguito verranno presentate le specifiche di tale progetto seguite da un'approfondita analisi dei requisiti.

3.1 Specifiche Generali

Il progetto trattato in questo elaborato presenta tre attori principali:

- Amministratore
- Medico
- Paziente

Il sistema ha le seguenti funzionalità principali:

- Registrazione degli utenti nel sistema con relativa assegnazione di username e password;
- Inserimento delle visite e dei referti da parte del medico nella cartella di un determinato paziente;
- Accesso del paziente alla propria area personale e visualizzazione delle visite effettuate.

3.2 Requisiti Funzionali

In questa sezione vengono analizzate nello specifico le mansioni dei diversi attori del sistema.

L'*amministratore*, dopo aver effettuato il login, dispone delle seguenti funzionalità:

- *Inserimento di un medico*: l'amministratore compila il form inserendo i dati anagrafici del medico;
- *Eliminazione di un medico*;
- *Inserimento di un nuovo amministratore*

Il *medico*, dopo essersi autenticato, dispone delle seguenti funzionalità:

- *Inserimento di un nuovo paziente*: il medico compila il form inserendo i dati anagrafici del paziente;
- *Visualizzazione della lista dei pazienti* registrati nel sistema;
- *Download della lista dei pazienti* registrati nel sistema: viene generato un file PDF contenente la lista di tutti i pazienti presenti nel sistema con i relativi dati anagrafici;
- *Accesso alla cartella di un paziente*: il medico può accedere alla cartella di un determinato paziente e disporre delle seguenti funzioni:
 - *Inserimento di una visita*: il medico compila il form della visita inserendo tutti i dati necessari (ad esempio il peso, l'altezza, e la circonferenza della vita del paziente) e successivamente il sistema memorizza tali informazioni nella cartella del paziente;
 - *Eliminazione di una visita*;
 - *Visualizzazione ed eventuale modifica della visita del paziente*: il medico, dopo aver selezionato una determinata visita all'interno della cartella del paziente può decidere di modificarla compilando il relativo form;
 - *Download della lista delle visite del paziente*: viene restituito un file di tipo PDF con la lista di tutte le visite effettuate al paziente;
 - *Inserimento di un referto medico*: il medico inserisce il referto (che generalmente può essere un file di tipo immagine o PDF) all'interno della cartella del paziente;
 - *Eliminazione di un referto medico*;
 - *Download di un referto medico* di un paziente: il referto, ovvero il file caricato precedentemente dal medico oppure dal paziente stesso, viene memorizzato sul dispositivo del medico;
 - *Calcolo del rischio cardiologico* di un paziente: quando il medico compila il relativo form inserendo i dati riguardanti il paziente (ad esempio la pressione e la colesterolemia) il sistema restituisce la percentuale di rischio per il soggetto interessato. Tale rischio viene inserito nella cartella del paziente. Il medico può calcolare in qualsiasi momento il rischio cardiologico del paziente e decidere se associarlo o meno alla sua cartella.

Il *paziente*, dopo aver effettuato il login, dispone delle seguenti funzionalità:

- *Registrazione nel sistema*: il paziente può registrarsi autonomamente nel sistema compilando il relativo form e inserendo i propri dati anagrafici;
- *Accesso alla propria cartella*: il paziente, dopo aver effettuato il login, può accedere alla propria cartella personale e da qui può:
 - *Visualizzare le proprie visite*;
 - *Inserire un referto*: il paziente può caricare nella propria cartella un referto medico (in genere si tratta di un file di tipo immagine o PDF)
 - *Scaricare un referto*: il referto verrà salvato sul dispositivo dell'utente
 - *Scaricare la lista delle visite*: il paziente può effettuare il download di tutte le visite che gli sono state effettuate; verrà generato un file PDF contenente tutte le informazioni inerenti le visite del paziente;
 - *Visualizzare il proprio rischio cardiologico*, se quest'ultimo è già stato calcolato dal medico (in caso contrario il paziente visualizzerà un messaggio di errore)
 - *Calcolare il proprio rischio cardiologico*: il paziente può compilare il form inserendo le opportune informazioni (ad esempio pressione e colesterolemia). Il rischio calcolato dal paziente però non può essere associato alla sua cartella in quanto si preferisce attendere la supervisione del medico.

3.3 Requisiti non Funzionali

Come è già stato esplicitato in precedenza, il sistema prevede tre attori principali: Amministratore, Medico e Paziente.

Per ognuno di essi vi è un livello di accesso differente.

L'operazione di autenticazione avviene mediante una pagina di login.

Il sistema riconosce il tipo di utente (ovvero amministratore, medico oppure paziente) e lo reindirizza all'opportuna pagina iniziale.

Un medico può essere inserito all'interno del sistema solamente da un amministratore.

Un paziente può essere inserito all'interno del sistema da un medico oppure si può registrare autonomamente.

Le operazioni di inserimento, ad esempio l'inserimento di un nuovo paziente oppure l'inserimento di una nuova visita, richiedono inizialmente la compilazione di un form in cui inserire i dati opportuni e, successivamente per completare l'operazione è necessario che l'utente faccia click sull'apposito pulsante di "salva" o "invia".

Il sistema verifica se i dati inseriti rispettano i vincoli definiti (ad esempio la tipologia di dato, o la mancanza del dato).

Nel caso in cui i vincoli del form non vengono soddisfatti allora il sistema restituisce un messaggio di errore.

In caso contrario il sistema provvede a salvare i dati inseriti all'interno del database.

L'operazione di download dei referti inseriti all'interno del sistema si conclude con il salvataggio del referto desiderato sul dispositivo dell'utente.

Le operazioni di download delle liste delle visite e delle liste dei pazienti generano un file pdf contenente tutte le informazioni rispettivamente delle visite e dei pazienti.

3.4 Casi d'uso

In questo capitolo vengono trattati i casi d'uso di: Login, Visualizzazione di una visita, Compilazione del form di una visita, Compilazione del form di inserimento di un medico e Inserimento di un referto.

La Figura 3.1 mostra in modo schematico i casi d'uso che si è scelto di trattare in questo capitolo, evidenziando gli attori di tali casi.

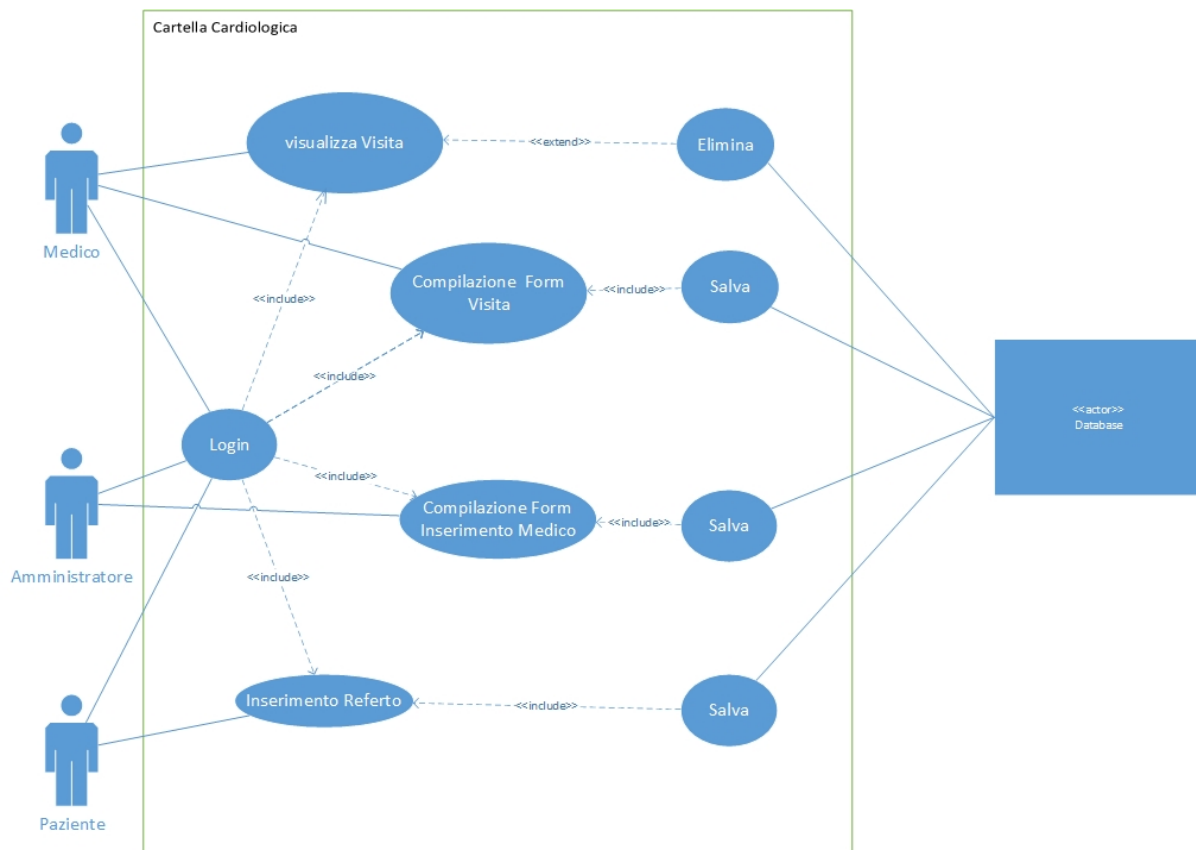


Figura 3.1: Diagramma Dei Casi D'Uso

Di seguito vengono approfonditi i casi d'uso sopra elencati e per ognuno si riportano i relativi diagrammi di sequenza e diagrammi delle attività.

3.4.1 Login

Nome: Login

Portata: Cartella Cardiologica Virtuale

Livello: Obiettivo Utente

Attore Primario: Amministratore, Medico, Paziente

Parti interessate e interessi: Un utente, ovvero un amministratore, un medico o un paziente, vuole accedere al sistema

Pre-condizioni: L'utente è registrato nel sistema

Garanzie di Successo: L'utente effettua il login e accede all'applicazione

Scenario principale di successo: L'utente effettua il login e accede alla sua area personale

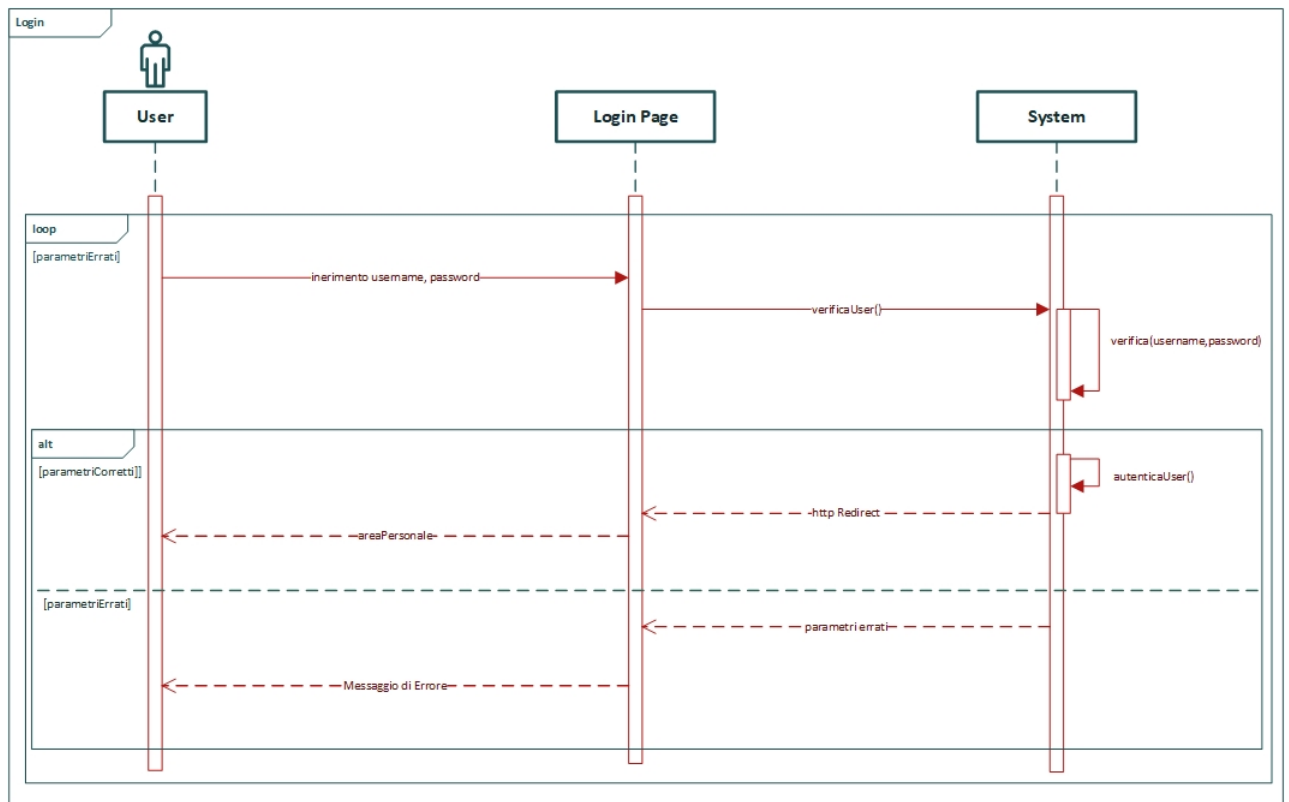


Figura 3.2: Diagramma Di Sequenza del Login

3.4.2 Visualizza Visita

Nome: Visualizza Visita

Portata: Cartella Cardiologica Virtuale

Livello: Obiettivo Utente

Attore Primario: Medico

Parti interessate e interessi: Il medico vuole visualizzare la visita di un paziente

Pre-condizioni: Il medico deve essere registrato nel sistema e deve aver effettuato il login

Garanzie di Successo: Il medico visualizza la visita del paziente

Scenario principale di successo:

- Il medico effettua il login
- Accede alla cartella del paziente
- Visualizza la visita del paziente

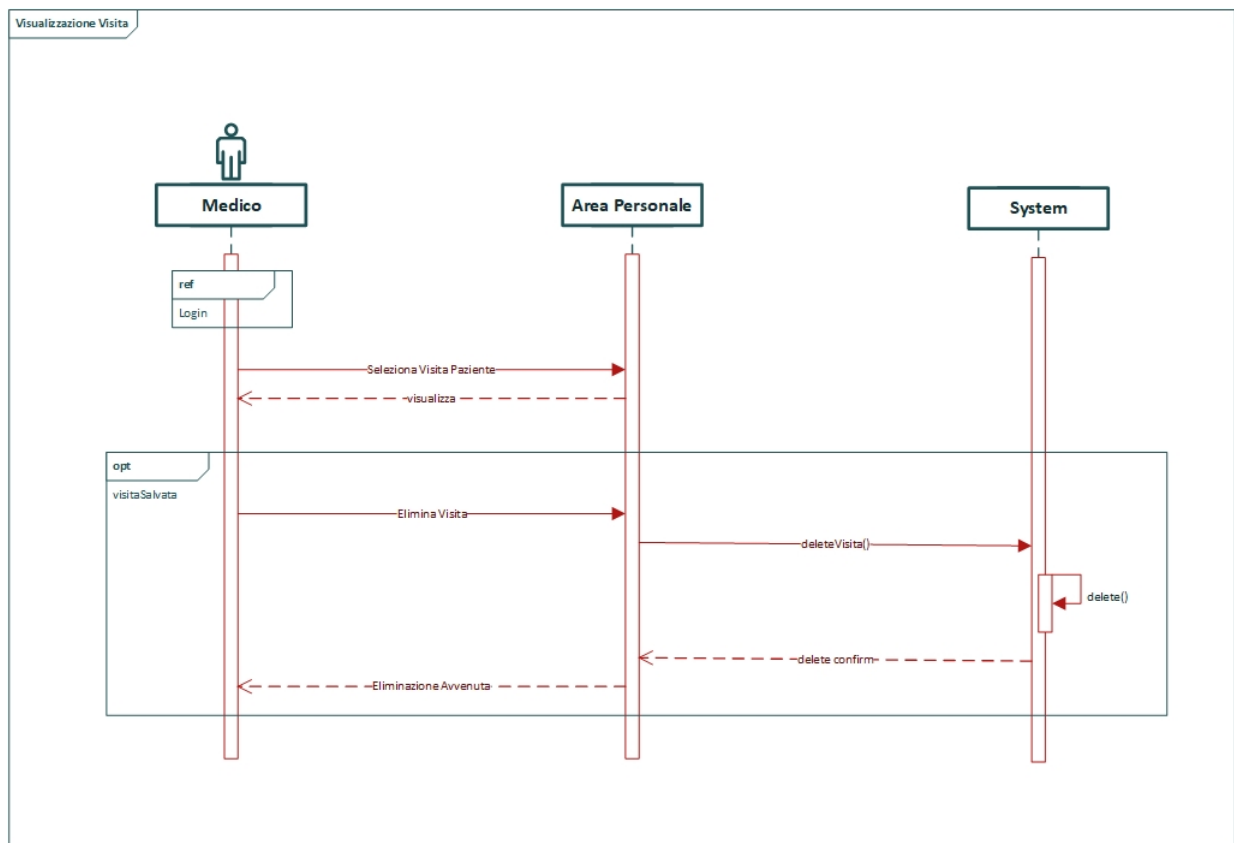


Figura 3.3: Diagramma Di Sequenza del caso d'uso: Visualizza Visita

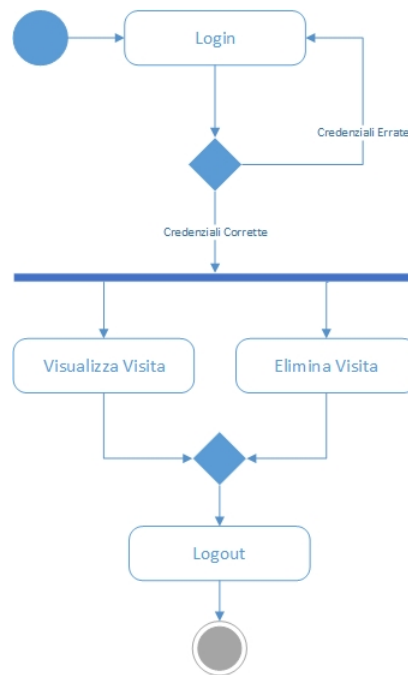


Figura 3.4: Diagramma delle Attività del caso d'uso: Visualizza Visita

3.4.3 Compilazione Form Inserimento Medico

Nome: Compilazione Form Inserimento Medico

Portata: Cartella Cardiologica Virtuale

Livello: Obiettivo Utente

Attore Primario: Amministratore

Parti interessate e interessi: L'amministratore vuole compilare il form per l'inserimento dei dati di un nuovo medico

Pre-condizioni: L'amministratore deve essere registrato nel sistema e deve aver effettuato il login

Garanzie di Successo: L'amministratore compila il form con i dati del nuovo medico

Scenario principale di successo:

- L'amministratore effettua il login
- compila il form con i dati del paziente
- il medico viene inserito nel sistema

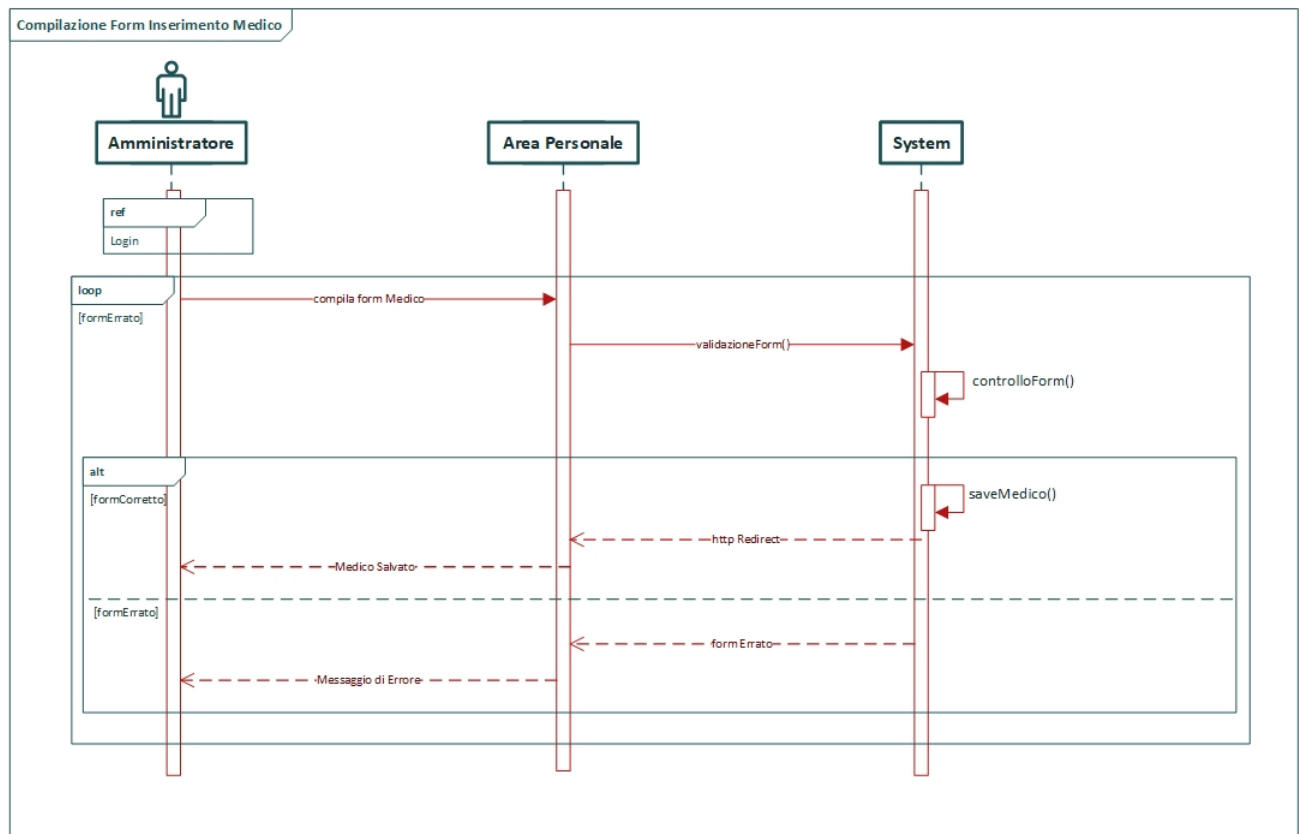


Figura 3.5: Diagramma Di Sequenza del caso d'uso: Compilazione Form Inserimento Medico

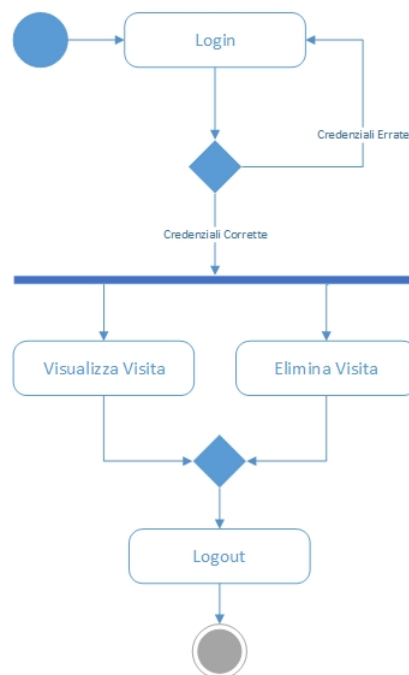


Figura 3.6: Diagramma delle Attività del caso d'uso: Compilazione Form Inserimento Medico

3.4.4 Compilazione Form Visita

Nome: Compilazione Form Visita

Portata: Cartella Cardiologica Virtuale

Livello: Obiettivo Utente

Attore Primario: Medico

Parti interessate e interessi: Il medico vuole compilare il form della visita per il paziente

Pre-condizioni: Il medico deve essere registrato nel sistema e deve aver effettuato il login

Garanzie di Successo: Il medico compila il form con i dati della visita

Scenario principale di successo:

- Il medico effettua il login
- Compila il form della visita con i valori del paziente
- La visita viene inserita nel sistema

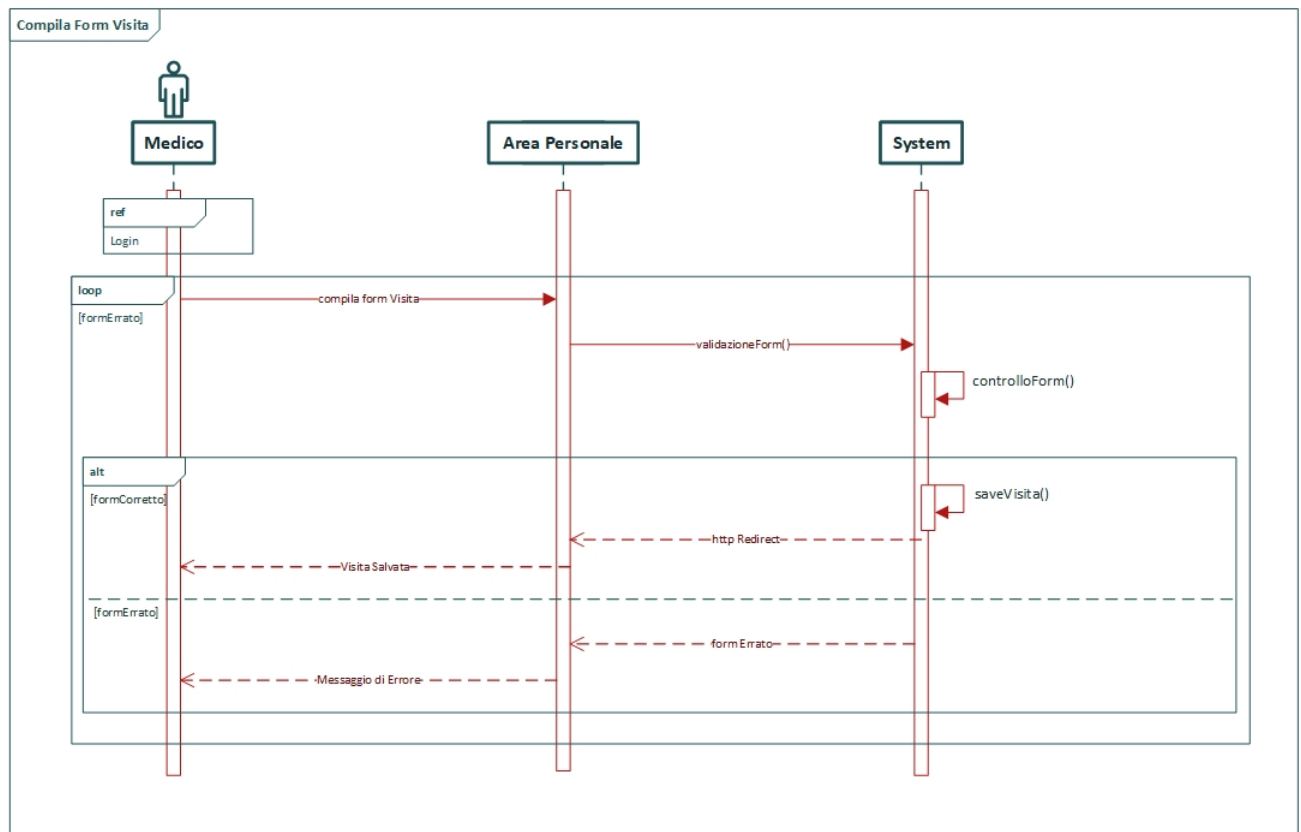


Figura 3.7: Diagramma Di Sequenza del caso d'uso: Compilazione Form Visita

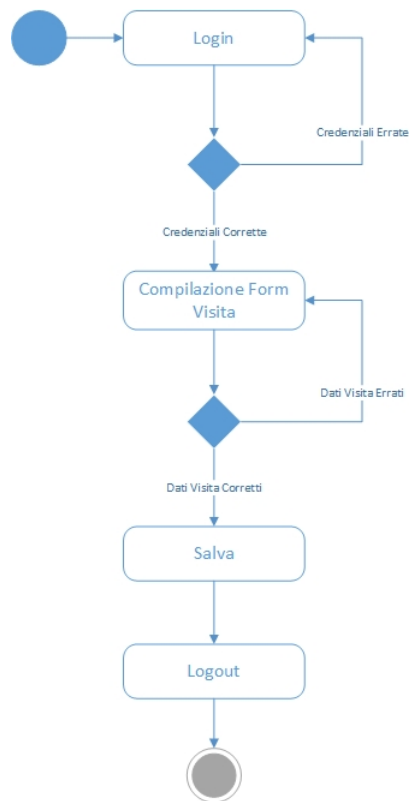


Figura 3.8: Diagramma delle Attività del caso d’uso: Compilazione Form Visita

3.4.5 Inserimento Referto

Nome: Inserimento Referto

Portata: Cartella Cardiologica Virtuale

Livello: Obiettivo Utente

Attore Primario: Paziente

Parti interessate e interessi: Il paziente vuole inserire un referto nella sua cartella clinica

Pre-condizioni: Il paziente deve essere registrato nel sistema e deve aver effettuato il login

Garanzie di Successo: Il paziente inserisce il referto, ovvero un file, nell’apposito form

Scenario principale di successo:

- Il paziente effettua il login
- accede alla sua area personale
- carica il file (referto) nel form
- il referto viene memorizzato nella cartella del paziente

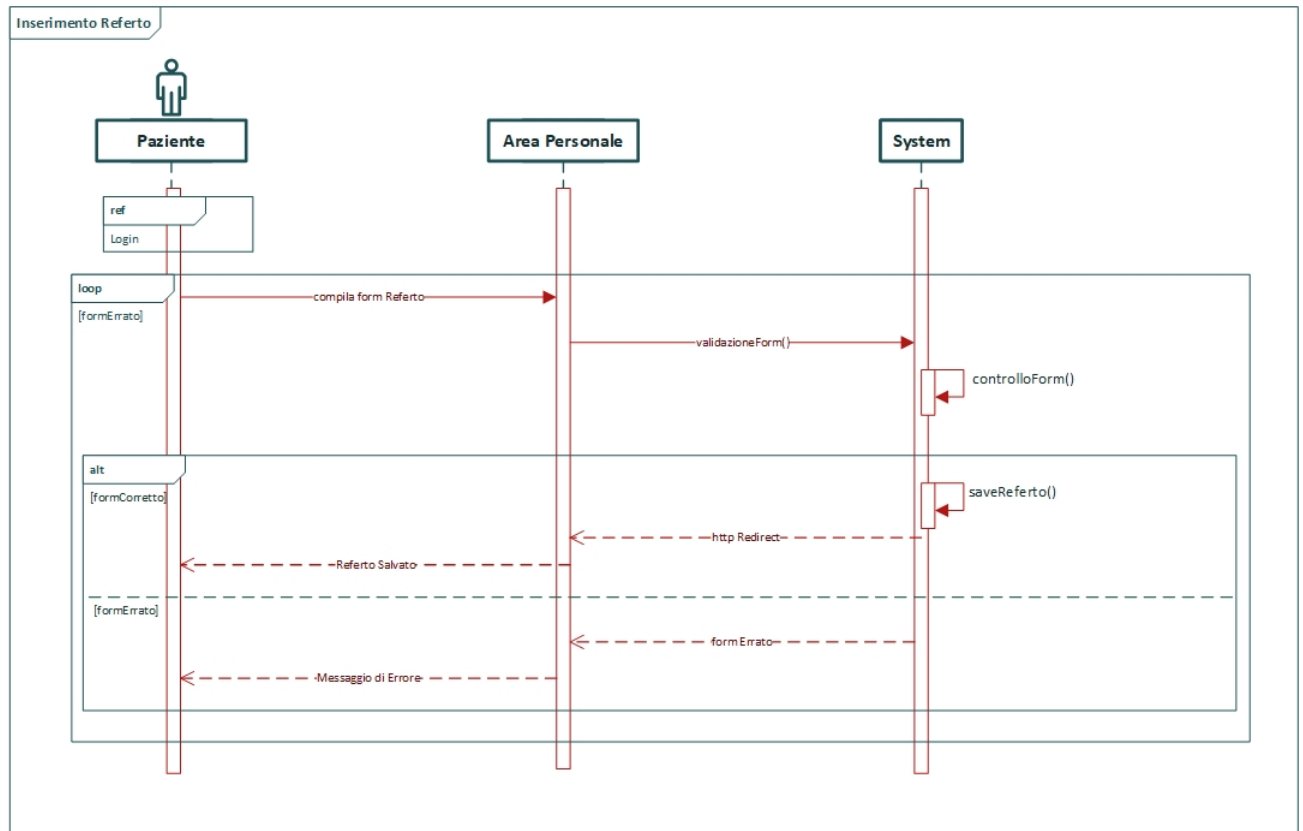


Figura 3.9: Diagramma Di Sequenza del caso d'uso: Inserimento Referto

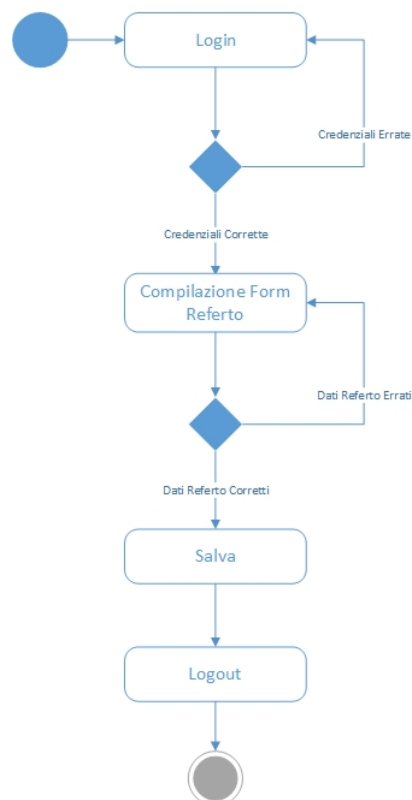


Figura 3.10: Diagramma delle Attività del caso d'uso: Inserimento Referto

Capitolo 4

Progettazione

In questo capitolo vengono presentate tutte le scelte di progettazione adottate per l'implementazione della Cartella Cardiologica Virtuale.

Come anticipato lo scopo di questa applicazione è quello di permettere ad un utente di accedere in qualsiasi momento alla propria cartella clinica per visualizzare le informazioni desiderate.

Il software realizzato è una web-app, ovvero un'applicazione progettata per essere accessibile mediante il web da qualsiasi utente.

L'architettura di questo software è stata implementata utilizzando il pattern Model View Controller che, come esplicitato nel Capitolo 1, è il design pattern maggiormente indicato per applicazioni di questa tipologia.

In seguito vengono presentate le implementazioni del lato Client e del lato Server di tale applicazione.

4.1 Client

Il lato Client è stato implementato mediante l'utilizzo del framework Angular JS e dei CSS per definirne l'aspetto grafico.

Il front-end segue il paradigma MVC e ciò è reso possibile grazie all'utilizzo di pagine jsp (JavaServer Pages) ovvero una tecnologia di programmazione web in Java che consente di sviluppare la logica di presentazione secondo il pattern MVC.

Le pagine jsp permettono infatti di realizzare l'aspetto grafico, ovvero tutto ciò che viene visualizzato dal client e con il quale egli può interagire.

Ogni pagina jsp viene interfacciata da una classe controller, più precisamente una Servlet, che si occupa di elaborare le richieste del client.

4.1.1 Interfaccia

L'interfaccia è stata progettata ponendo come obiettivo principale la semplicità di interazione.

L'applicazione presentata in questo elaborato è orientata all'ambito medico, nel quale l'utilizzatore ha la necessità di acquisire le informazioni attraverso un sistema semplice ed immediato.

Per tale ragione si è scelto di realizzare un'interfaccia sufficientemente semplice e intuitiva.

La Figura 4.1 mostra l'Homepage della web-app descritta in questo elaborato



Figura 4.1: Homepage

L'aspetto grafico delle diverse pagine jsp è stato realizzato mediante l'utilizzo dei CSS, e si è scelto di adottare colori tenui che facciano risaltare le importanti informazioni del client.

Interfaccia del Medico



Figura 4.2: Area privata del Medico - Lista dei Pazienti

La Figura 4.2 mostra l'area personale del medico dalla quale egli può visualizzare la lista dei pazienti registrati nell'applicazione e, se ne ha la necessità, può inserire un nuovo utente.

Area Riservata del Medico

[Logout](#)

[Homepage](#) [Visualizza i pazienti](#) [Inserisci un nuovo paziente](#)

Cartella Cardiologica del Paziente - Lorenzo De Rosa

[Cartella di Lorenzo De Rosa](#) [Visite di Lorenzo De Rosa](#) [Rischio Cardiologico](#) [Referti di Lorenzo De Rosa](#)

 [Stampa le visite del paziente](#)

Medico	Paziente	Data	Anamnesi	Motivo Visita	Pressione	Frequenza Cardiaca	Peso	Altezza	Circonferenza Vita	
Angelo Rossi	Lorenzo De Rosa	22/11/2020	medica	controllo	60.0	60.0	70.0	170.0	50.0	Elimina Modifica
Carlo Bianchi	Lorenzo De Rosa	25/12/2020	medica	controllo	60.0	60.0	70.0	170.0	60.0	Elimina Modifica

Figura 4.3: Cartella Clinica Del Paziente

Nella propria area personale il medico può visualizzare la lista dei pazienti registrati nell'applicazione (come mostrato in Figura 4.2) e per ognuno può scegliere di eliminarlo, effettuare una visita oppure accedere alla sua cartella clinica.

La Figura 4.3 mostra la cartella clinica di un paziente vista dall'area personale del medico. Da qui il medico può visualizzare tutte le visite del paziente, effettuare l'upload o il download di eventuali referti, oppure calcolarne il rischio cardiologico.

Interfaccia del Paziente

Cartella Cardiologica

[Logout](#)

[Homepage](#) [Vedi le tue visite](#) [Visualizza il tuo rischio Cardiologico](#) [Calcola il Rischio Cardiologico](#) [I tuoi Referti](#)

Ciao **Sandro Fubelli**,

Questa è la tua cartella cardiologica e da qui puoi:

- Visualizzare tutte le visite mediche che hai ricevuto
- Visualizzare il tuo rischio cardiologico
- Calcolare il tuo rischio cardiologico
- Visualizzare e inserire i tuoi referti medici

 [I Tuoi Dati](#)

Nome: Sandro
Cognome: Fubelli
Codice Fiscale: FBLSNDR67TGFHDJK
Data di Nascita: 05/03/1967
Indirizzo: via Platani 8, MI
Telefono: 47635473
Email: sandro@mail.it

 [Modifica Dati](#)

Figura 4.4: Area privata del Paziente

La Figura 4.4 mostra l'area personale del paziente, dalla quale egli può visionare le visite che gli sono state effettuate e può effettuare l'upload e il download di eventuali

referti medici.

Cartella Cardiologica

Logout



Figura 4.5: Rischio Cardiologico del Paziente

Inoltre, come mostrato in Figura 4.5, il paziente può visionare il proprio rischio cardiologico.

L'applicazione infatti restituisce un messaggio che riporta tale rischio associando ad esso un colore significativo come definito nell'indice di Figura 1.5.

4.2 Server

Il lato Server è stato implementato nel linguaggio di programmazione Java adottando i framework Spring e Hibernate.

Nella fase di progettazione si è scelto di adottare il pattern MVC in quanto più indicato per la realizzazione di applicazioni web.

Al fine di implementare tale architettura è stato utilizzato il framework Spring, o più precisamente, il modulo Spring MVC.

Spring è una piattaforma basata su una struttura modulare, rendendolo leggero e facilmente integrabile con altri framework, quali Hibernate.

Nella web-app in questione infatti, la persistenza dei dati è stata gestita mediante il framework Hibernate che ha la proprietà di presentare i dati relazionali come oggetti di Java e di mapparli all'interno del database mediante lo strumento ORM (Object Relational Mapping).

Questa sua caratteristica rende Hibernate un framework molto utile garantendo una gestione semplice e veloce del database.

I framework Spring e Hibernate vengono illustrati in modo accurato nel capitolo sulle Tecnologie Utilizzate.

In seguito vengono presentati: il Diagramma delle classi e la Base Di Dati.

4.2.1 Diagramma delle Classi

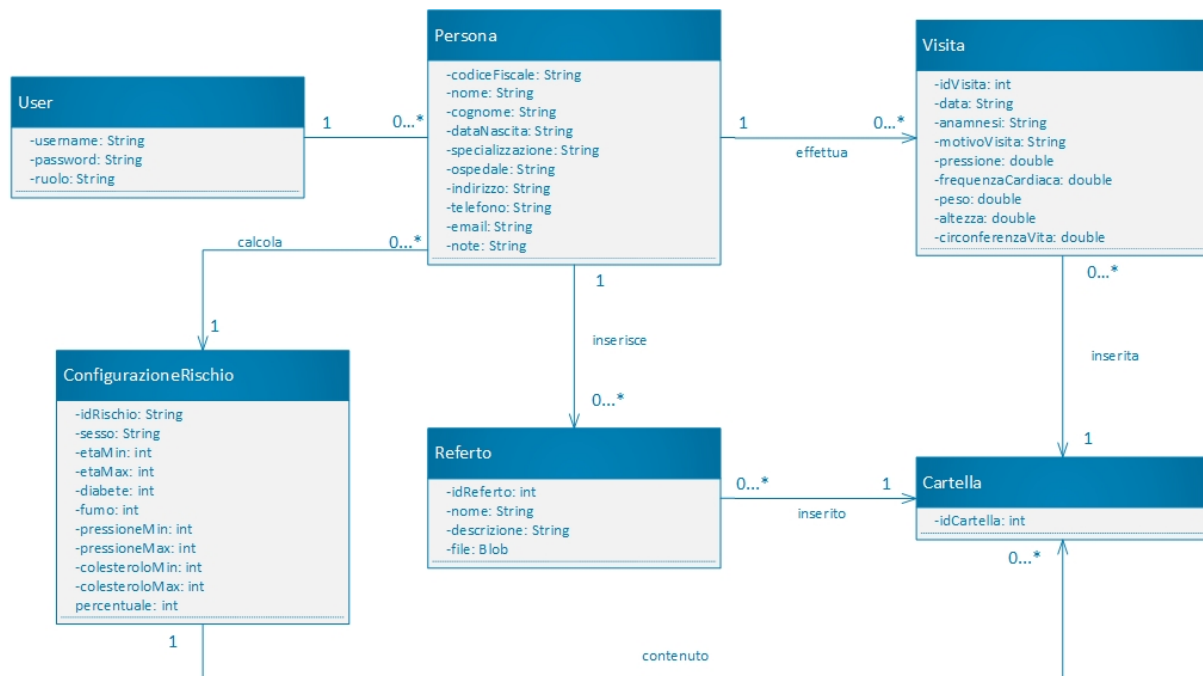


Figura 4.6: Diagramma Delle Classi

4.3 Base Di Dati

Nella fase di progettazione della base di dati sono state inizialmente individuate le entità principali e le relazioni esistenti tra loro.

Questi legami vengono rappresentati in modo schematico mediante il diagramma E-R (Entity Relationship) presentato in Figura 4.7:

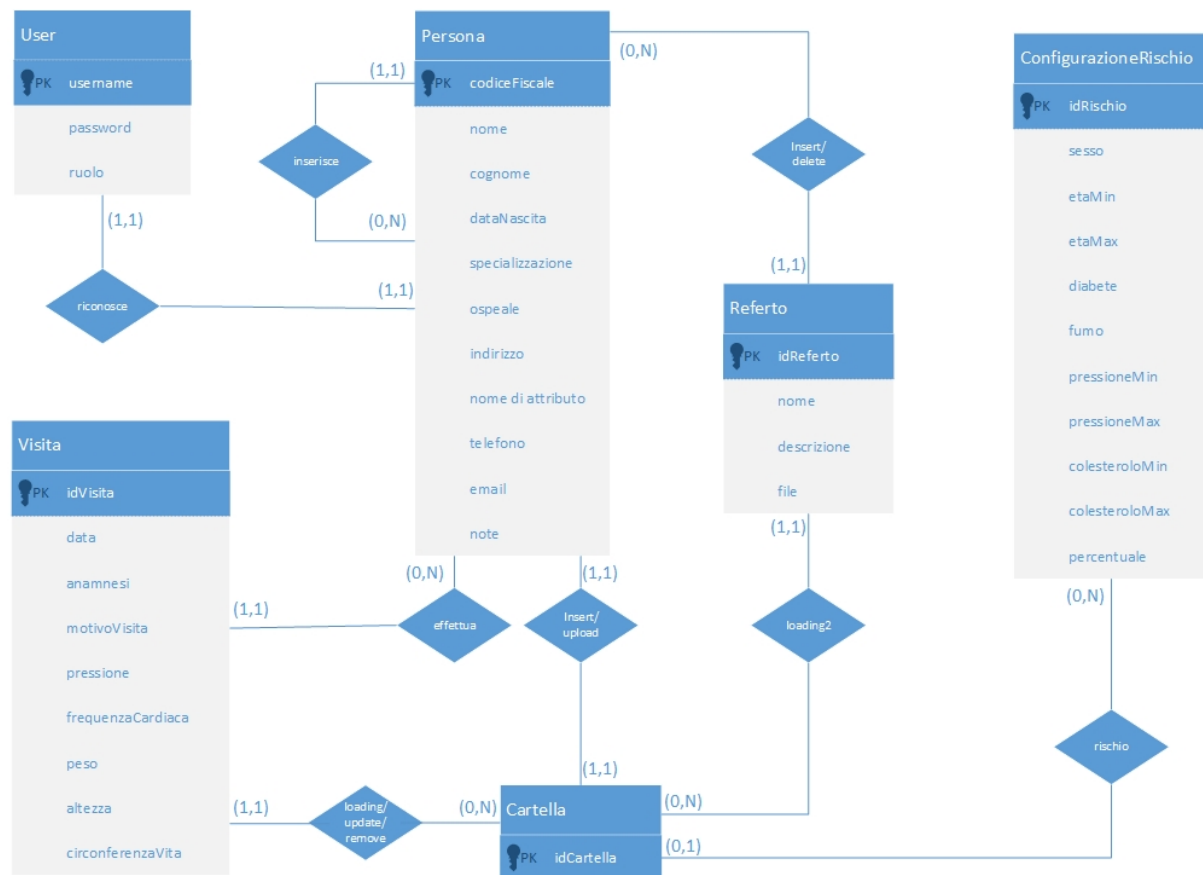


Figura 4.7: Diagramma E-R

In tale modello le entità vengono rappresentate mediante dei rettangoli e ne vengono definiti gli attributi. Gli attributi che rappresentano la chiave primaria vengono evidenziati mediante il simbolo "PK" preceduto da una chiave. Le relazioni esistenti tra le diverse entità vengono rappresentate mediante i rombi.

Successivamente il modello E-R è stato tradotto nel modello relazionale presentato in Figura 4.8.

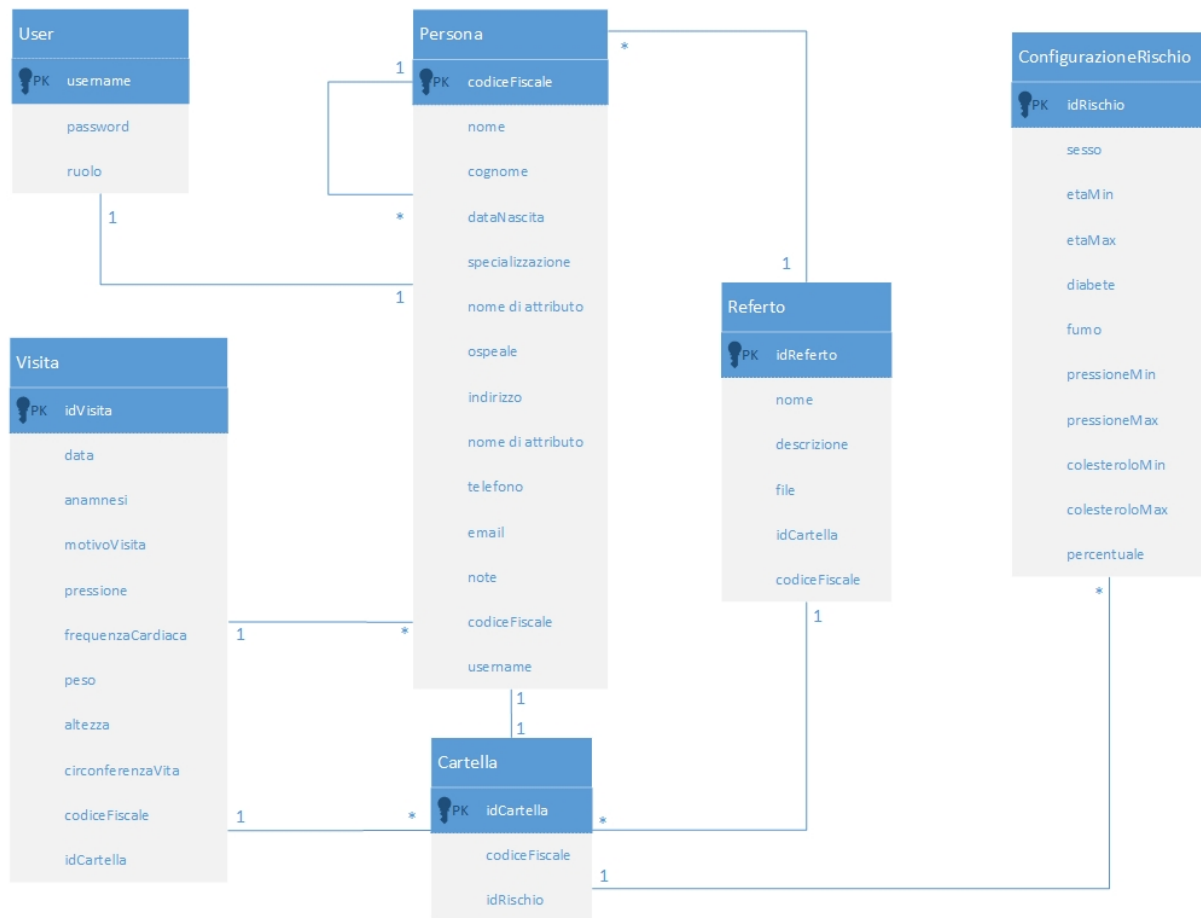


Figura 4.8: Modello Relazionale

Tale modello è una rappresentazione logica della struttura del database. Tutti i dati sono rappresentati come un insieme di relazioni o tabelle. Le colonne di una tabella corrispondono agli attributi dell'entità che essa rappresenta, mentre le righe corrispondono alle tuple (o record), ovvero le informazioni contenute.

Per chiarire tale concetto si introduce il seguente esempio:

Si consideri la tabella User così definita:

Username	Password	Ruolo
lucaRossi	prova	paziente
user	psw123	medico

Le colonne di tale tabella rappresentano gli *attributi*, ovvero Username, Password e Ruolo.

La seconda e la terza riga identificano invece due *record*: il primo ha come username: "lucaRossi", come password: "prova" e come ruolo: "paziente". Mentre il secondo record ha come username: "user", come password: "psw123 " e come ruolo: "medico".

L'implementazione della base di dati è stata effettuata mediante il linguaggio SQL (Structured Query Language), un linguaggio per la gestione di database basati sul modello relazionale.

Nello specifico, per l'implementazione della base di dati, è stato utilizzato uno script nel linguaggio SQL, mediante il quale sono state definite le tabelle del database con i relativi attributi e vincoli su di essi.

Capitolo 5

Conclusioni e Sviluppi Futuri

L'esperienza di tirocinio svolto presso l'azienda Sync Lab Srl è stata molto significativa ed ha potenziato le mie competenze in ambito informatico.

Il progetto descritto in questo elaborato mi ha fornito l'opportunità di lavorare, per la prima volta, allo sviluppo Full-Stack di un'applicazione, ovvero alla realizzazione del back-end e del front-end della web-app.

Un ulteriore aspetto positivo di questa esperienza è stata la possibilità di apprendere e implementare nello sviluppo dell'applicazione i due framework Spring e Hibernate.

L'applicazione sviluppata può rappresentare una soluzione efficiente per la virtualizzazione di una cartella clinica.

Grazie a questa applicazione infatti, sia i pazienti di un ospedale che i medici, potrebbero accedere in qualsiasi momento all'applicazione acquisendo le informazioni desiderate.

La web-app sviluppata, inoltre, potrebbe fornire una duplice soluzione ai problemi di inquinamento e di archiviazione dei dati.

Al giorno d'oggi una delle tematiche maggiormente discusse è quella dell'impatto ambientale.

Il consumo di carta, infatti, ha un impatto negativo sull'ambiente in diverse modalità, tra cui la produzione di significanti quantità di rifiuti, l'uso di risorse naturali come alberi, acqua e, combustibili fossili con il conseguente rilascio di sostanze inquinanti nell'atmosfera.

La virtualizzazione dei dati ospedalieri, quali referti medici, documenti e simili, potrebbe rivelarsi significativa per la riduzione di tali problematiche.

Inoltre l'archiviazione fisica dei documenti, come quelli ospedalieri richiede grandi quantità di spazio e tempi di consultazione rilevanti.

La virtualizzazione di tali documenti permetterebbe quindi di ridurre la necessità di avere spazi di dimensioni rilevanti con il solo scopo di conservare documenti, e garantirebbe un accesso ai dati in tempi brevi.

L'applicazione realizzata rappresenta un'ottima base per un eventuale sviluppo futuro della stessa.

Una possibile modifica potrebbe riguardare il processo di registrazione nel sistema, con l'aggiunta di una funzionalità che permetterebbe all'utente di ricevere un'e-mail di conferma della registrazione, incrementando il livello di sicurezza.

Un'ulteriore funzionalità potrebbe essere quella di favorire la comunicazione diretta tra pazienti e medici inserendo all'interno dell'area personale degli utenti un apposito form o una casella di testo.

Ciò garantirebbe un semplice ed efficace scambio di messaggi in caso di necessità.

Bibliografia

- [1] Gazi Imtiyaz Ahmad, Jimmy Singla, and Kaiser J Giri. Security and privacy of e-health data. In *Multimedia Security*, pages 199–214. Springer, 2021.
- [2] Marc L Berger, Melissa D Curtis, Gregory Smith, James Harnett, and Amy P Abernethy. Opportunities and challenges in leveraging electronic health record data in oncology. *Future Oncology*, 12(10):1261–1274, 2016.
- [3] Andreas Chang. Utaut and utaut 2: A review and agenda for future research. *The Winners*, 13(2):10–114, 2012.
- [4] Joan Devin, Joyce Costello, Naomi McCallion, Eavan Higgins, Brian Kehoe, Brian J Cleary, and Shane Cullinan. Impact of an electronic health record on task time distribution in a neonatal intensive care unit. *International Journal of Medical Informatics*, 145:104307, 2021.
- [5] Odai Enaizan, AA Zaidan, NH M Alwi, BB Zaidan, MA Alsalem, OS Albahri, and AS Albahri. Electronic medical record systems: Decision support examination framework for individual, security and privacy concerns using multi-perspective analysis. *Health and Technology*, 10(3):795–822, 2020.
- [6] Eleni Entzeridou, Evgenia Markopoulou, and Vasiliki Mollaki. Public and physician’s expectations and ethical concerns about electronic health record: benefits outweigh risks except for information security. *International journal of medical informatics*, 110:98–107, 2018.
- [7] Dana B Gal, Brian Han, Chistopher Longhurst, David Scheinker, and Andrew Y Shin. Quantifying electronic health record data: A potential risk for cognitive overload. *Hospital Pediatrics*, 11(2):175–178, 2021.
- [8] Cara A Harshberger, Abigail J Harper, George W Carro, Wayne E Spath, Wendy C Hui, Jessica M Lawton, and Bruce E Brockstein. Outcomes of computerized physician order entry in an electronic health record after implementation in an outpatient oncology setting. *Journal of oncology practice*, 7(4):233–237, 2011.
- [9] Morten Hertzum. Electronic health records in danish home care and nursing homes: Inadequate documentation of care, medication, and consent. *Applied Clinical Informatics*, 12(01):027–033, 2021.
- [10] Talha Iqbal and Hazrat Ali. Generative adversarial network for medical images (mi-gan). *Journal of medical systems*, 42(11):1–11, 2018.

- [11] Richelle J Koopman, Linsey M Barker Steege, Joi L Moore, Martina A Clarke, Shannon M Canfield, Min S Kim, and Jeffery L Belden. Physician information needs and electronic health records (ehrs): time to reengineer the clinic note. *The Journal of the American Board of Family Medicine*, 28(3):316–323, 2015.
- [12] Randhir Kumar and Rakesh Tripathi. Towards design and implementation of security and privacy framework for internet of medical things (iomt) by leveraging blockchain and ipfs technology. *The Journal of Supercomputing*, pages 1–40.
- [13] Elizabeth Moore, Roger Newson, Miland Joshi, Thomas Palmer, Kieran J Rothnie, Sally Singh, Azeem Majeed, Michael Soljak, and Jennifer K Quint. Effects of pulmonary rehabilitation on exacerbation number and severity in people with copd: an historical cohort study using electronic health records. *Chest*, 152(6):1188–1202, 2017.
- [14] Stephen D Persell, Alexis P Dunne, Donald M Lloyd-Jones, and David W Baker. Electronic health record-based cardiac risk assessment and identification of unmet preventive needs. *Medical care*, pages 418–424, 2009.
- [15] John Powell and Iain Buchan. Electronic health records should support clinical research. *Journal of medical Internet research*, 7(1):e4, 2005.
- [16] Madison A Price, Beatriz E Alvarado, Nicole TA Rosendaal, Saionara MA Câmara, Catherine M Pirkle, and Maria P Velez. Early and surgical menopause associated with higher framingham risk scores for cardiovascular disease in the canadian longitudinal study on aging. *Menopause*, 2021.
- [17] Subramanian Venkatesan, Shubham Sahai, Sandeep Kumar Shukla, and Jaya Singh. Secure and decentralized management of health records. In *Applications of Blockchain in Healthcare*, pages 115–139. Springer, 2021.
- [18] Quan Qiu Wang, David C Kaelber, Rong Xu, and Nora D Volkow. Covid-19 risk and outcomes in patients with substance use disorders: analyses from electronic health records in the united states. *Molecular psychiatry*, 26(1):30–39, 2021.
- [19] Julian Wolfson, David M Vock, Sunayan Bandyopadhyay, Thomas Kottke, Gabriela Vazquez-Benitez, Paul Johnson, Gediminas Adomavicius, and Patrick J O’Connor. Use and customization of risk scores for predicting cardiovascular events using electronic health record data. *Journal of the American Heart Association*, 6(4):e003670, 2017.
- [20] Mustafa Yurdakul and Yusuf Tansel Ic. Application of correlation test to criteria selection for multi criteria decision making (mcdm) models. *The International Journal of Advanced Manufacturing Technology*, 40(3-4):403–412, 2009.
- [21] Ziqi Zhang, Chao Yan, Diego A Mesa, Jimeng Sun, and Bradley A Malin. Ensuring electronic medical record simulation through better training, modeling, and evaluation. *Journal of the American Medical Informatics Association*, 27(1):99–108, 2020.