

POLITECNICO DI TORINO

Corso di Laurea in Ingegneria del cinema e dei mezzi di comunicazione

Tesi di Laurea Magistrale

GUCH
Progettazione e sviluppo di
un'applicazione Web in ambito
clinico



Relatore
prof. Laura Farinetti

Laureando
Giorgia GAMERRO
matricola: s233087

CGM Consulting S.r.l
Azienda

ANNO ACCADEMICO 2017 – 2018

Sommario

L'obiettivo di questo testo è illustrare il processo di progettazione e sviluppo di un'applicativo Web gestionale. Il progetto, denominato GUCH (*Grown up congenital heart - Cardiopatici congeniti adulti*) ha lo scopo di realizzare un registro online, per far sì che medici specializzati in cardiologia possano tenere traccia e documentare l'evoluzione e il decorso delle patologie cardiache nei propri pazienti. Il fine ultimo è quello di acquisire informazioni statistiche, utili per migliorare le cure e progredire nei risultati ottenibile con una costante ricerca scientifica in questo ambito.

L'applicativo è stato sviluppato all'interno della azienda CGM Consulting di Torino, in un'esperienza di tesi in azienda iniziata ad ottobre. Il progetto, estremamente ampio e di grandi dimensioni, ha coinvolto tre fasi fondamentali: la progettazione dell'interfaccia, lo sviluppo e la fase di testing dell'applicativo; esse vengono analizzate ed approfondite all'interno della tesi andando a contrapporre alle nozioni tecniche e teoriche, le scelte implementative.

L'elaborato, nel primo capitolo, mette in luce gli scopi del progetto, descrivendo il cliente committente e gli obiettivi di utilizzo dell'applicativo, mentre nel successivo vengono analizzate le principali linee guida per la progettazione di un'efficace interfaccia grafica che tenga conto dell'utente e delle sue necessità, al fine di consegnare un prodotto completo, efficiente e funzionale. L'analisi del progetto prosegue con una piccola panoramica sui linguaggi utilizzati (HTML, CSS-Less, Javascript, AngularJS) per lo sviluppo del codice e nel quarto capitolo viene studiata nel dettaglio la navigazione dell'applicativo. In questa sezione vengono approfondite le entità protagoniste, le funzionalità specifiche e la loro organizzazione. Con l'aiuto di schemi e mappe di navigazione vengono messe in evidenza le difficoltà incontrate in fase di progettazione e la complessità generale dell'applicativo, mentre l'ultima sezione si concentra sulla fase di testing. Partendo dall'osservazione degli utenti nell'utilizzo dell'applicativo si è cercato di mettere in evidenza elementi non conformi alle loro necessità o mal funzionamenti non riscontrati durante lo sviluppo, allo scopo di migliorare il risultato finale. È bene sottolineare come la conclusione della scrittura della tesi non coincida con il termine di sviluppo del progetto; nei prossimi mesi il personale medico utilizzerà il prodotto, ogni loro osservazione ed eventuali problemi riscontrati saranno utilizzati come base per perfezionare ancora l'applicativo e far sì che sia per loro uno strumento valido e di supporto al loro ruolo professionale.

L'esperienza in azienda mi ha consentito di osservare da una posizione privilegiata e lavorare su di un progetto complesso e articolato con molti aspetti di interesse per questo corso di laurea e per la costruzione di una futura figura professionale solida e competente.

Indice

Elenco delle figure	5
1 Introduzione	9
1.1 Il progetto	10
1.1.1 Il cliente	10
1.1.2 Il progetto ed i suoi scopi	12
1.1.3 Perché un'applicazione Web?	13
1.2 Riassunto dei prossimi capitoli	14
2 La User Experience	15
2.1 La definizione	15
2.2 I principi dell'usabilità	16
2.2.1 Prima legge dell'usabilità	17
2.2.2 Le tre verità assolute sul mondo del Web	19
2.2.3 Le convenzioni	22
2.2.4 Gerarchie visive e definizione di aree	23
2.2.5 I bottoni e i link	26
2.2.6 Ridurre ciò che crea disturbo	27
2.2.7 Un click di distanza	30
2.2.8 L'incompatibilità fra le piattaforme	30
2.2.9 Un layout responsive	31
2.2.10 URL complesse	31
2.2.11 Il colore e i font	33
2.2.12 L'usabilità dei form	35
2.2.13 Feedback per l'utente	39
2.3 L'usabilità, valutare i problemi	42
3 Le tecnologie utilizzate	45
3.0.1 HTML	45
3.0.2 CSS e LESS	46
3.0.3 Javascript	47
3.0.4 AngularJS	47

4	La navigazione	49
4.1	Introduzione	49
4.2	Analisi del precedente progetto	50
4.2.1	Il login	51
4.2.2	La vista elenco pazienti	52
4.2.3	Inserimento di una nuova utenza	53
4.2.4	La registrazione di nuove voci	54
4.2.5	Gestione del follow-up	55
4.2.6	Filtri e ricerca	56
4.2.7	Le statistiche	57
4.2.8	Il manuale	61
4.3	La navigazione in GUCH	62
4.3.1	Le entità	63
4.3.2	Il login	64
4.3.3	Mappa complessiva dell'applicativo	64
4.3.4	L'entità nazionale	67
4.3.5	L'entità regionale	70
4.3.6	L'entità medico	73
4.3.7	L'entità paziente	79
5	I test	83
5.1	Introduzione	83
5.1.1	I test di usabilità	83
5.1.2	I test di usabilità e i risultati	85
5.1.3	Scegliere che cosa testare	85
5.1.4	Il debriefing	86
5.2	I test di GUCH	87
5.2.1	Prima fase di test	87
5.2.2	Seconda fase di test	92
6	Conclusioni e sviluppi futuri	97
	Bibliografia	99

Elenco delle figure

1.1	Logo dell'Associazione Amici del Cuore	11
2.1	Esempio di utilizzo dei tab all'interno dell'applicativo per la suddivisione delle aree di interesse - Scheda paziente	18
2.2	Esempio di utilizzo delle tabelle per la presentazione di elenchi - Elenco dei pazienti.	18
2.3	Posizionamento del campo ricerca all'interno della pagina.	19
2.4	Esempio di utilizzo della pagination per l'organizzazione degli elenchi.	20
2.5	Esempio di icone utilizzate.	23
2.6	Esempio menu laterale per l'entità nazionale.	24
2.7	Esempio menu laterale per l'entità struttura.	25
2.8	Esempio menu laterale per l'entità paziente relative alla scheda paziente.	25
2.9	Esempio utilizzo dei tab per la presentazione delle informazioni.	26
2.10	Esempio di riferimento dell'artificio estetico utilizzato.	28
2.11	Esempio di riferimento dell'artificio estetico utilizzato.	29
2.12	Font scelti.	33
2.13	Esempio temi colore.	34
2.14	Form di esempio -Registrazione nuovo paziente	35
2.15	Form di esempio - Messaggi di errori.	36
2.16	Form di esempio - Percepibilità dei form.	38
2.17	Feedback per l'utente.	40
2.18	Problemi di usabilità - Diagramma a torta.	43
4.1	Login dell'applicazione	51
4.2	Home - Prima pagina visualizzata a seguito del login	52
4.3	Vista elenco pazienti	53
4.4	Vista elenco pazienti	53
4.5	Registrazione nuovo paziente	54
4.6	Menu della scheda paziente	54
4.7	Follow-up del paziente	55
4.8	Ordinamento voci in tabella	56
4.9	Ordinamento voci in tabella	56
4.10	Campo ricerca	57
4.11	Estrazione delle statistiche	58
4.12	Statistiche con grafico a barre - Aritmie	60
4.13	Statistiche con grafico a torta - Aritmie	61
4.14	Sezione istruzioni	61

4.15	Le entità	63
4.16	Mappa del login	64
4.17	Mappa complessiva dell'applicazione	66
4.18	Mappa riferita all'entità nazionale	68
4.19	Diagramma entità nazionale	69
4.20	Mappa riferita all'entità regionale	71
4.21	Diagramma entità regionale	72
4.22	Mappa riferita all'entità medico	76
4.23	Seconda mappa riferita all'entità medico	77
4.24	Diagramma entità medico	78
4.25	Sezione diagnosi e agenda - Dispositivo smartphone	80
4.26	Sezione timeline - Dispositivo desktop	80
4.27	Mappa riferita all'entità paziente	81
4.28	Diagramma entità Paziente	82
5.1	Mock-up: vista entità medico. Registrazione diagnosi	90
5.2	Mock-up: vista entità medico. Registrazione di un nuovo follow-up	90
5.3	Mock-up: vista entità medico. Vista elenco pazienti	91
5.4	Mock-up: vista entità nazionale. Gestione database	91
5.5	Mock-up: vista entità paziente.	92

Listings

2.1	Pagination - HTML - estratto di codice	20
2.2	Ordinamento - HTML- estratto di codice	20
2.3	Tab - HTML - estratto di codice	26
2.4	Artificio estetico - HTML - estratto di codice	29
2.5	Artificio estetico - LESS - estratto di codice	29
2.6	URLEncoded - Javascript - estratto di codice	32
2.7	formData - Javascript - estratto di codice	32
2.8	Codice colori e font - CSS	34
2.9	Messaggi d'errore - HTML	36
2.10	Form di esempio - HTML - estratto di codice	38
2.11	Messaggi di conferma - Javascript - Estratto di codice	40
3.1	Esempio utilizzo mixin - Less - Estratto di codice documentazione ufficiale	46
4.1	Campo ricerca - HTML e Javascript - estratto di codice	57
4.2	Grafico aritmie - HTML - estratto di codice	58
4.3	Grafico aritmie - Javascript - estratto di codice	59
4.4	Istruzioni - HTML - estratto di codice	61
4.5	Istruzioni - Javascript - estratto di codice	62

*Guardate lontano, e anche quando credete di
star guardando lontano, guardate ancora più
lontano.*

[ROBERT BADEN POWELL]

Capitolo 1

Introduzione

Negli ultimi anni abbiamo assistito ad una progressiva espansione e sviluppo del ICT (*Information and Technology*) e di conseguenza, delle sue relative applicazioni. Si può affermare, ormai con assoluta certezza, che ognuno di noi quotidianamente utilizza il mondo del Web per portare a termine anche la più piccola attività, dallo svago alle attività lavorative e siamo arrivati a ritenere la tecnologia Internet come indispensabile componente, piuttosto che uno strumento alternativo e di supporto alle nostre attività.

Le innovazioni tecnologiche in questo ambito sono diverse e hanno interessato negli anni molti settori; notevole è stato per esempio il miglioramento apportato dalla rete Internet sia nel settore della gestione, che in quello della progettazione. I siti Internet si sono trasformati da semplici pagine statiche a complesse e sofisticate vetrine e sono aumentati, in modo esponenziale, il numero delle applicazioni Web che quotidianamente utilizziamo. Facendo un'analisi e un confronto con l'ambiente attuale si può osservare come il modo in cui utilizziamo Internet, il Web e le tecnologie ad esse collegate sia mutato profondamente. Fino agli anni 2000 (circa) si utilizzava il Web usufruendo di schermi di dimensioni abbastanza grandi, con un mouse ed una tastiera e per lo più il suo utilizzo era legato alla scrivania; oggi invece, siamo abituati ad utilizzare per navigare: smartphone, tablet, computer portatili che abbiamo sempre con noi e che ci consentono di essere connessi in qualunque momento; tutto ci appare a meno di un click di distanza.

Negli ultimi dieci anni abbiamo assistito ad un forte e progressivo processo di sviluppo che ha portato il Web ad una veloce trasformazione; ogni sito è più potente e completo del suo predecessore ed è diventata abitudine per ognuno di noi acquistare, vendere, condividere informazioni, documentarsi e studiare utilizzando il Web e l'impossibilità di compiere anche solo una di queste azioni, ritenute ormai semplici e normali, ci infastidisce.

Contemporaneamente al miglioramento e allo sviluppo del Web, anche l'usabilità è diventata elemento fondamentale e imprescindibile, si è assistito ad una progressiva crescita della figura professionale dello User Experience. Attualmente risulta difficile e complesso dare una definizione di User Experience e conseguentemente analizzare e costruire la sua figura professionale; questo perchè coniuga ed accoglie differenti discipline. È interessante osservare come tutto quello che utilizziamo quotidianamente porti con se un'attenta progettazione dell'esperienza d'uso dell'utente anche in settori molto diversi fra loro. È utile analizzare come, nell'era dell'Internet of Things si stia rivoluzionando tutto, dall' Industrial Internet, alla domotica, dal wellness alla digital health, tutti settori ad altissimo consumo

di competenze di User Experience. È altrettanto importante osservare come gli studi condotti e i principi sviluppati negli anni nel campo della User Experience, seppur il panorama sia notevolmente cambiato, siano comunque ancora validi ed utilizzabili; questo perchè in realtà l'usabilità si riferisce alle persone e a come esse capiscono e imparano ad utilizzare gli strumenti e non la tecnologia in se. Si può affermare che: le persone cambiano molto più lentamente rispetto a qualsiasi tecnologia; Jakob Nielsen ¹ afferma: la capacità del cervello umano non cambia da un anno all'altro e quindi le intuizioni che provengono dallo studio del comportamento umano hanno una durata molto lunga, quello che era difficile per gli utenti vent'anni fa continua a essere difficile anche oggi.

1.1 Il progetto

L'obiettivo di questa tesi è analizzare il processo di progettazione e realizzazione di una Web application per la gestione del Registro Nazionale delle Cardiopatie. Il lavoro ha interessato tre fasi fondamentali:

- progettazione e studio dell'interfaccia utente (processo che ha portato alla delineazione di un mockup);
- sviluppo del progetto (lato Front-end) attraverso l'utilizzo di strumenti quali: HTML, CSS, Javascript, Angular 1.6;
- testing dell'applicativo.

Il progetto è stato sviluppato all'interno dell'azienda CGM Consulting di Torino dall'inizio di ottobre 2017 in un'esperienza di tesi in azienda; a partire dai prossimi capitoli inizieremo ad analizzare il lavoro che è stato portando avanti, osservando nel dettaglio quali siano state le difficoltà incontrate nel suo sviluppo e come esse siano state risolte.

1.1.1 Il cliente

Per poter comprendere meglio gli scopi e le finalità del progetto è necessario dare una piccola descrizione del committente del lavoro.

Il progetto è stato commissionato dall'Associazione di Torino, Amici del Cuore ². La ON-LUS nasce a Torino nel 1977 con lo scopo di essere d'aiuto ai propri soci nell'avere un costante e quotidiano supporto nella gestione delle malattie cardiovascolari. L'Associazione si propone di fornire informazioni sulle malattie cardiache a persone in condizioni fisiche, psichiche, economiche, sociali o familiari svantaggiate mediante l'organizzazione d'iniziativa culturali e sociali. Ogni manifestazione viene organizzata e promossa con lo scopo

¹Jakob Nielsen è titolare del Nielsen Norman Group, insieme a Donald Norman. Ha fondato il movimento dell' "usabilità a prezzi usabili" con lo scopo di diffondere metodi veloci ma allo stesso tempo efficienti per migliorare la qualità delle interfacce. È considerato un'autorità di riferimento per il design interattivo, autore di Web Usability e di Web Usability 2.0, considerati, dai professionisti del settore due punti di riferimento per la progettazione di interfacce Web. [1]

²Sito dell'Associazione <http://www.amicidelcuoretorino.it/> - Per informazioni dettagliate rispetto agli scopi dell'Associazione



Figura 1.1: Logo dell'Associazione Amici del Cuore

di divulgare le informazioni necessarie al fine di evitare l'insorgere di patologie cardiache (in un'ottica di prevenzione e sensibilizzazione dell'opinione pubblica ai problemi medico-sociali delle malattie cardiovascolari), di sostenere moralmente e fornire il necessario aiuto concreto, diretto od indiretto, ad adulti e/o bambini prima e dopo un trapianto o durante le cure specialistiche del caso. Contemporaneamente si propone di collaborare e promuovere l'attività di studio, di formazione e ricerca nell'ambito dei malati di cuore e prestare assistenza negli ospedali della Regione. Sono numerose le occasioni in cui l'Associazione Amici del Cuore ha organizzato e pubblicizzato manifestazioni per sensibilizzare l'opinione pubblica sulle vitali problematiche della carenza di donatori di organi.

Per comprendere a pieno gli obiettivi e le finalità del progetto che l'Associazione Amici del Cuore ha commissionato, è necessario approfondire la realtà in cui essa opera; le sue azioni, infatti, si vanno ad inserire all'interno di un progetto di più ampio respiro denominato "Rete GUCH".

La rete GUCH

GUCH è un acronimo inglese, coniato dagli anglosassoni, che significa *"grow up congenital heart"* ovvero "cardiopatici congeniti adulti". La rete GUCH è un programma che è stato attivato a partire dal 2010 presso l'Istituto Giannina Gaslini;³ cerchiamo quindi di comprendere in modo più completo quali siano le sue finalità.

È importante sottolineare come il numero di pazienti affetti da cardiopatie congenite che raggiungono l'età adulta sia in forte crescita e questo risultato lo si deve, in primo luogo, al miglioramento delle tecniche chirurgiche che comportano una decrescita del tasso di mortalità. Contemporaneamente è necessario considerare che, occuparsi di un paziente adulto con una cardiopatia congenita semplice è profondamente diverso dal trattare paziente adulti a cui sono state diagnosticate cardiopatie congenite complesse. Infatti l'accumularsi degli interventi chirurgici a cui si sovrappongono le tematiche proprie dell'età adulta rendono

³Per approfondire le tematiche relative è possibile consultare il sito Internet dell'Istituto Giannina Gaslini - <http://www.gasliniblog.org/>

il trattamento di tali casistiche un quadro estremamente variabile e talvolta di difficile interpretazione per tutti i medici che operano nel settore. Gli inglesi, già 20 anni fa, si sono accorti di questo fenomeno e hanno organizzato strutture adeguate per il trattamento di questa tipologia di pazienti affiancandoli a figure professionali, quali cardiologi e chirurghi, dall'alto profilo professionale e con competenze e specialità condivise fra l'ambito pediatrico e quello dell'adulto. Nasce così l'organizzazione GUCH il cui scopo e finalità principale quindi è quello di evitare la promiscuità tra età pediatrica ed età adulta, potendo considerare con maggiore attenzione le nuove esigenze di tipo psicologico e medico che emergono con la crescita e il perdurare della malattia e che rendono ancora più complesso il trattamento di tali pazienti.

L'Istituto Gaslini, in collaborazione con altre realtà si è differenziato per aver introdotto la "rete GUCH" in Italia raggruppando sotto una stessa organizzazione tutte le professionalità in grado di assicurare a questi pazienti continuità assistenziale e risposte a tutte le esigenze di cura, mediche e chirurgiche. Uniti in un'unica rete, cardiologi, cardiocirurghi dell'età pediatrica, cardiologi dell'adulto, medici di medicina generale, medici del lavoro e dello sport e personale infermieristico collaborano in modo sinergico mettendo il paziente GUCH al centro dell'attenzione.

Il programma permette:

- follow-up ⁴ ambulatoriale completo;
- ricovero.

Grazie a tale rete, la cura del paziente e il suo percorso diagnostico migliorano sensibilmente e di conseguenza anche la qualità di vita del paziente stesso. Avendo definito in modo più completo ed esaustivo la realtà in cui operano la rete GUCH e, congiuntamente, l'Associazione Amici del Cuore ci è possibile comprendere ed approfondire in modo più dettagliato le finalità e gli scopi di tale progetto.

1.1.2 Il progetto ed i suoi scopi

Il Registro Nazionale delle Cardiopatie, che è stato sviluppato, è uno dei tasselli della rete GUCH ed è stato finanziato dalla medesima Associazione; il nome del progetto mette in luce una delle sue principali finalità: registrare. I principali scopi per cui è stato pensato e proposto all'intera rete GUCH risultano essere:

- registrare lo stato clinico di ogni singolo paziente a seguito di una nuova visita, in modo che sia più semplice farvi accesso e ogni informazione possa essere reperibile e aggiornabile velocemente;
- disporre per ogni paziente di una storia clinica che possa documentare l'evolvere della sua patologia e, allo stesso tempo, che permetta l'interconnessione dei dati anche con altre specialità sanitarie;

⁴Con il termine follow-up si intendono le visite di controllo periodiche atte a seguire il decorso di una malattia e verificare l'efficacia delle terapie.

- definire uno strumento unico, ma soprattutto condiviso, per tutti medici;
- estrarre informazioni statistiche.

L'ultimo punto vuole essere il fulcro centrale del progetto, questo perchè risulta essere chiaro e sotto l'occhio di tutti l'elevata rilevanza che ha la ricerca in tale ambito; non è unicamente importante per la nostra vita, ma anche per il nostro futuro. Allo stesso modo ci è chiaro come il progresso della scienza dipenda strettamente dalle risorse che vi vengono investite: se scienziati e ricercatori disponessero di maggiori finanziamenti, le ricerche progredirebbero più velocemente riducendo drasticamente i tempi di attesa. Il medesimo discorso vale per lo sviluppo tecnologico e, in generale, per ogni invenzione o scoperta che si prefigge essere una soluzione ai molti problemi dell'uomo. Si deduce quindi facilmente l'importanza vitale e l'estrema convenienza di investire in maggior misura, ma soprattutto meglio ed in modo più canalizzato nella ricerca scientifica. In tempi odierni, dove le risorse devolute alla ricerca sono state progressivamente ridotte e ridistribuite, le possibilità di disporre di dati statistici validi e concreti che possono essere estratti da strumenti che ne permettano l'incrocio e il filtraggio in modo semplice e veloce acquisiscono elevata rilevanza.

Il progetto si inserisce all'interno della realtà ospedaliera italiana e sarà quindi uno strumento disponibile e utilizzabile sia dai medici specializzati in cardiologia che richiederanno di farne parte, sia dai loro pazienti che disporranno così di un portale in cui poter facilmente accedere ai propri dati personali.

Nello sviluppo di questa tesi analizzeremo nello specifico tutte le funzionalità dell'applicazione e sarà così più semplice comprendere le interazioni che intercorrono fra le entità interessate dall'applicativo.

1.1.3 Perchè un'applicazione Web?

Con il termine *application Web* si intende un' applicazione distribuita web-based; ovvero, si intende un'applicazione utilizzabile e accessibile attraverso un network, attraverso quindi una rete Internet. Per utilizzare un'applicativo Web, ovviamente, non è necessario installare nessun tipo di programma sul proprio PC in quanto essa è resa disponibile su di un server e vi si può fare accesso in un qualunque istante attraverso un comune Web Browser. Il client (l'entità che si collega attraverso il Web Browser all'applicativo) a seguito dell'instaurazione di una connessione di tipo client-server, invia richieste di servizio e il server, dopo aver elaborato la richiesta, rende disponibili al client tutti i servizi necessari. Attraverso l'utilizzo di applicazioni Web è possibile sfruttare alcuni pratici vantaggi:

- facilità di distribuzione e di aggiornamento. Risiedendo su server ogni aggiornamento è automaticamente disponibile a tutti gli utenti;
- accesso multiplatforma, è possibile accedervi indipendentemente dall'hardware del sistema e dal sistema operativo utilizzando da ogni singolo utente;
- riduzione del costo di gestione. L'utilizzo di Internet riduce notevolmente i costi di gestione dei client;
- scalabilità: un'applicativo Web ben progettato può crescere, anche in modo non regolare, in base alle esigenze senza generare problemi.

1.2 Riassunto dei prossimi capitoli

Il processo di analisi del progetto si svilupperà in quattro capitoli. Nel prossimo analizzeremo le principali nozioni di User Experience che sono state approfondite per poter sviluppare in modo organico e completo l'interfaccia dell'applicativo, nel terzo, verranno spiegati i fondamentali strumenti di sviluppo front-end che sono stati adoperati quali HTML, CSS - Less, AngularJS e Javascript e nel quarto analizzeremo invece la struttura di navigazione dell'intera applicazione.

Prima di concentrarci però sulla struttura della nuova applicazione faremo un'analisi preliminare del precedente applicativo, analizzando principalmente i punti di forza e di debolezza del progetto che sono risultati essere il principale punto di partenza per la definizione del nuovo layout. Parlando del nuovo applicativo sviluppato approfondiremo la sua complessità strutturale, le entità protagoniste e le principali interazioni, azioni e funzionalità. Il quinto, ed ultimo capitolo, si concentrerà invece sull'analisi della fase di testing del prodotto sviluppato. Nella prima parte ci concentreremo sullo studio delle principali metodologie di testing per poi, nella seconda parte, analizzare i risultati ottenuti e le eventuali scelte e decisioni che sono state adottate per la risoluzione dei problemi riscontrati.

Capitolo 2

La User Experience

In questo capitolo cercheremo di approfondire meglio e sistematizzare le conoscenze sull'usabilità Web che sono state utilizzate e approfondite nello sviluppo dell'intero progetto. Obiettivo di questa sezione è indicare e tracciare alcuni principi guida, che chi si occupa di sviluppo Web non può ignorare e per approfondire in modo organico e completo tale argomento, sono stati seguiti gli studi e le analisi portate avanti dalla società Nielsen Norman Group ¹. In oltre vent'anni di attività hanno individuato migliaia di problemi di usabilità e sviluppato altrettante linee guida con il fine di prevenirli. Le ricerche condotte hanno portato ad oltre cinquemila pagine scritte; il processo che ha permesso di arrivare a distillare le informazioni più preziose di quelle analisi in un unico volume ed utilizzarlo come guida è stato complesso, ma al tempo stesso assolutamente fondamentale per tutti coloro che si occupano di User Experience.

Per potermi orientare all'interno di un così vasto argomento, ho scelto di utilizzare come punto e base per lo studio dell'usabilità proprio le ricerche e le linee guida frutto delle osservazioni e analisi condotte dal Nielsen Normal Group in quanto esse derivano direttamente da studi comportamentali e non da opinioni di esperti. ² Tutti i test portati avanti negli anni hanno avuto come unico fulcro centrale l'utente, che è stato considerato come il solo punto di partenza per le valutazioni e da cui è stato possibile raccogliere linee guida dettate dall'osservazione stessa dell'utente nella sua interazione con l'interfaccia. Nei test, ad ogni utente, è sempre stato chiesto di portare a termine dei compiti pratici e realistici, ed è utilizzando questa specifica modalità che è stato possibile ottenere dati e informazioni specifiche sull'utilizzo diretto del mondo del Web.

2.1 La definizione

Prima di poter partire studiando e approfondendo quelli che possono essere definiti i principi dell'usabilità Web è bene darne, o quanto meno provare a farlo, una sua definizione.

¹Il Nielsen Norman Group è una società di consulenza fondata nel 1988 da Jakob Nielsen, Don Norman e Bruce Tognazzini che si occupa di User ed Experience interface. Si sono recentemente occupati dello studio dell'analisi dell'interfaccia del sistema operativo Microsoft's Windows 8.

²Documentazione consultabile alla pagina <https://www.nngroup.com/> [6]

Risulta essere molto complicato dare una spiegazione univoca, infatti il termine usabilità ³ può essere spesso riportato e spiegato utilizzando uno dei seguenti attributi:

- utile perchè fa qualcosa di cui ha bisogno la persona;
- facile da capire: le persone riescono a capire subito come si utilizza ed il suo funzionamento;
- facile da memorizzare: le persone non devono investire energie ogni volta che lo utilizzano per capirne il suo utilizzo riducendo la fatica per ottenerlo;
- idoneo: raggiunge in modo semplice e veloce uno specifico obiettivo;
- gradito: la gente farebbe di tutto pur di averlo e risolvere il suo problema attraverso proprio quello strumento;
- attraente: il suo utilizzo non è pesante, ma al contrario divertente.

Una definizione ritenuta esaustiva e completa, ma allo stesso tempo estremamente semplice è stata fornita da Jakob Nielsen. Se qualcosa è usabile, un sito Web, un forno a microonde o un'applicazione per lo smartphone significa che: una persona con una capacità e un'esperienza nella media (e perchè no, anche al di sotto della media) può capire come si utilizza un determinato strumento per raggiungere uno scopo senza dover risolvere più problemi di quanto valga la pena ⁴. L'usabilità può essere quindi considerata come un indicatore di qualità e ci permette di raccogliere informazioni su quanto risulti essere facile o meno utilizzare un determinato oggetto. Possiamo utilizzarla come indice di quanto sia necessario investire in energia al fine di imparare ad usare quello strumento, con quanta efficienza la si possa utilizzare, quanto sia semplice tenerne a mente il funzionamento e allo stesso tempo quanto sia alta la probabilità di fare errori.

2.2 I principi dell'usabilità

Il Web è cambiato molto radicalmente negli ultimi anni, non si è modificato solamente nel tipo di informazioni che vi possiamo trovare, ma anche per come vi facciamo accesso, ovvero tutte le tecnologie che utilizziamo. ⁵ All'inizio delle prime ricerche l'utente faceva accesso al Web attraverso modem da 28.8 Kbps, ma oggi lo stesso utente medio dispone

³Definizione del termine usabilità - 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». In particolare, in informatica, u. del Web, la disciplina che regola la costruzione del sito sulla base delle esigenze dell'utente, cercando di semplificare la sua esperienza di navigazione.

⁴ J. Nielsen, H. Horanger *Web Usability 2.0. L'usabilità che conta*, Apogeo, 2010. [1]

⁵1991, nasce il Web come strumento di raccolta di articoli di fisica. Per i primi due anni viene utilizzato come un sistema ipertestuale di solo testo ed è utilizzato unicamente da ricercatori. 1993: viene generata attraverso Mosaic la prima interfaccia grafica. Lo sviluppo è continuato attraverso differenti generazioni di tecnologie. Il processo evolutivo ha portato ad utilizzare pagine colorate basate su tabelle Netscape per arrivare alle pagine multilayer ricche di svariate funzionalità di Microsoft Explorer, Safari, Firefox e molti altri sviluppate attraverso linguaggi quali: Javascript e CSS.

di connessioni nell'ordine delle centinaia di Megabit per secondo. I radicali sviluppi tecnologici non hanno rivoluzionato i problemi di usabilità, gli stessi principi studiati anche solo negli anni 2000 sono ancora validi e attuali. Le linee guida rimangono invariate negli anni in quanto l'usabilità è unicamente un problema comportamentale umano e l'uomo non si evolve in un decennio; le caratteristiche che lo identificano sono rimaste pressochè invariate e così i suoi comportamenti. Molti principi di usabilità sono diventati, proprio grazie allo sviluppo e l'evoluzione dei browser, della banda e di altre tecnologie Internet meno problematici.

Attenzione però ad utilizzare in modo corretto il concetto di principio, infatti non sempre esiste un'univoca risposta alla maggior parte dei problemi di usabilità che all'interno di un progetto possono emergere. Lo sviluppo e la progettazione sono processi complicati e la risposta a molte domande può essere, nella maggior parte delle situazioni "dipende". Ad oggi resta impossibile delineare lo sviluppo ed i cambiamenti futuri dei principi di usabilità utilizzati, anche la più piccola supposizione potrebbe infatti rivelarsi non corretta.

In questa sezione, si cercherà di delineare i principi dell'usabilità che sono stati utilizzati e presi in considerazione durante il processo di progettazione dell'applicativo Web. Per ogni principio e linea guida che verrà analizzata, sarà fatto un collegamento diretto al progetto GUCH con lo scopo di esaminare le motivazioni che hanno portato a prendere in considerazioni determinate scelte.

2.2.1 Prima legge dell'usabilità

La prima legge dell'usabilità, può essere riassunta utilizzando un'unica frase: "Non costringermi a pensare."⁶ Quando si guarda una pagina Web, essa deve essere autoevidente, essere ovvia; tutte le informazioni che sono presenti devono essere chiare e non devono costringere l'utente che ne fa uso ad alcun sforzo mentale. L'obiettivo in fase di progetto deve essere quello di far sì che l'utente, una volta giunto su una qualsiasi pagina, guardando e interagendo con essa, non abbia dubbi sul suo utilizzo. Alleviare, se non addirittura eliminare ogni possibile sforzo cognitivo dell'utente alle prese con l'interfaccia deve essere uno dei principali scopi di colui che si occupa di questo tipo di progettazione.

Possono essere fonte di dubbio e far fermare l'utente a pensare:

- nomi poco autoesplicativi, esempio l'utilizzo di nomi e nomenclature che fanno uso di termini poco noti che confondono l'utente dai suoi reali scopi;
- link e pulsanti poco riconoscibili o nascosti da altri elementi. L'utente non deve perdere tempo a pensare se quell'elemento sia cliccabile o meno.

Ogni punto interrogativo che si genera nell'utente, ogni dubbio che si fa scaturire e che si alimenta in lui sull'usabilità dell'interfaccia comporta necessariamente un aumento del carico cognitivo; questo fa sì che l'utente sia inevitabilmente portato ad essere distratto dai suoi scopi.

⁶ S. Krug *Don't make me think. Un approccio di buon senso all'usabilità Web e mobile.*, Tecniche nuove, Milano, 2016. [4]

Non sempre però è possibile rendere ogni elemento ed informazione autoevidente, ma talvolta si è costretti a rendere qualcosa di complicato e complesso solamente autoesplicativo.

Per far sì che l'interfaccia si presenti all'utente autoesplicativa si è scelto di utilizzare:

- nomi esemplificativi. Si è scelto di utilizzare termini chiari sia ad un medico che al paziente preferendo utilizzare comunque gli stessi termini al fine di creare un ambiente il più simile possibile, anche se le due entità necessitano di funzionalità differenti; ⁷
- icone chiare e di semplice comprensione da utilizzare in sostituzione al testo;
- suddivisione in tab per le varie sezioni;
- utilizzo di tabelle per la presentazione di elenchi di persone. ⁸

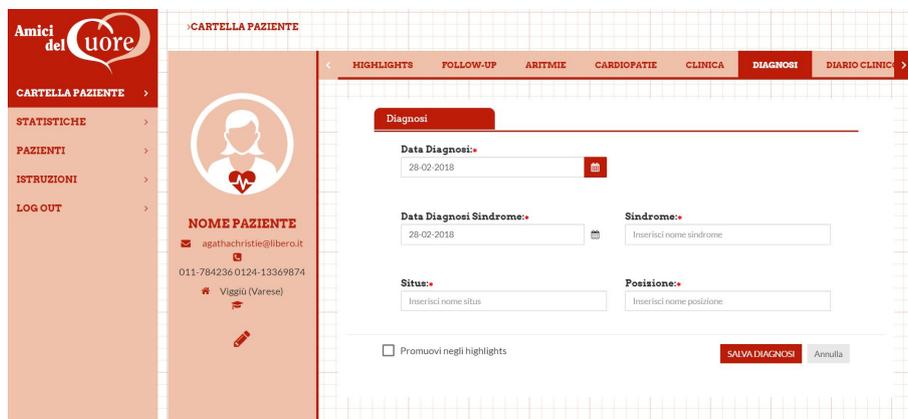


Figura 2.1: Esempio di utilizzo dei tab all'interno dell'applicativo per la suddivisione delle aree di interesse - Scheda paziente

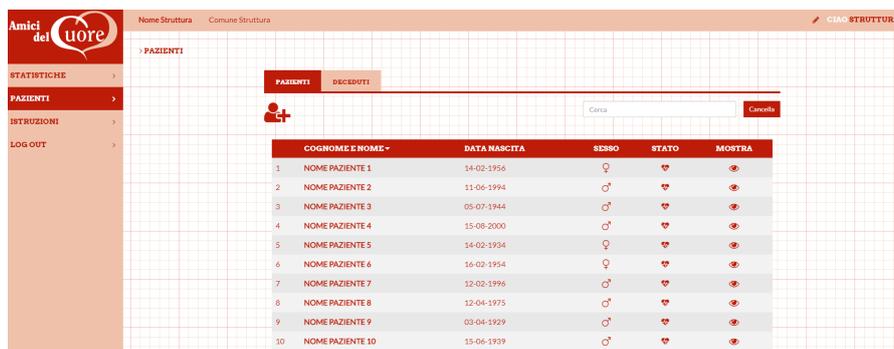


Figura 2.2: Esempio di utilizzo delle tabelle per la presentazione di elenchi - Elenco dei pazienti.

⁷Per un maggior approfondimento sulle funzionalità specifiche dell'entità medico e dell'entità paziente, vedere Capitolo 4 - La navigazione. 4.1

⁸Per ragioni di privacy non è possibile riportare nomi di pazienti. Per questa ragione a scopo di sviluppo si sono utilizzati nomi di fantasia.

2.2.2 Le tre verità assolute sul mondo del Web

Le persone, per come si sono abituate ad utilizzare il mondo del Web negli anni, nella maggior parte dei casi si limitano a dare un rapido sguardo complessivo alla pagina. Pertanto l'utente guarda molto velocemente parte del testo ed in genere clicca sul primo link che cattura il suo interesse o che sembra rispondere a quello che sta cercando. Ovviamente le cose risultano essere leggermente più complicate di così e dipendono fortemente dal tipo di pagina, da quale obiettivo ha l'utente e soprattutto quanta fretta ha nel portare a termine il suo compito; per questa ragione, quando si progettano pagine Web è utile prendere in considerazione tre dati.

Scorrimento dei contenuti

L'utente è tendenzialmente pigro e pretende di avere tutte le informazioni che gli interessano sott'occhio. Di fronte a pagine in cui sono inserite notizie, documenti o comunque informazioni importanti il fruitore alternerà momenti di lettura a momenti di rapido scorrimento. Per questa ragione gli stessi utenti nella maggior parte dei casi non fanno scorrere mai integralmente la pagina accontentandosi di guardarne una piccola porzione. Secondo gli studi del Nielsen Norman Group mediamente l'utente scorre una pagina su tre, da tale ricerca si deduce che per una pagina decisamente lunga e con molto testo, la probabilità che non venga guardata e fatta scorrere è decisamente elevata.

All'interno della nostra applicazione, per far fronte a tale problematica, sono stati utilizzati due accorgimenti:

- utilizzo del campo ricerca. È stato inserito nella pagina "Pazienti" che contiene l'elenco dei pazienti registrati e per cui il medico può visionare la scheda personale. È un campo che consente di ricercare molto più velocemente un paziente utilizzando come input il suo nome e/o il suo cognome e la sua data di nascita;



	PAZIENTI	DECEDUTI				
			Cerca		Cancella	
	COGNOME E NOME ~	DATA NASCITA	SESSO	STATO	MOSTRA	
1	NOME PAZIENTE 1	14-02-1956	♀	♥	👁	
2	NOME PAZIENTE 2	11-06-1994	♂	♥	👁	
3	NOME PAZIENTE 3	05-07-1944	♂	♥	👁	
4	NOME PAZIENTE 4	15-08-2000	♂	♥	👁	

Figura 2.3: Posizionamento del campo ricerca all'interno della pagina.

- utilizzo della pagination. In questo modo è stato possibile suddividere i vari elenchi (esempio: elenco dei pazienti) in pagine. In questo modo all'utente non viene presentato un unico elenco che deve scorrere integralmente per giungerne alla fine, ma viene presentata una porzione di N voci. Il numero di pagine generate viene dinamicamente aggiornato al crescere del numero di pazienti registrati in elenco;

15	ATRESIA POLMONARE CON DIV	Cardiopatie - polmonare	
16	BICUSPIDIA AORTICA	Cardiopatie - aorta	
17	CAV COMPLETO	Cardiopatie - CAV	
18	CAV FALLOT	Cardiopatie - CAV	
19	CAV INTERMEDIO	Cardiopatie - CAV	
20	CAV PARZIALE	Cardiopatie - CAV	

[Prima](#)
[Prec](#)
1
[2](#)
[3](#)
[4](#)
[Succ](#)
[Ultima](#)

Figura 2.4: Esempio di utilizzo della pagination per l'organizzazione degli elenchi.

- ordinamento. Le voci in tabella sono ordinabili per nome e cognome e data di nascita, la disposizione può essere crescente o decrescente. In questo modo la ricerca di eventuali voci è ulteriormente facilitata.

```

1
2 <div id="pagination-aritmie" class="pagination-container" ng-if="
  filteredAritmie.length > $ctrl.itemPage">
3 <ul uib-pagination total-items="filteredAritmie.length" ng-model="$ctrl.
  currentPage" max-size="$ctrl.maxSize" class="pagination-sm" boundary-
  links="true" boundary-link-numbers="true" force-ellipses="true" items-
  per-page="$ctrl.itemPage" first-text="Prima" last-text="Ultima" next-
  text="Succ" previous-text="Prec">
4 </ul>
5 </div>

```

Listing 2.1: Pagination - HTML - estratto di codice

```

1
2 <table id="table-aritmie" class="table table-hover">
3 <thead>
4 <tr>
5 <th></th>
6 <th>
7 <a ng-click="sortBy = 'aritmia'; sortReverse = !sortReverse">
  Aritmia
8 <span ng-show="sortBy == 'aritmia' && !sortReverse" class="fa
  fa-caret-down"></span>
9 <span ng-show="sortBy == 'aritmia' && sortReverse" class="fa
  fa-caret-up"></span>
10 </a>
11 </th>
12 <th>
13 <a ng-click="sortBy = 'aritmiecategorie'; sortReverse = !
  sortReverse"> Categoria<span ng-show="sortBy == 'aritmiecategorie' &&
  !sortReverse" class="fa fa-caret-down"></span>
14 <span ng-show="sortBy == 'aritmiecategorie' && sortReverse"
  class="fa fa-caret-up"></span>
15 </a>
16 </th>
17 </th>
18 <th class="action"> Modifica </th>
19 </tr>

```

```

20 </thead>
21 <tbody>
22 <tr ng-repeat="item in filteredAritmie = ($ctrl.aritmie | filter:
    searchAritmia) | orderBy : sortBy: sortReverse | pagination : ($ctrl.
    currentPage - 1) * $ctrl.itemPage | limitTo: $ctrl.itemPage track by
    $index">
23 <td {{ $index +1 + ($ctrl.currentPage -1)*$ctrl.itemPage }} </td>
24 <td class="topic"> {{ item.aritmia }} </td>
25 <td {{ item.aritmiecategorie }}</td>
26 <td class="action">
27 <a ng-click="modalModificaAritmia(item)">
28 <i class="fa fa-pencil" aria-hidden="true"></i>
29 </a>
30 </td>
31 </tr>
32 <tr ng-if="filteredAritmie.length == 0" class="empty-list">
33 <td colspan="4">Nessun elemento da visualizzare</td>
34 </tr>
35 </tbody>
36 </table>

```

Listing 2.2: Ordinamento - HTML- estratto di codice

Si fanno sempre scelte soddisfacenti

L'utente non scorre mai una pagina nella sua interezza, di conseguenza non è portato a valutare quale sia la scelta migliore fra più possibilità, tendendo sempre a optare per la prima scelta ragionevole. Questo processo è stato definito dall'economista Herbert Simon *satisficing* incrociando due termini *satisfying* (soddisfare) e *suffing* (sufficienza). Ciò accade in quanto l'utente ha in genere fretta ed una scelta sbagliata non rappresenta un problema nel mondo del Web, in quanto è sempre possibile tornare indietro e riprovare. Spesso anche valutando attentamente quali possano essere le possibilità non siamo immuni agli errori.

Non si comprende come funziona

Molto spesso i fruitori di un prodotto si fanno un'idea completamente sbagliata dell'applicativo e di conseguenza del suo funzionamento. All'utente è sufficiente sapere di essere in grado di portare a termine un compito; non è importante per lui conoscere come quello stesso scopo potrebbe essere risolto e soprattutto non ha interesse nel sapere come è stato pensato. Se un utente si imbatte in un problema, cerca di risolverlo utilizzando la procedura imparata e memorizzata, difficilmente cerca di proposito una metodologia migliore, a meno che non si imbatte in essa casualmente. Far sì che l'utente sia in grado di comprendere determinati meccanismi è fondamentale, perchè lo preserva da commettere errori e/o dal procedere in modo inefficiente.

Per dare ai medici tutti gli strumenti utili per utilizzare l'applicativo è stato preparato per loro una piccola guida. Il manuale li accompagna nella gestione delle principali funzioni (esempio: registrazione di un nuovo paziente, modifica dei dati di un paziente) illustrando in modo semplice ed intuitivo i pochi passi che devono essere compiuti.

2.2.3 Le convenzioni

Per assicurarci che i fruitori della nostra pagina Web possano comprendere il più possibile le informazioni che l'applicativo veicola e affinché sia possibile per lo sviluppatore farle arrivare nel modo più semplice ed adeguato possibile è utile sfruttare a proprio vantaggio le convenzioni; le consuetudini infatti, permettono di rendere tutto più facile da cogliere. Reinventarsi tutto quello che è già stato pensato e consolidato non fa che generare solamente pagine Web troppo difficili da utilizzare, quindi, al fine di eliminare, o quanto meno ridurre la confusione è utile adottare il più possibile strumenti già affermati. Per un utente poter contare su standard consolidati e facilmente riconoscibili lo aiuta a rafforzare la sua percezione ed avere controllo sull'intera applicazione; raggiungere un obiettivo in modo diretto per il fruitore, è sinonimo di soddisfazione. Secondo gli studi portati avanti da Jakob Nielsen, gli standard fanno sì che gli utenti:⁹

- sappiano che tipo di funzionalità aspettarsi;
- sappiano come tali funzionalità verranno presentate graficamente;
- sappiano dove trovare una specifica funzionalità;
- sappiano come dover utilizzare una determinata funzionalità al fine di raggiungere uno specifico obiettivo;
- non debbano rompersi il capo sul significato di elementi sconosciuti dell'interfaccia (esempio significato di icone, termini linguistici ecc...);
- non rischino di trascurare caratteristiche importanti del sito, ma nascoste sotto un elemento non classificabile come standard dell'interfaccia;
- non abbiano brutte sorprese quando qualcosa non funziona e sappiano al contrario farvi fronte.

Le convenzioni non sono altro che il frutto di un'idea brillante, che è stata poi replicata più e più volte in diversi siti e applicativi fino al punto che per gli utenti quell'elemento è diventato familiare e non ha bisogno di essere spiegato.

Reinventare, ovvero ri-disegnare e ri-progettare una convenzione già consolidata può portare a perdere del tempo; essere innovativi, spesso per un designer significa proprio capire quale sia il valore di quello che si sta pensando di sostituire; molte volte si sottovaluta il significato di uno standard e la grande quantità di tempo che è stato investito al fine di ottenere il risultato perfetto.

È possibile non utilizzare una convenzione già esistente, ma se così fosse è bene essere consapevoli ed assicurarsi che l'elemento con cui la si sostituisce debba essere per prima cosa, chiaro e autoesplicativo in modo da non dover richiedere nessuna curva di autoapprendimento o in caso contrario che comporti un valore aggiunto tale da giustificarla.

Usufruendo delle convenzioni all'interno del progetto GUCH sono stati utilizzati ad esempio:

⁹J. Nielsen, H. Horanger *Web Usability 2.0. L'usabilità che conta*, Apogeo, 2010. [1]

- simboli e icone chiare e familiari;
- nomenclatura semplice dei bottoni;
- medesima struttura dell'applicazione per tutte le entità coinvolte. Per i diversi ruoli (analizzati in modo approfondito nel capitolo 4 (4.1)) la struttura è rimasta invariata, ma va a modificarsi la tipologia di funzionalità a cui posso far accesso, ad esempio attraverso il menu di selezione posto nella banda laterale sinistra. In questo modo anche gli utenti coinvolti che contemporaneamente ricoprono due ruoli differenti non corrono il rischio di essere disorientati. Può infatti accadere che un medico sia allo stesso tempo anche referente regionale, in questo caso quindi alle funzionalità specifiche del ruolo medico si sommano quelle da responsabile regionale.



Figura 2.5: Esempio di icone utilizzate.

2.2.4 Gerarchie visive e definizione di aree

Per fare in modo che le pagine siano al tempo stesso facili da comprendere e da navigare, che ogni elemento rifletta accuratamente le relazioni che concorrono fra loro, quali siano gli elementi più importanti, quale tipo di relazione sussista e quali siano parte di altre aree è necessario definire una gerarchia visiva. Questo fa sì che:

- gli elementi più importanti debbano essere posizionati, dimensionati, colorati al fine di renderli più evidenti;
- gli elementi che sono in relazione fra di loro debbano essere collegati visivamente. Gli elementi collegati possono essere ad esempio posizionati sotto un titolo, accumulati da un medesimo stile visivo o raggruppati all'interno di una zona ben definita;
- gli elementi debbano essere visivamente nidificati per rendere ben visibile quali elementi facciano parte di quell'area.

È importante notare come anche in questo caso ritorni il discorso precedentemente fatto sulle convenzioni; ogni utente infatti è abituato ad utilizzare ordinamenti visivi e leggere

informazioni gerarchicamente strutturate. Basti pensare alle pagine di giornale e all'organizzazione delle informazioni che utilizzano, tutte le pagine mettono in rilievo, raggruppano e nidificano in modo da presentare al fruitore informazioni utili sul contenuto delle pagine prima ancora di incominciare a leggerne il testo.

Una buona gerarchia visiva e di conseguenza un'opportuna organizzazione dei contenuti permette ad ogni utente di risparmiare fatica pre-elaborando il contenuto della pagina, ordinando e assegnando la corretta priorità ad ogni elemento in modo da coglierli istintivamente. Al contrario, per un'utente, guardare una pagina poco organizzata gerarchicamente significa costringerlo ad un processo molto più lento nella ricerca di parole chiave ed elementi rivelatori.

Collegato a ciò possiamo introdurre il concetto di area; una pagina ben costruita consente all'utente di prendere in visione, di identificare le aree e di conseguenza, intuire più facilmente quali informazioni è possibile reperire al suo interno.

Gli studi di eye-tracking ¹⁰ dimostrano come l'utente medio in pochi secondi decida quali siano gli elementi e le informazioni utili e quali parti della pagina stessa possano essere trascurate. Una volta selezionate le aree da escludere difficilmente vi fa ritorno.

Presi in considerazione questi due principi, sono stati integrati all'interno del progetto utilizzando:

- Menu laterale. L'utente ha, a fianco a se, le voci delle sezioni a cui può far accesso. In questo modo ogni sua principale azione è facilmente accessibile ed in qualunque momento è in grado di sapere in quale sezione si trovi ¹¹.



Figura 2.6: Esempio menu laterale per l'entità nazionale.

¹⁰Eye-tracking è una metodologia operativa che permette di studiare e tracciare il movimento oculare determinando così con precisione dove l'utente sta guardando e di conseguenza ponendo la sua attenzione. Tali studi sono utili al fine di comprendere: il comportamento del soggetto durante l'esecuzione di una determinata attività, valutare in modo oggettivo l'usabilità.

¹¹Per ragioni di privacy non è possibile riportare nomi di pazienti. Per questa ragione a scopo di sviluppo si sono utilizzati nomi di fantasia.

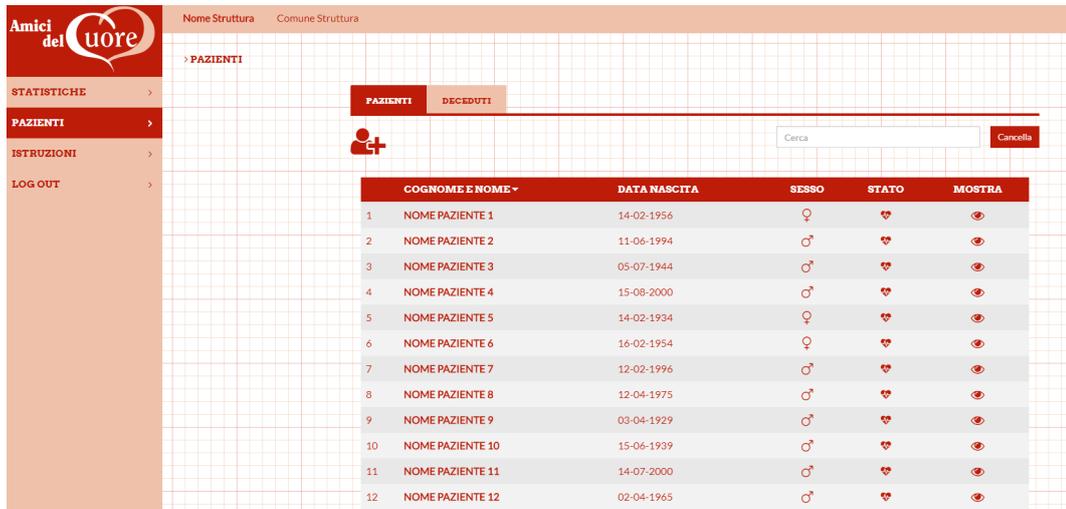


Figura 2.7: Esempio menu laterale per l'entità struttura.

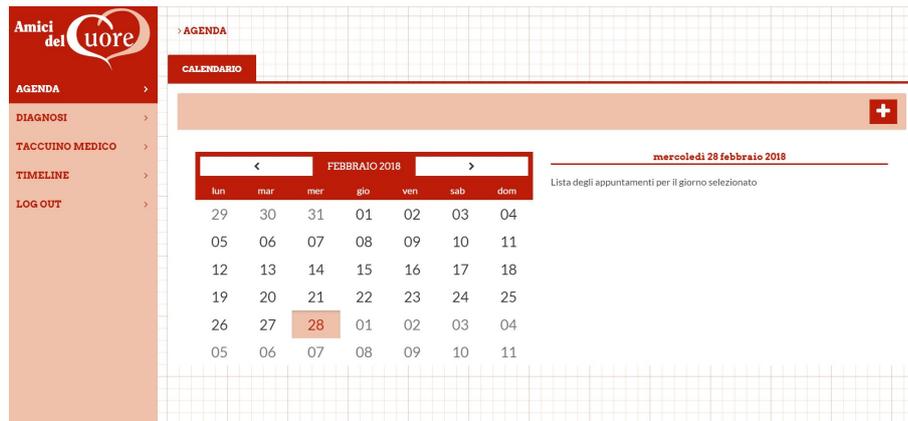


Figura 2.8: Esempio menu laterale per l'entità paziente relative alla scheda paziente.

- Tab. Dovendo presentare ad ogni entità molte informazioni differenti si è scelto di organizzarle utilizzando dei tab. Essi consentono di etichettare il contenuto di una determinata area rendendo evidente all'utente quali informazioni è possibile reperire all'interno del settore scelto e allo stesso tempo mantenendo visibile a quali altre informazioni è possibile accedere. Si è scelto di organizzare i sotto contenuti delle differenti aree utilizzando proprio le etichette, in quanto nella loro veste grafica richiamano le cartelle (schede mediche o divisori per schedari) utilizzate negli ospedali dal personale medico per aggiornare lo stato di salute di ogni paziente.

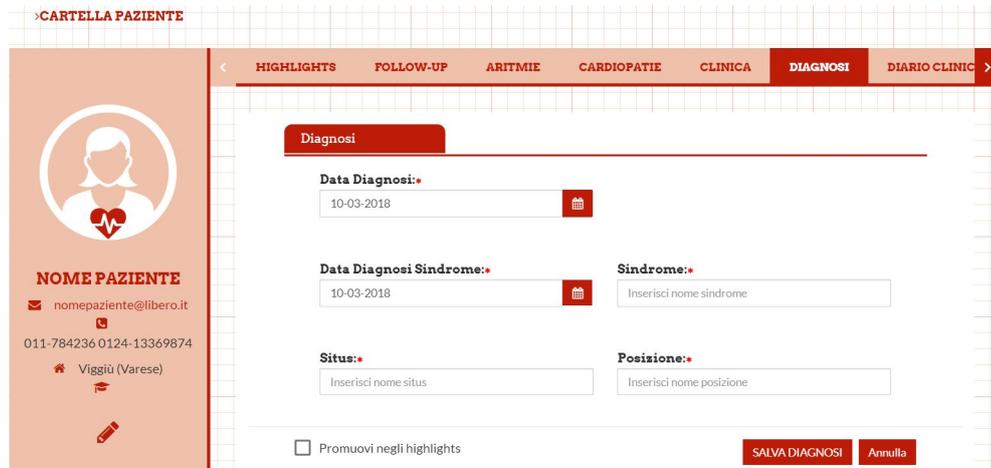


Figura 2.9: Esempio utilizzo dei tab per la presentazione delle informazioni.

```

1
2 <md-tabs md-dynamic-height md-no-ink md-selected="2">
3   <md-tab id="tabHighlights">
4     <md-tab-label>Highlights</md-tab-label>
5     <md-tab-body>
6       <!-- ... -->
7     </md-tab-body>
8   </md-tab> <!-- end tab highlights -->
9   <md-tab id="tabDiagnosi">
10    <md-tab-label>Diagnosi</md-tab-label>
11    <md-tab-body>
12      <div class="container-fluid">
13        <div class="tab-cartella-container">
14          <div class="modal-body">
15            <div ng-include="layoutDiagnosi.url"></div>
16          </div>
17        </div>
18      </div>
19    </md-tab-body>
20  </md-tab> <!-- end tab diagnosi -->
21 </md-tabs>

```

Listing 2.3: Tab - HTML - estratto di codice

2.2.5 I bottoni e i link

La principale attività che l'utente intraprende all'interno del mondo del Web può essere identificata con l'attività di ricerca del prossimo elemento su cui è possibile cliccare per produrre una nuova azione, pertanto è importante rendere indubbio quali siano gli elementi che il soggetto può selezionare. Quando si scorre una pagina l'utente cerca indizi visivi che gli consentono di capire quali siano i componenti cliccabili; segni distintivi di elementi selezionabili possono essere la forma, la posizione e la formattazione. È ormai una consolidata convenzione per l'utente vedere il cursore a forma di freccia trasformarsi in un cursore a

mano quando esso si trova su di un elemento cliccabile (che sia un link , un bottone o un qualsiasi altro elemento). Attenzione però che questa consuetudine non è applicabile ed utilizzabile nel caso di dispositivi touch screen, è quindi necessario in tali casi utilizzare altre strategie visive.

Trattando questo tema è necessario citare la *Legge di Fitts* ¹². Essa stabilisce che il tempo necessario per raggiungere uno specifico bersaglio con uno strumento di puntamento (mouse, trackball, joystick) è proporzionale al logaritmo della distanza da coprire diviso la dimensione del bersaglio. Di conseguenza più lontano sarà l’oggetto, maggiore sarà il tempo necessario per arrivare a cliccarlo. La legge stabilisce che il tempo necessario cresce in modo logaritmico, quindi il suo tasso di crescita non varia molto rapidamente infatti, dovendo raggiungere un punto lontano sullo schermo si compiono movimenti più veloci. La legge enunciata da Fitts afferma anche che più grande è il bersaglio, minore sarà il tempo necessario a cliccarlo; maggiori sono le sue dimensioni, più bassa è la precisione che si deve utilizzare. Tale legge è stata ripetutamente confermata empiricamente.

Negli anni sono stati raccolti molti dati statistici che mettono in evidenza come il puntatore del mouse graviti in media più spesso vicino alla barra di scorrimento laterale (a destra) per tali ragioni è conveniente posizionare gli elementi che si intendono far notare e che si vogliono rendere maggiormente accessibili proprio nella porzione destra dello schermo.

Tenendo conto della Legge di Fitts, tutti i bottoni di conferma (ad esempio i bottoni di Salva, Ok) o di annullamento sono stati posizionati nella parte destra dello schermo, per la stessa ragione è stato posizionato in alto a destra anche il campo input che consente la ricerca. Nella parte sinistra è stato invece posizionato il menu. A tutti i bottoni è stato assegnato il medesimo stile (esempio: colore, forma) e l’area cliccabile è stata estesa a tutta la superficie del bottone e nel caso di tabelle (esempio: tabella elenco pazienti ¹³) a tutta la riga.

2.2.6 Ridurre ciò che crea disturbo

Il rumore in una pagina Web è rappresentato da tutto ciò che crea disturbo visivo e per l’utente, avere degli elementi di intralcio significa necessariamente perdere delle informazioni rilevanti. Crea disturbo visivo:

- la disorganizzazione. Pagine che non presentano uno schema definito e preciso si presentano per il fruitore caotiche e dissonanti;
- il sovraffollamento. Viene causato quanto vengono unite troppe informazioni differenti all’interno della medesima pagina;

¹² È stata enunciata nel 1954 da Paul Fitts, ricercatore presso la Human Engineering Division della US Air Force.

¹³ Cliccando sulla riga di un determinato paziente è possibile accedere alla sua scheda personale. È possibile accedere quindi a tale sezione facendo click in un qualunque punto della riga e non necessariamente nel punto in cui è presente l’icona mostra.

- troppe informazioni importanti. Non è conveniente inserire all'interno della medesima pagina troppe informazioni classificabili come rilevanti. Fondamentale è studiare quali sono i dati realmente essenziali che devono essere evidenziati utilizzando font e colori differenti;
- parole chiavi non evidenziate;
- testo e paragrafi non formattati. È utile, tutte le volte in cui è possibile e soprattutto conveniente farlo, utilizzare elenchi puntati per presentare al soggetto delle informazioni, l'enumerazione facilita la lettura e la leggibilità del testo.

All'interno dell'interfaccia di GUCH per limitare il rumore presentato all'utente si è scelto di utilizzare:

- tab, pagination e tabelle per organizzare gli elementi ed evitare il sovraffollamento delle sezioni e la loro disorganizzazione;
- due tipologie di font utilizzati in modalità grassetto minuscolo, grassetto stampatello in base alle necessità;
- utilizzo di un artificio estetico per la presentazione delle informazioni. In questo modo è stato possibile organizzare i dati in modo semplice ma funzionale pur non conoscendo a priori la quantità di informazioni da mostrare e garantendo un risultato soddisfacente sia in presenza di molti che di pochi dati.

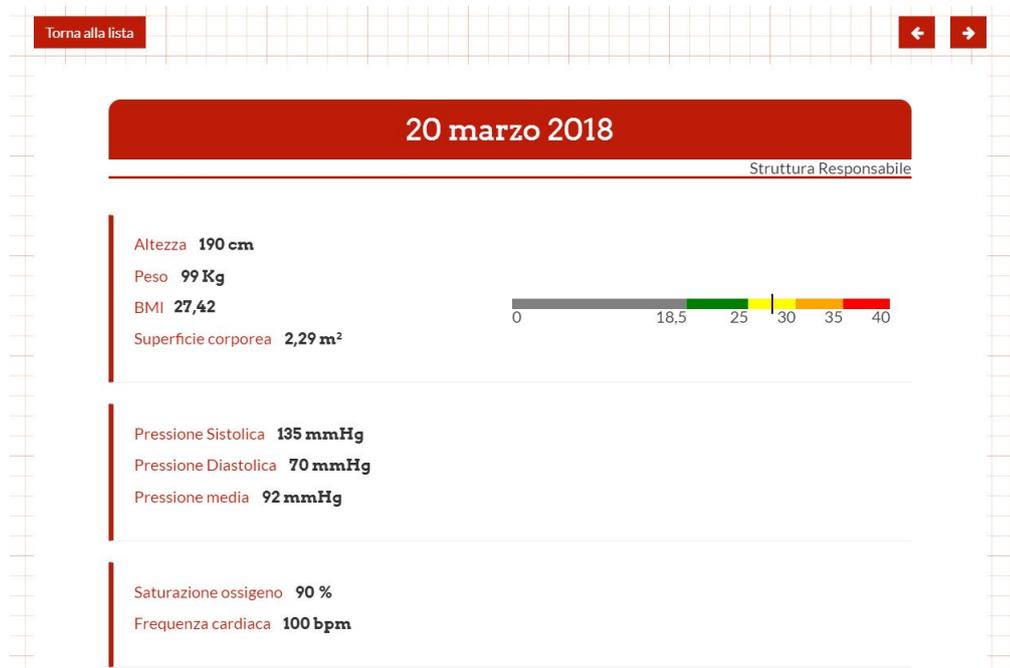


Figura 2.10: Esempio di riferimento dell'artificio estetico utilizzato.

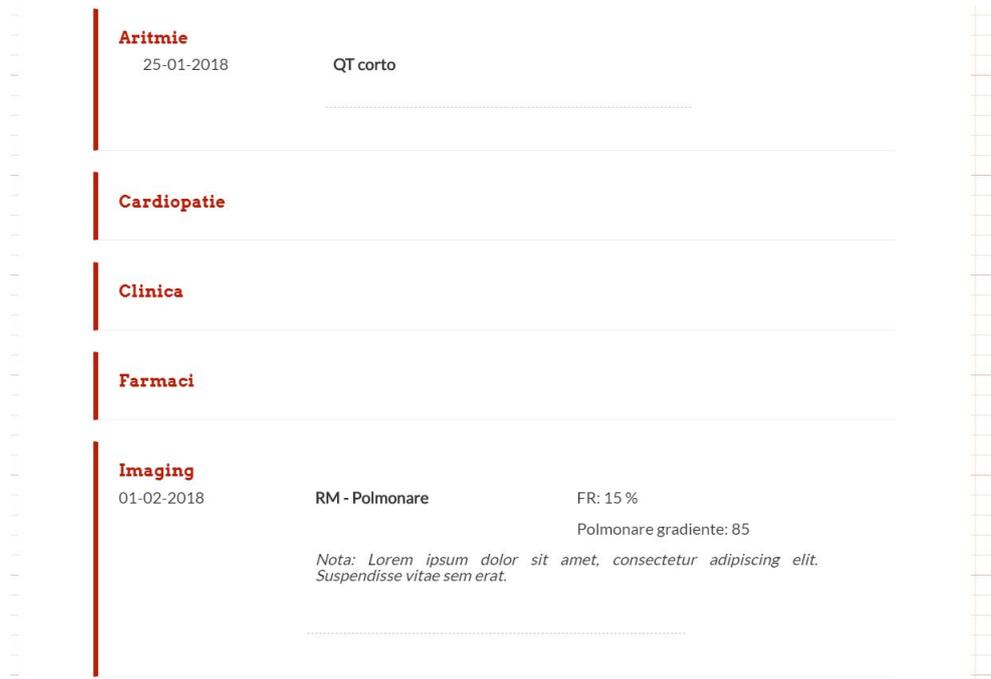


Figura 2.11: Esempio di riferimento dell'artificio estetico utilizzato.

```

1
2
3 <div class="bs-callout bs-callout-danger">
4   <p>
5     <span class="etichetta">Altezza &nbsp;&nbsp;&nbsp;</span>
6     <span>{{oggShowFollowUp.altezza}} cm</span>
7   </p>
8
9   <p>
10    <span class="etichetta">Peso &nbsp;&nbsp;&nbsp;</span>
11    <span>{{oggShowFollowUp.peso}} Kg</span>
12  </p>
13
14  <p>
15    <span class="etichetta">Massa corporea &nbsp;&nbsp;&nbsp;</span>
16    <span>{{oggShowFollowUp.massacorporea | number : 2}}</span>
17  </p>
18
19  <p>
20    <span class="etichetta">Superficie corporea &nbsp;&nbsp;&nbsp;</span>
21    <span>{{oggShowFollowUp.superficiecorporea | number : 2}} m<sup>2</sup>
22    </span>
23  </p>
24 </div>
<br>

```

Listing 2.4: Artificio estetico - HTML - estratto di codice

```
1 .bs-callout {
2   border-color: @white;
3   border-bottom-color: #eee;
4   border-image: none;
5   border-radius: 1px;
6   border-style: solid;
7   border-width: 1px 1px 1px 5px;
8   margin-bottom: 5px;
9   padding: 20px;
10
11   h3{
12     font-family: @serif;
13     font-weight: bold;
14     font-size: 17px;
15     padding-bottom: 10px;
16   }
17 }
```

Listing 2.5: Artificio estetico - LESS - estratto di codice

2.2.7 Un click di distanza

In molti studi si legge che il numero di click rappresenta un ottimo indice di usabilità di un sito o di un'applicazione, ma oltre al mero numero è bene considerare anche la difficoltà con cui l'utente esegue quel determinato click. L'importante quindi sembrerebbe non far perdere all'utente la convinzione che sia sulla strada corretta seguendo quello che viene definito da specialisti del settore *"information foraging"* ¹⁴. La teoria afferma che l'utente naviga nel Web al pari di un animale in caccia, fiutando tutte le tracce informative disponibili e attivando i vari collegamenti che ritiene utile per il raggiungimento della specifica informazione cercata.

2.2.8 L'incompatibilità fra le piattaforme

Ruolo fondamentale hanno assunto le piattaforme e di conseguenza, nello sviluppo Web è necessario considerarle. Per un'applicazione Web gestire la compatibilità significa funzionare con browser differenti ed è bene considerare che motori di ricerca quali Firefox, Safari per Apple, Opera e Internet Explorer stanno sviluppando e rilasciando continuamente nuove versioni. Questo fa sì che ogni potenziale utente possa collegarsi e navigare l'applicazione utilizzando versioni diverse ed altri browser. Il problema della compatibilità delle piattaforme rispetto a qualche anno fa non è più così grave, ma rimane comunque una questione che deve essere considerata in fase di sviluppo; è proprio grazie al progresso tecnologico che la criticità del problema risulta essere attualmente diminuita.

Ad oggi ogni browser risulta essere più concorde agli standard affermati rispetto alle sue precedenti versioni ed è raro che un applicativo non funzioni con uno specifico motore di

¹⁴L'information foraging è stata la novità più interessante ed importante nella ricerca sull'interazione uomo-macchina. La ricerca è stata condotta da Stuart Card e Peter Pirolli ed è stata sviluppata presso Xerox PARC - Palo Alto Research Center.

ricerca, ma può succedere che determinati elementi vengano presentati in modo differente. Alle criticità legate alle possibili diverse versioni di un browser va legata l'eventualità di disporre di dispositivi (PC, schermi ecc...) dalle diverse prestazioni. Sono presenti sul mercato una grande varietà di dispositivi che possono, ad esempio disporre di diverse risoluzioni di schermo; a causa di questo la qualità di ciò che vede l'utente può essere inficiata.

Per lo sviluppo del progetto questa tematica è stata presa in considerazione, valutando ogni elemento estetico su diversi browser. Per lo più sono stati effettuati dei semplici test su Firefox, Chrome (principale browser di sviluppo), Internet Explorer e Safari (per garantire una corretta funzionalità a tutti gli utenti che dispongono di un dispositivo Mac).

2.2.9 Un layout responsive

Possono essere sviluppati due tipologie di layout: i layout congelati o bloccati e i layout liquidi. Un layout si dice bloccato o congelato quando le informazioni che contiene hanno una larghezza fissa, indipendente da quella della finestra che le contiene. Ciò comporta che se la finestra è eccessivamente piccola una parte delle informazioni non potrà essere presentata a meno di scorrere lateralmente la pagina. Evitare di realizzare layout fissi è una buona regola di User Experience per due fondamentali ragioni:

- gli utenti non si aspettano lo scorrimento laterale. Essendo uno standard ormai consolidato è preferibile assecondare le aspettative dell'utente e non violarle;
- Se la pagina può scorrere sia orizzontalmente che verticalmente il fruitore è costretto a muoversi lungo due dimensioni. Questo compito può essere complesso per tutti gli utenti che presentano difficoltà nella visualizzazione spaziale e a gestire due movimenti differenti lungo, fra l'altro, due direzioni non percepibili. Al contrario il movimento di scorrimento verticale è una metodologia semplice e radicata di navigazione per tutti gli utenti, anche quelli poco avvezzi al mondo del Web.

Per far fronte a questo problema possono essere implementati quelli che vengono definiti layout liquidi o responsive. Layout di questo tipo permettono di rispondere alle modifiche dell'ambiente in modo reattivo, riaggiustando automaticamente la dimensione e la posizione di tutti gli elementi al variare della grandezza della finestra Web o dello schermo del dispositivo. L'intero progetto GUCH è stato pensato e sviluppato per essere responsive ed essere adattabile a qualunque dispositivo.

2.2.10 URL complesse

Molto spesso, gli utenti si affidano alla lettura delle URL ¹⁵ per orientarsi all'interno di un applicativo e ottenere qualche informazione in più ed è pertanto controproducente generare URL molto complesse e lunghe. A tal proposito all'interno dell'applicativo GUCH, si è cercato di mantenere le URL il più brevi possibile ed in modo che fossero, allo stesso

¹⁵Acronimo di Uniform Resource Locator. È una sequenza di caratteri che permette di identificare in modo univoco l'indirizzo di una risorsa all'interno della rete Internet al fine di renderla disponibile per gli utenti.

tempo autoesplicative. Per ogni voce del menu laterale viene sempre presentata all'utente, in basso a sinistra, la URL della pagina che verrà caricata a seguito dell'azione di selezione. Essendo presenti numerosi form sono state utilizzate due modalità per l'invio dei dati al server.

Il primo metodo *URLencoded*, predispone il passaggio dei dati in chiaro costruendo quindi una stringa che contiene le informazioni prelevate dei campi. In questo caso quindi viene costruita a mano l'URL della pagina generando una query string. La query string non è altro che una serie di coppie nome/valore che vengono separate da una "e" commerciale o da un punto interrogativo nel caso in cui sia necessario separare la variabile dall'estensione del file corrente. Questo metodo è stato utilizzato in tutte le situazioni in cui il numero di elementi estratti dal form era limitato.

La seconda metodologia prevede lo scambio delle informazioni attraverso il metodo *formData*; in questo caso i dati vengono mandati al server attraverso l'invio di un oggetto. L'oggetto contiene al suo interno i dati di interesse precedentemente prelevati dalla compilazione dei campi del form ed in questo caso i dati non sono accessibili. Il metodo *formData* è di fondamentale utilizzo proprio nelle situazioni in cui il numero di elementi è molto elevato.

Di seguito è possibile visionare la porzione di codice che mette in evidenza la differenza di utilizzo di queste due metodologie.

```

1 let postNewAritmia = (oggetto) => {
2   var date_encoded = "";
3   let newAritmiaRequest = Restangular.one(CONFIG.rest.baseAritmie.url + '?
   aritmia=' + oggetto.name + '&idaritmiecategorie=' + oggetto.categoria.
   id);
4   let promise = newAritmiaRequest.customPOST(date_encoded, undefined,
   undefined, {
5     'Content-Type': 'application/x-www-form-urlencoded'
6   })
7   .then(function(response) {
8     return response[1];
9   });
10  return promise;
11 };

```

Listing 2.6: URLencoded - Javascript - estratto di codice

```

1 let postNewDiarioClinico = (oggetto) => {
2   var formData = cgmFormDataObj.generate(oggetto);
3   let diarioRequest = Restangular.one(CONFIG.rest.baseDiario.url);
4   let promise = diarioRequest
5   .withHttpConfig({
6     transformRequest: angular.identity
7   })
8   .customPOST(formData, undefined, undefined, {
9     'Content-Type': undefined
10  })
11  .then(function(response) {
12    return response[1];
13  });
14  return promise;};

```

Listing 2.7: formData - Javascript - estratto di codice

2.2.11 Il colore e i font

Gli elementi estetici quali ad esempio i colori, i font e le immagini utilizzate acquisiscono importanza e aiutano a dare una prima buona impressione all'utente. Curare la parte estetica aiuta l'utente nel sentirsi accolto dall'applicativo e concorre alla creazione del rapporto di fiducia che deve sussistere affinché sia utilizzato efficacemente. Il giusto utilizzo della tipografia e gradevoli accostamenti cromatici sono componenti fondamentali per determinare un buon design visuale.

Una buona leggibilità del testo è un elemento importante da tenere in considerazione e l'utente non deve essere solo facilitato ad accedere ai contenuti di suo interesse, ma deve poterli fruire con facilità. Per scegliere il giusto font ci si può far aiutare da alcune semplici regole ed è importante conoscere le due famiglie in cui essi si classificano.

I font infatti non sono tutti uguali, ma si possono ordinare in: font con (serif) e font senza grazie (bastoni); le grazie non sono altro che degli abbellimenti nel tratto del carattere, al contrario i font bastoni presentano un tratto dritto più o meno spesso. Dagli studi è emerso che i font senza grazie (bastoni) nell'ambito del Web facilitano la lettura, la risoluzione dello schermo del PC o del dispositivo che si sta utilizzando per la navigazione compromette la lettura di font con grazie; i dettagli e gli abbellimenti appaiono infatti ad una qualità minore rispetto ad una stampa tipografica dove invece la tendenza è opposta. All'interno di GUCH sono stati utilizzati unicamente due tipologie di font. Un font serif (con grazie) "Arvo" ed uno bastoni denominato "Lato". Per i titoli delle pagine ed altre scritte è stato utilizzato il font "Arvo", al contrario per la presentazione di dati (come ad esempio nel follow-up) è stato utilizzato il carattere tipografico "Lato". Si è poi sfruttato la dimensione del carattere, il peso (bold) e lo stile (italic) per richiamare l'attenzione dell'utente e sottolineare l'importanza di determinate scritte o titoli.



Figura 2.12: Font scelti.

Per dare all'utente la possibilità di personalizzare il proprio applicativo sono stati pensati tre differenti temi in cui i colori protagonisti e i dettagli estetici sono differenti. Ogni utente ha quindi la possibilità di scegliere la combinazione cromatica che preferisce, salvarla e in qualunque momento modificarla. Nei tre temi abbiamo sempre due colori in contrapposizione che vengono utilizzati per definire la colorazione dei diversi elementi e sono combinati insieme al fine di costruire una gradevole sensazione estetica complessiva. Si è scelto di utilizzare la combinazione di due soli colori al fine di non caricare eccessivamente la vista estetica ed applicarli in modo rigoroso creando una sorta di legge colore. Il design è stato utilizzato con lo scopo di aiutare la comunicazione e l'ordine, non unicamente come strumento decorativo. Di seguito alcune immagini di esempio mostrano i tre temi a confronto.

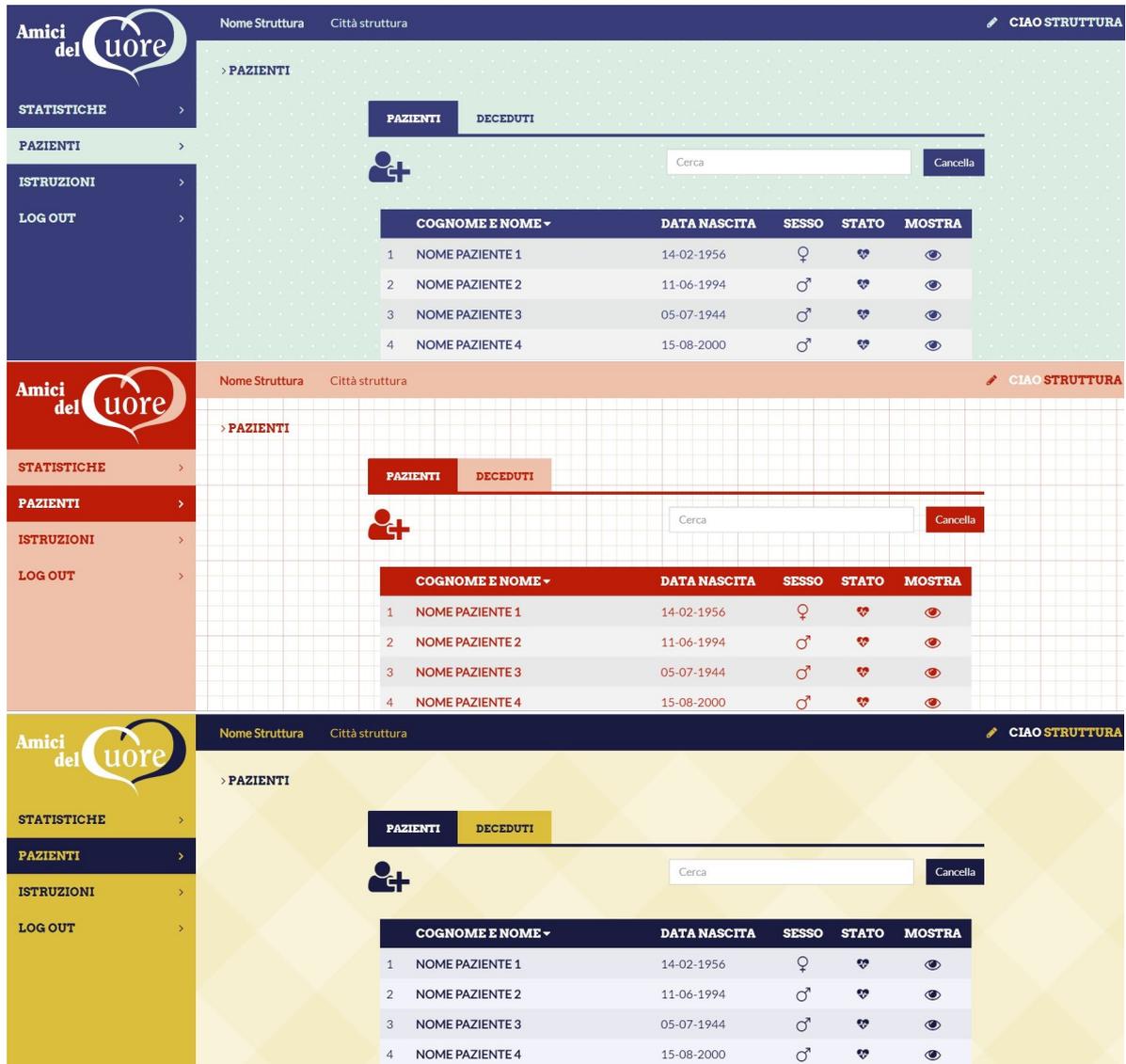


Figura 2.13: Esempio temi colore.

```

1 /* COLORI */
2 @white: #FFFFFF;
3 @grigiomorte: #7F7978;
4 //tema rosso
5 @lightred: #EFC1AA;
6 @red: #BD1C08;
7 //tema blu
8 @blue: #383D79;
9 @lightgreen: #DBECE3;
10 //tema verde
11 @acid: #DBBF3C;
12 @navy: #1A1B41;
13 /* FONTS */
    
```

```

14 @basefont: 'Lato', sans-serif;
15 @serif: 'Arvo', serif;

```

Listing 2.8: Codice colori e font - CSS

2.2.12 L'usabilità dei form

Nello sviluppo e nella progettazione dell'applicativo GUCH è stato dedicato molto tempo alla definizione del layout dei molti form presenti (esempio: form per la registrazione di un nuovo paziente, form per la registrazione di una nuova visita). I form possono essere definiti come quegli elementi interattivi della pagina Web in cui è l'utente stesso a fornire informazioni e interagire con il sito. Possiamo considerarli come il fondamento di qualsiasi applicazione Web dove, proprio attraverso il form gli utenti modificano impostazioni e gestiscono l'inserimento di nuovi dati che potranno poi, a loro volta, essere visualizzati e modificati. In anni di studi e di valutazione sono state definite, anche in questo caso, convenzioni e linee guida che sono state integrate anche all'interno dell'applicativo stesso. In questa sezione verranno indicate le principali linee guida utilizzate nella progettazione dei form.

Comprensibilità linguistica

- utilizzare, nel caso di form estremamente complessi, testi introduttivi che consentano di chiarire all'utente il senso e il funzionamento dei campi sottostanti. La spiegazione deve risultare essere semplice, focalizzata e sintetica;
- presentare informazioni di supporto in funzione del campo. Gli aiuti devono essere utilizzati per spiegare meglio il dato richiesto e guidare l'utente nella compilazione. Ad esempio nel form di registrazione di un nuovo paziente, il medico è guidato nella compilazione del campo data di nascita. Un piccolo messaggio di supporto suggerisce il formato accettato con cui la data deve essere inserita nel database.

The image shows a patient registration form with the following fields and labels:

- Nome:** Inserisci nome paziente
- Cognome:** Inserisci cognome paziente
- Sesso:** Maschio Femmina
- Telefono:** Inserisci telefono
- Email:** Inserisci email
- Data Di Nascita:** Inserisci data di nascita nel formato gg-mm-aaaa
- Provincia Di Nascita:** Inserisci provincia di nascita paziente
- Comune Di Nascita:** Inserisci comune di nascita paziente
Inserisci prima la provincia.
- Provincia Di Residenza:** Inserisci provincia di residenza paziente
- Comune Di Residenza:** Inserisci comune di residenza paziente
Inserisci prima la provincia.

Figura 2.14: Form di esempio -Registrazione nuovo paziente

- utilizzare per ogni campo un’etichetta chiara ed esplicativa;

Operabilità

- assicurarsi che su tutti i campi e sui bottoni sia possibile operare con facilità. È importante considerare lo spazio occupato dalle tendine dei menu di selezione, identificare in modo univoco i campi non selezionabili e la posizione dei bottoni cancella/annulla.

Messaggi e recupero dell’errore

- I messaggi d’errore devono essere presentati all’interno della pagina, e devono essere segnati nel punto in cui l’errore si verifica;
- I messaggi devono essere presentati in un linguaggio chiaro e comprensibile e spiegare cosa l’utente deve fare per risolvere il suo problema;
- Far sì che il sistema possa tollerare l’inserimento dei dati in formati differenti oppure rendere evidente il formato unico con cui l’utente deve inserire i dati. Il problema si presenta per lo più quando utilizziamo dati di tipo numerico, come date e codici. Per far sì che il personale medico inserisca correttamente la data di nascita oppure l’altezza (espressa in centimetri) del paziente sono stati posizionati al di sotto del campo da completare degli avvisi che indirizzano la compilazione e la formattazione corretta.

Figura 2.15: Form di esempio - Messaggi di errori.

```

1 <div class="form-group required">
2   <div class="dati-paziente col-sm-5 col-sm-offset-1">
3     <label for="datanascita" class="control-label">Data di nascita:</label
4     >
5     <input type="text" name="datanascita" class="form-control" placeholder
6     = "Inserisci data di nascita nel formato gg-mm-aaaa" ng-model="data.
7     nuovopaziente.datanascita" required ng-pattern="regexNascita">

```

```

5   <div ng-messages="addPazienteForm.datanascita.$error" class="form-
    error" ng-if="!addPazienteForm.datanascita.$pristine">
6     <p ng-message="required">Campo obbligatorio!</p>
7     <p ng-message="pattern">La data deve avere il formato gg-mm-aaaa!</p>
    >
8   </div>
9 </div>
10 </div>

```

Listing 2.9: Messaggi d'errore - HTML

Complessità della procedura

- Richiedere sempre solo le informazioni strettamente necessarie. Minore è il numero di campi che è necessario compilare e più efficace e solida sarà l'interazione con l'utente. Questa affermazione è tanto più vera tanto più è complesso il form;
- Raggruppare in aree tematiche le informazioni che devono essere inserite. Per ridurre la complessità o quanto meno il rumore visivo percepito dall'utente possono essere utilizzati vari accorgimenti, spaziature, allineamenti o cornici (fieldset) e raggruppamenti. Riunire in gruppi tematici i campi è utile all'utente ed è considerato un ottimo espediente di organizzazione visiva;
- Chiedere subito le informazioni fondamentali (Esempio: nome - cognome - data di nascita)

Percepibilità

- Utilizzare sfondi bianchi o neutri. Uso di questi colori aumenta la leggibilità dei contenuti.
- Identificare una dimensione del font corretta a garantire un'ottima leggibilità. Come per tutti i testi la soglia di leggibilità è posta a circa 10-12 pixel. Al di sotto tale dimensione non è più garantita la leggibilità senza problemi da tutti i possibili utenti.
- Ridurre al minimo il rumore visivo eliminando elementi di disturbo: colori brillanti, elementi estetici inutili o pesanti.
- Allineare le etichette e i campi.
- Attribuire ad ogni campo la giusta lunghezza in funzione proprio del tipo di informazione che deve contenere. Se sono presenti molti campi e volendo allinearli valutare una lunghezza media corretta.
- Far sì che tutti i campi abbiano la medesima larghezza e la loro disposizione sia omogenea per quanto possibile. Scegliere 2-3 dimensioni standard da replicare ed utilizzare per tutti i form.
- I campi obbligatori devono essere identificati in modo chiaro ed univoco. L'indicazione deve essere esplicitata sempre per ogni campo, a meno che tutti i campi non siano obbligatori o opzionali. In tal caso è necessario indicare quest'informazione all'inizio

del form. Il metodo classico e standard per far riconoscere un campo obbligatorio è l'asterisco.

Per garantire un'ottima percezione dei form ai soggetti fruitori si è scelto di applicare a tutti uno stile comune, estremamente essenziale e pulito. Tutti i campi risultano essere allineati e dimensionati in funzione del contenuto del campo che devono ospitare. A sostituzione di menù a tendina si è preferito utilizzare l'autocompletamento; in questo modo il sistema presenta al fruitore una serie di possibili risposte selezionabili che vengono filtrate dinamicamente in funzione delle lettere progressivamente inserite all'interno del campo di input.

The image shows a medical form with several input fields and a modal dialog. The form fields include:

- Data:** 22-12-2017
- Altezza:** 190 cm
- Peso:** 78 Kg
- BMI:** 21,61
- Superficie Corporea:** 2,03 m²
- Pressione Sistolica:** 101 mmHg
- Pressione Diastolica:** 94 mmHg
- Pressione Media:** 96
- Saturazione Ossigeno:** 100 %
- Frequenza Cardiaca:** 101 bpm

Buttons: SALVA (red), Annulla (grey).

Modal Dialog: AGGIUNGI ARITMIA

- Aritmia:** Inserisci aritmia (with a dropdown menu and the text "Scegli un'opzione dalla lista")
- Data:** (with a date picker icon)

Buttons: OK (red), Annulla (grey).

Figura 2.16: Form di esempio - Perceibilità dei form.

```

1
2 <div class="form-group required" ng-if="ruoliUtente.struttura">
3   <div class="dati-paziente col-sm-5 col-sm-offset-1">
4     <label for="nome" class="control-label">Nome:</label>
5     <input type="text"
6       name="nome"
7       class="form-control"

```

```

8     placeholder="Inserisci nome {{data.subject}}"
9     ng-model="data.pazienteMod.nome" required>
10    <div ng-messages="editPazienteForm.nome.$error"
11        class="form-error" ng-if="!editPazienteForm.nome.$pristine">
12        <p ng-message="required">Campo obbligatorio!</p>
13    </div>
14 </div>
15
16 <div class="dati-paziente col-sm-5">
17     <label for="cognome" class="control-label">Cognome:</label>
18     <input type="text" name="cognome"
19         class="form-control"
20         placeholder="Inserisci cognome {{data.subject}}"
21         ng-model="data.pazienteMod.cognome" required>
22     <div ng-messages="editPazienteForm.cognome.$error"
23         class="form-error" ng-if="!editPazienteForm.cognome.$pristine">
24         <p ng-message="required">Campo obbligatorio!</p>
25     </div>
26 </div>
27 <div class="col-sm-10 col-sm-offset-1">
28     <div>
29         <md-radio-group layout="row" layout-align="start center"
30             md-no-ink ng-model="data.pazienteMod.sesso">
31             <label for="sesso" class="control-label">Sesso:</label>
32             <md-radio-button value="M" aria-label="Maschio">
33                 Maschio</md-radio-button>
34             <md-radio-button value="F" aria-label="Femmina">
35                 Femmina</md-radio-button>
36         </md-radio-group>
37     </div>
38     <div ng-messages="editPazienteForm.sesso.$error" class="form-error"
39         ng-if="!editPazienteForm.sesso.$pristine">
40         <p ng-message="required">Campo obbligatorio!</p>
41     </div>
42 </div>
43 </div>

```

Listing 2.10: Form di esempio - HTML - estratto di codice

2.2.13 Feedback per l’utente

Concludo la panoramica sui principi dell’usabilità trattando un argomento estremamente importante e che non può essere sottovalutato: i feedback per l’utente. È necessario infatti fornire ad esso tutti i feedback visivi adeguati in funzione delle azioni portate a termine e nel caso di applicazioni complesse, come può essere GUCH, la risposta del sistema è necessaria per valutare il risultato di azioni importanti e che possono avere effetti sui dati presenti all’interno del database. Nel mondo del Web, dove ci si muove fra pagine, fornire risposte di ritorno significa nella maggior parte dei casi dare degli strumenti di orientamento che equivalgono a indicazioni di stato e di posizione. La definizione di feedback corretti e mirati nella comunicazione è fondamentale per portare a termine in modo proficuo una procedura e far mantenere l’attenzione all’utente.

Il ruolo del feedback è utile anche nella costruzione del rapporto di fiducia con l'applicazione. Essa non è esclusivamente collegata alla funzionalità pratica della procedura, ma viene nutrita proprio dalle continue risposte date dal sistema; più il compito è complesso e più i messaggi di conferma per l'utente acquistano importanza e lo confortano sulle sue azioni e decisioni. Una risposta di ritorno si trasforma in questo caso in uno strumento di costruzione della fiducia e contribuisce alla creazione di un'esperienza positiva per l'utente.

All'interno del progetto GUCH sono stati utilizzati molti messaggi di feedback, presentati attraverso modali, che guidano l'utente all'interno delle azioni che può svolgere. È importante sottolineare come l'usabilità non si componi solo dell'efficacia o dalla facilità d'uso degli elementi dell'applicazione, ma anche da tanti altri strumenti che vanno a costituire l'esperienza complessiva. La relazione che si costruisce con l'utente deve essere posta, come importanza, al pari dell'efficacia.

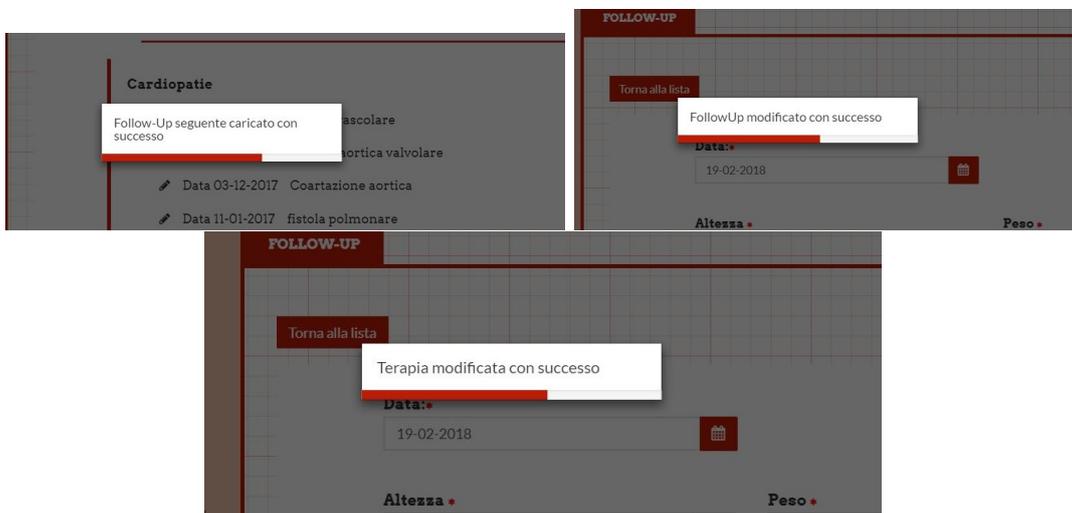


Figura 2.17: Feedback per l'utente.

```

1 $scope.addImaging = function () {
2   var oggPerPost = {};
3   oggPerPost.idfollowup = $scope.idFollowUp;
4   oggPerPost.idimaging = $scope.oggnNewElement.imaging.id;
5   oggPerPost.idesame = $scope.oggnNewElement.esame;
6   oggPerPost.value01 = $scope.oggnNewElement.imaging.value01;
7   oggPerPost.value02 = $scope.oggnNewElement.imaging.value02;
8   oggPerPost.value03 = $scope.oggnNewElement.imaging.value03;
9   oggPerPost.value04 = $scope.oggnNewElement.imaging.value04;
10  oggPerPost.value05 = $scope.oggnNewElement.imaging.value05;
11  oggPerPost.value06 = $scope.oggnNewElement.imaging.value06;
12  oggPerPost.value07 = $scope.oggnNewElement.imaging.value07;
13  oggPerPost.value08 = $scope.oggnNewElement.imaging.value08;
14  oggPerPost.value09 = $scope.oggnNewElement.imaging.value09;
15  oggPerPost.value10 = $scope.oggnNewElement.imaging.value10;
16  oggPerPost.nota = $scope.oggnNewElement.nota;
17  console.log("ogg per post");
18  console.log(oggPerPost);

```

```

19 $scope.oggNewElement.datarecord.setHours(12, 0, 0, 0);
20 oggPerPost.datarecord = $scope.oggNewElement.datarecord.toISOString().
    slice(0, 10);
21
22
23
24 restFollowUp.addImaging(oggPerPost)
25 .then(
26     function(response) {
27         if (response != null) {
28
29             restFollowUp.getFollowUp($scope.idPaziente)
30                 .then(function(response) {
31                     if (response != null) {
32                         if (response.length != 0) {
33                             self.listaFollowUp = $filter('orderBy')(response, '-
datafollowup');
34                             self.nDiFollowUp = self.listaFollowUp.length;
35                             $scope.ultimoFollowUp = self.listaFollowUp[0];
36                             $scope.oggNewFollowUp = angular.copy($scope.
ultimoFollowUp);
37                             $scope.oggNewFollowUp.datafollowup = self.oggi;
38                             $scope.oggNewFollowUp.datafollowup.setHours(12, 0, 0, 0)
;
39                             angular.forEach(self.listaFollowUp, function(obj, index)
{
40                                 if (obj.id == $scope.idFollowUp) {
41                                     $scope.oggShowFollowUp = obj;
42                                     console.log("ogg Show followup");
43                                     console.log($scope.oggShowFollowUp);
44                                     $scope.oggEditFollowUp = obj;
45                                     $scope.oggEditFollowUp.datafollowup = new Date(
$scope.oggEditFollowUp.datafollowup);
46                                     $scope.oggEditFollowUp.datafollowup.setHours(12, 0,
0, 0);
47                                 }
48                             });
49                         } else {
50                             $scope.oggNewFollowUp.datafollowup = self.oggi;
51                             $scope.oggNewFollowUp.datafollowup.setHours(12, 0, 0, 0)
;
52                         }
53                     }
54                     const modalData = {
55                         confirmMessage: "Nuovo imaging aggiunto con successo"
56                     };
57                     const modalFunctions = {};
58                     cgmModalHandler.openModal('confirm', modalData,
modalFunctions, 'sm');
59                     preloaderManager.removePreloader();
60                 },
61                 function(response) {
62                     console.log(response);
63                     preloaderManager.removePreloader();
64                 });

```

```
65 }  
66 });  
67 }
```

Listing 2.11: Messaggi di conferma - Javascript - Estratto di codice

2.3 L'usabilità, valutare i problemi

Nella progettazione di un'applicazione è comune imbattersi in problemi di usabilità che dovranno essere risolti talvolta giungendo a necessari compromessi con le necessità e le funzionalità dell'applicativo stesso. È indispensabile quindi procedere attribuendo ad ogni problematica che si incontra una priorità e cercare di risolvere per primo i problemi che appaiono più sfavorevoli; possono essere considerati problemi gravi tutti quelli che causano all'utente un senso crescente di frustrazione e confusione. L'importanza di assegnare ad ogni problema una valutazione di criticità ci aiuta anche a "calcolare" l'effettivo vantaggio che si potrebbe ricavare dal risolvere la problematica in funzione dello sforzo che però è necessario investire. In determinate situazioni è più conveniente rimandare la ricerca della soluzione, ed approfondire la risposta del fruitore durante l'utilizzo dell'applicazione, per poi in un secondo momento valutare una più efficace azione. La criticità di un problema è determinata principalmente da tre elementi:

- frequenza;
- impatto;
- continuità.

Dagli studi di usabilità condotti da Steve Krug¹⁶, emerge un efficace paragone; ogni utente che utilizza un'applicazione Web ha un proprio serbatoio di fiducia a cui attinge nell'utilizzo dell'applicativo. Incontrare un problema comporta un intaccamento del livello di fiducia, causandone quindi un abbassamento. Causa un abbassamento della fiducia:

- nascondere alla vista delle informazioni importanti;
- esagerare con le regole da seguire nel completamento di un form;
- chiedere troppe informazioni, anche quelle non necessarie;
- ostacolare la navigazione;
- non curare l'aspetto estetico dell'applicativo.

Al contrario, fa aumentare il livello di fiducia nell'utente:

- mettere in luce le funzionalità principali dell'applicativo e renderle riconoscibili e facilmente raggiungibili;

¹⁶Steve Krug autore di *Don't make me think*, è un'esperto e consulente di usabilità. Nella sua ventennale esperienza è stato consulente per una grande quantità di clienti fra questi citiamo: Apple, Lexus e Bloomberg. [4]

- ridurre al minimo il numero di click necessari a raggiungere le informazioni importanti e significative per il soggetto fruitore;
- rendere semplice la correzione di errori.

Jakob Nielsen studiando i problemi di usabilità e raggruppandoli in categorie ha ottenuto un esplicativo diagramma a torta così ripartito che illustra quali siano quindi i problemi principali che inducono nell'utente un maggior tasso di difficoltà. Nel diagramma vengono relazionati i diversi problemi di usabilità rapportandoli alla frequenza con cui ostacolano l'utente nel raggiungimento del proprio obiettivo.

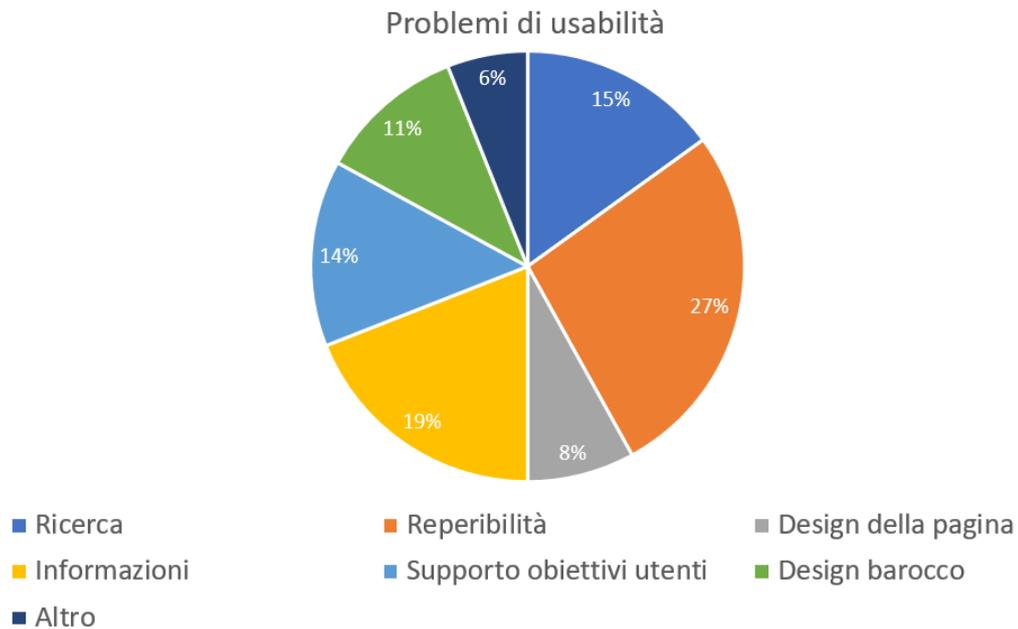


Figura 2.18: Problemi di usabilità - Diagramma a torta.

Osservando il grafico si evince come la ricerca sia un problema non trascurabile, così come la reperibilità degli elementi dell'interfaccia (menù, link, bottoni) e si deduce che la principale fonte di frustrazione per un utente sia proprio l'impossibilità di raggiungere in modo semplice e soprattutto veloce la sezione desiderata. Le informazioni incomprensibili o incomplete rappresentano un'altra ampia fetta nel diagramma andando ad intaccare il livello di fiducia del soggetto. Da non trascurare il design che influisce con una percentuale rilevante; importante tenere infatti in considerazione il discorso precedentemente fatto sull'organizzazione spaziale delle informazioni e sulla suddivisione in aree tematiche.

Capitolo 3

Le tecnologie utilizzate

Dopo aver fatto una panoramica sugli essenziali principi di usabilità che sono stati studiati e applicati nella fase di progettazione, analizziamo le principali tecnologie che sono state utilizzate in fase di sviluppo front-end ¹.

Lo scopo di questa tesi non è quello di soffermarsi sul codice che è stato scritto in questi mesi, ma questa sezione ha l'obiettivo di dare un piccolo sguardo complessivo ai principali linguaggi utilizzati. Lo studio dei linguaggi che verranno presentati di seguito è stato ampiamente approfondito durante il lavoro in azienda e le conoscenze pratiche apprese sono difficilmente riassumibili e riportabili all'interno di questo elaborato.

3.0.1 HTML

HTML ² è un acronimo che sta per: *HyperText Markup Language* che in italiano può essere tradotto in Linguaggio di Contrassegno Ipertestuale. Può essere definito utilizzando la dicitura: linguaggio a markup, in quanto a differenza di un linguaggio di programmazione tradizionale permette di indicare come posizionare i vari elementi all'interno di una specifica pagina. Le indicazioni vengono fornite scegliendo quali marcatori, chiamati più comunemente tag (etichette), utilizzare. I differenti tag sono facilmente riconoscibili all'interno di una pagina HTML in quanto sono posizionati fra parentesi angolari (esempio di dichiarazione di tag: <nome del tag></nome del tag>). I vari elementi definiti e posizionati utilizzando proprio le varie etichette appariranno sullo schermo all'interno di pagine HTML.

È importante sottolineare come un codice HTML non possieda i costrutti tipici della programmazione quali: cicli, meccanismi condizionali o iterativi. Le strutture condizionali

¹Con il termine front-end si intendono tutte le attività di sviluppo di un' applicativo che consentono di rendere visibili informazioni e dati agli utenti. È pertanto responsabile dell'acquisizione dei dati in ingresso e della loro elaborazione al fine di renderli utilizzabili dal back-end.

² È stato sviluppato all'inizio degli anni '90 da Tim Berners-Lee al CERN di Ginevra insieme al protocollo HTTP utilizzato per il trasferimento di tutti i documenti nel formato, per l'appunto html. Lo sviluppo di tale linguaggio si inserisce all'interno di un progetto di più ampio respiro denominato e conosciuto con il nome "world wide Web". Proprio all'interno di questo progetto nasce il primo browser della storia che diventa operativo, a tutti gli effetti, nel 1991.

permettono di definire differenti comportamenti che vengono determinati al verificarsi della specifica condizione; si parla pertanto in questo caso di linguaggio dichiarativo. Dal lontano 1991, anno in cui Tim Berners Lee ne ha definito e pubblicato la prima versione, il linguaggio ha continuato ad evolversi arrivando allo standard HTML 5.

3.0.2 CSS e LESS

Anche in questo caso CSS non è altro che un acronimico il cui significato è: *Cascading Style Sheets* ovvero in italiano fogli di stile a cascata. È un linguaggio di programmazione di tipo dichiarativo (proprio come il linguaggio HTML) con cui è possibile definire lo stile di un documento Web. Attraverso il CSS si definiscono ad esempio: font, colori, immagini di sfondo, posizionamento ed allineamento specifico e tutti gli aspetti grafici dei diversi elementi presenti all'interno della pagina. Per arricchire l'aspetto estetico di una qualsiasi porzione della pagina Web è necessario utilizzare questo strumento; la prima versione è stata resa pubblica nel 1996.

All'interno del progetto GUCH però è stato utilizzato Less ³ il quale può essere definito come un pre-processore del linguaggio CSS che permette la sua estensione facendo sì che sia possibile utilizzare, oltre alla normale sintassi CSS, anche funzioni, operazioni e variabili, nidificazioni e istruzioni. Esso si basa sulla normale sintassi CSS, ma le funzionalità aggiuntive messe a disposizione da questo tipo di linguaggio fanno sì che sia possibile scrivere un codice più compatto, strutturato ed elegante. I principali vantaggi nell'utilizzo di Less sono rappresentati dall'eventualità di utilizzare i mixin e la nidificazione. Quest'ultima fa sì che sia possibile scrivere istruzioni nidificate rendendo il codice più compatto e leggibile; la struttura della nidificazione richiama la stessa organizzazione del listato HTML. I mixin invece, possono essere definiti come dei set di istruzioni che possono essere chiamati nuovamente all'interno della definizione delle proprietà di un qualsiasi selettore.

```
1 .bordered {
2   border-top: dotted 1px black;
3   border-bottom: solid 2px black;
4 }
5
6 #menu a {
7   color: #111;
8   .bordered;
9 }
10
11 .post a {
12   color: red;
13   .bordered;
14 }
```

Listing 3.1: Esempio utilizzo mixin - Less - Estratto di codice documentazione ufficiale

³Per maggiori informazioni sulla sintassi si consiglia la lettura della documentazione ufficiale - <http://lesscss.org/> [9]

3.0.3 Javascript

Nel 1992, viene rilasciato il primo browser Web che permetteva di visualizzare la grafica oltre a porzioni di testo.

Nel 1994, viene fondata la Netscape Communications Corporation ⁴ dando vita ad un browser di qualità superiore. Inizia così lo sviluppo del linguaggio Javascript, il quale apportò una serie di miglioramenti e vantaggi; molti accorgimenti estetici divennero più facili da realizzare e la dinamicità della pagina non fu più limitata.

Principale vantaggio del linguaggio Javascript è quello di essere in grado di aggiungere alle pagine HTML la possibilità di essere modificate in modo dinamico, in funzione dell'interazione dell'utente con il browser utilizzando potenzialità di calcolo e la possibilità di manipolare i documenti senza coinvolgere il server.

Javascript è definito come un linguaggio di scripting integrato, ovvero all'interno di un altro programma; il browser quindi legge una riga, la interpreta e la esegue; al termine passa all'istruzione successiva fino ad arrivare alla chiusura dello script. Uno dei principali utilizzi di Javascript è legato all'ambito Web e alla scrittura di piccole funzioni da integrare nelle pagine HTML con lo scopo di interagire con il DOM ⁵ del browser e per poter compiere determinate azioni non eseguibili con il solo HTML statico. Esempi di manipolazione del DOM possono essere: controllare i valori nei campi di input, nascondere alla vista o visualizzare specifici elementi e la sua sintassi deriva principalmente da altri linguaggi quali Java e C. L'enorme diffusione di questo linguaggio è certamente da legare al diffondersi delle molte librerie nate con lo scopo di semplificare lo sviluppo e la programmazione e dei diversi framework che lo utilizzano come linguaggio principale.

3.0.4 AngularJS

AngularJS ⁶ è un framework Javascript utilizzabile ed essenziale per lo sviluppo di tutte quelle applicazioni che possono essere definite *Single Page Application*. Con il termine framework si intende, nell'ambito della programmazione Web, un'architettura di supporto (che viene sviluppata spesso attraverso un'implementazione logica particolare ed utilizzando pattern) allo scopo di realizzare applicazioni Web dinamiche. L'obiettivo con il quale si utilizza un framework è quello di alleviare lo sviluppo di codice, fornendo allo sviluppatore librerie (funzioni e/o strutture dati) già predefinite che consentono di risparmiare la scrittura di codice già qualificato per quello specifico compito.

Per Single Page Application facciamo invece riferimento ad un'applicazione che viene eseguita all'interno di una singola pagina HTML facendo in modo che tutti gli elementi utili vengono caricati dinamicamente ed aggiunti al DOM della pagina corrente. Volendo utilizzare altre parole si può affermare che nel contesto di una SPA (acronimo di Single Page

⁴Importante azienda statunitense di servizi informatici.

⁵Document Object Model, modello ad oggetti del documento. È una rappresentazione dei documenti strutturati come modello orientato agli oggetti. Esempio di struttura ad albero di tipo DOM è quello che viene generato da un browser Web durante l'interpretazione di un documento HTML. Permette di accedere e aggiornare dinamicamente il contenuto, la struttura e lo stile dei documenti.

⁶Per ulteriori approfondimenti si consiglia la consultazione della documentazione ufficiale disponibile al seguente link - <https://docs.angularjs.org/guide/introduction> [7, 8]

Application) quello che cambia è la vista che viene presentata all'utente e che si modifica proprio all'interazione del fruitore con la pagina stessa. La transizione da una vista alla successiva illude l'utente, che crede di navigare in pagine differenti in quanto ognuna di esse dispone di un proprio indirizzo URL, anche se in realtà il tutto avviene in un'unica pagina HTML. Il meccanismo attraverso cui si mappano gli URL sulle differenti pagine viene chiamato routing ed è implementato dal servizio *route*.

AngularJS si ispira al pattern MVC (*Model, View, Controller* in italiano modello-vista-controllo). È un paradigma di tipo architetturale utilizzato da molti linguaggi di programmazione che permette di rendere la struttura del codice al tempo stesso semplice e flessibile in quanto prevede una distinzione logica della struttura del codice. Il pattern, separa i compiti fra componenti in questo modo:

- model (modello): si occupa di fornire i metodi per accedere ai dati utilizzabili;
- view (vista): visualizza i dati contenuti nel model e si occupa dell'interazione con utenti e agenti;
- controller (controllore): riceve i comandi dell'utente (proprio attraverso la vista) e li attua modificando lo stato degli altri due componenti.

Le specifiche interazioni fra i tre elementi dipendono fortemente dalle relative tecnologie utilizzate e proprio dal tipo di applicazione che si sta sviluppando. Molto si potrebbe dire riguardo l'utilizzo e le specifiche tecniche di questo framework, ma sarebbe davvero complesso riportarle in modo completo ed esaustivo all'interno di questo elaborato. Due sicuramente sono i concetti fondamentali che è bene conoscere di questo framework.

- il data-binding bidirezionale. È un processo che permette la sincronizzazione degli elementi presenti nella User Interface con un componente della logica applicativa. Il processo in AngularJS è bidirezionale, ovvero la modifica può essere effettuata sia dalla logica applicativa ed essere poi sincronizzato nella vista oppure al contrario può essere modificata la vista e il cambiamento riflettersi sulla logica applicativa;
- le direttive. Permettono ad AngularJS di lavorare anche attraverso un approccio dichiarativo, ricordano i tag HTML, ma sono delle effettive entità funzionali e sono facilmente riconoscibili in quanto iniziano con il prefisso *ng-*. L'estremo vantaggio che si palesa nell'utilizzare le direttive è che possono essere definite direttive specifiche dallo stesso sviluppatore. Una volta creata una direttiva essa è riutilizzabile in qualsiasi momento all'interno dell'applicazione.

Capitolo 4

La navigazione

4.1 Introduzione

Navigare un'applicativo per il Web, equivale in larga misura a cercare un prodotto all'interno di un grande magazzino o in un supermercato. Per farci guidare nella ricerca del prodotto di nostro interesse ci affidiamo a degli indizi (cartelli dei prodotti) e proviamo a ragionare per valutare quale sia la corsia corretta. Cercare un determinato prodotto all'interno di un supermercato nel mondo reale o cercare di raggiungere un determinato scopo usando un'applicazione Web presenta, sorprendentemente moltissime similitudini; addirittura più di quelle che si possano immaginare. Per convincerci di ciò basta riflettere su quali termini utilizziamo per riferirci all'esperienza di navigazione. Utilizziamo spesso termini come: "caricare", "mostrare", "portare a" ecc... ¹

Il problema principale che limita l'utente del mondo del Web è l'assenza di una serie di indizi, che in qualche modo ne ostacolano l'esperienza; ad esempio l'utente non ha alcun senso delle proporzioni. Egli difficilmente è in grado di spiegare quanto sia vasto e articolato l'applicativo anche dopo averlo utilizzato frequentemente ed inoltre il soggetto non ha, nella sua esperienza di navigazione, modo di avere indicazioni della direzione in cui si sta muovendo e della sua posizione; il fruitore difficilmente è in grado di comprendere a quale altezza della gerarchia si trovi.

Due sono gli obiettivi fondamentali della navigazione aiutare ogni utente a trovare quello che sta cercando e far sì che capisca dove si trovi, ma altre funzioni spesso sottovalutate possono essere rivelare: che cosa contiene l'applicativo e come esso si utilizza.

Con design empatico, ma soprattutto attento alle esigenze dei futuri fruitori, tutto sarà a portata di click e sarà possibile gestirlo in ogni momento; ogni elemento della navigazione e le relazioni fra le distinte parti del sito saranno chiare. Una navigazione ben progettata e studiata sugli utenti consente di raggiungere ogni obiettivo senza intoppi dando al soggetto che ne fa uso fiducia e sicurezza.

Dagli studi e dai test condotti dal Nielsen Norman Group emerge come gli applicativi

¹S. Krug *Don't make me think. Un approccio di buon senso all'usabilità Web e mobile.*, Tecniche nuove, Milano, 2016. [4]

e i siti che riescono a indirizzare gli utenti a raggiungere i loro personali obiettivi, sono quelli che sono riusciti ad adeguarsi proprio alle loro aspettative. Gli sviluppatori che si impegnano a studiare la navigazione dell'applicazione fan sì che tutti gli utenti riescano a trovare le risposte proprio dove le stanno cercando. A nessun utente piace dover ragionare su dove poter trovare una risposta, né tanto meno fare difficili ragionamenti, si aspettano al contrario di trovare le informazioni di interesse strutturate secondo criteri che loro stessi trovano ragionevoli.

È evidente che strutturare la navigazione di un'applicazione non è un'operazione facile e che non deve essere trascurata; purtroppo non esistono regole scritte da seguire pedestramente per essere sicuri di raggiungere il risultato perfetto. Ciò che gli utenti cercano e come ragionano cambia sicuramente da applicazione ad applicazione, di conseguenza la perfetta navigazione dipende dagli obiettivi che caratterizzano l'applicativo e che il soggetto deve poter portare a termine. Una regola non scritta potrebbe essere racchiusa nello slogan "una buona navigazione è una navigazione prevedibile"²; l'utente non ha bisogno di memorizzare come fare determinate azioni perchè la struttura dell'applicazione riflette proprio le esigenze e i bisogni degli utenti.

Sicuramente, ancora una volta, ci possono venire in aiuto nella progettazione le convenzioni e la coerenza; realizzare una struttura di navigazione coerente è una caratteristica fondamentale a cui deve tendere lo sviluppatore in fase di progettazione. Jakob Nielsen, per spiegare ai propri lettori che cosa voglia dire progettare una navigazione ottimale utilizza un'efficace confronto; paragona la navigazione di un'applicazione ad una scala, ogni click che esegue l'utente coincide con un passo avanti su di essa, e secondo lui, modificare l'esperienza di navigazione equivale a spostare un gradino nel momento in cui il soggetto stesso sta transitando. Quando il cambiare pagina combacia con un cambio della struttura di navigazione stiamo obbligando l'utente a spostare la sua attenzione dal proprio scopo per comprendere invece come funziona in quell'istante la pagina e dove possono trovarsi le informazioni di interesse.

All'interno di questo capitolo affronteremo l'analisi della struttura del progetto GUCH. Il viaggio alla scoperta della navigazione dell'applicativo partirà proprio dall'analisi e dallo studio della precedente applicazione sviluppata qualche anno prima da un'altra azienda. Si passerà poi ad esaminare la struttura che è stata proposta e sviluppata. In questo modo, al termine della lettura del capitolo, saremo in grado di comprendere la complessità e ampiezza del progetto e individuare quali sono stati i compromessi e le difficoltà affrontate.

4.2 Analisi del precedente progetto

Come ho già anticipato del progetto GUCH, sviluppato in questi mesi, esiste una versione precedente e l'associazione Amici del Cuore, trovandosi in difficoltà nell'utilizzarla ha deciso di investire nuovi fondi al fine di realizzarne una nuova. Il precedente progetto non risulta dunque essere più funzionale per i medici che hanno riscontrato delle difficoltà nel suo

² J. Nielsen, H. Horanger Web Usability 2.0. L'usabilità che conta, Apogeo, 2010. [1]

utilizzo e nel portare a termine i loro obiettivi. ³.

In questa sezione vengono analizzate le principali problematiche e incoerenze riscontrate che hanno causato in chi ne faceva utilizzo alcuni problemi facendo, se possibile un parallelismo rispetto al nuovo progetto sviluppato.

4.2.1 Il login

Analizziamo subito la pagina di accesso (pagina con l’inserimento delle proprie credenziali di accesso) e la pagina successiva presentata all’utente dopo aver concluso con successo la prima procedura. Di seguito le immagini di riferimento.



Figura 4.1: Login dell’applicazione

Possiamo subito osservare come al raggiungimento del link di riferimento dell’applicativo si presenti all’utente una schermata in cui è presente un form di login che richiede delle credenziali di accesso. Non è presente nessuna landing page ⁴ che mostri ad un soggetto fruitore (ad esempio: un nuovo medico che venuto a conoscenza del progetto desidera farvi parte) alcune informazioni utili, scopi del progetto, contatti utili o un rimando al sito ufficiale dell’Associazione promotrice del progetto.

³Dettaglio sugli obiettivi del progetto: Capitolo 1 - Il progetto e i suoi scopi [1.1.2](#)

⁴La landing page è una pagina specificatamente costruita che l’utente raggiunge a seguito di un click, ad esempio su di un link. Lo scopo della pagine è quello di comunicare specifiche informazioni, ritenute importanti per tutti i visitatori oppure indurre l’utente a compiere un’azione (ad esempio il login o la registrazione). La sua principale funzione quindi, è quella di spingere il visitatore a compiere una determinata azione.



Figura 4.2: Home - Prima pagina visualizzata a seguito del login

A seguito del login effettuato con successo, il soggetto fruitore accede alla pagina che viene presentata qui sopra e dove poche sono le informazioni presenti. Analizzando la navigazione, un nuovo utente, che ha fatto il suo primo o secondo accesso all'applicazione, non è molto aiutato nella ricerca delle funzionalità principali; le possibili azioni a disposizione dell'utente sono rese accessibili dalla sequenza di bottoni posizionati nella parte alto dello schermo.

4.2.2 La vista elenco pazienti

Un medico per accedere alla sua vista elenco pazienti, a seguito dell'accesso, deve cliccare sul bottone "Visualizza elenco" e selezionare a sua volta, dal menu a tendina che viene aperto, la relativa voce "Propri pazienti"; il processo non è ottimale. Con elevata probabilità la prima operazione che desidera fare un medico a seguito del login è poter osservare, cercare ed in seguito selezionare la scheda di un proprio paziente e nel caso di un nuovo medico al primo accesso aggiungerne uno nuovo all'elenco, ma con una navigazione così strutturata l'utente è obbligato ad eseguire delle azioni aggiuntive. La vista elenco pazienti si presenta nella seguente forma (per ragioni di privacy i nomi e i cognomi sono stati oscurati alla vista così come altri dati sensibili).

Identificativo paziente	Data di nascita	Sesso	Comune di residenza
1	1980-04-01	M	VERCELLI
2	1975-10-01	M	VERCELLI
3	1977-11-04	M	TRURGO
4	1933-05-06	M	CEVA
5	1946-05-01	M	SARONNO
6	1931-10-04	M	VERCELLI
7	1960-08-01	M	VERCELLI
8	1936-10-07	M	VERCELLI
9	1930-05-01	M	VERCELLI
10	1946-08-01	M	VERCELLI
11	1931-07-06	M	VERCELLI
12	1980-05-01	M	VERCELLI
13	1931-08-01	M	VERCELLI
14	1979-02-01	M	VERCELLI
15	1977-02-01	M	VERCELLI
16	1947-05-01	M	VERCELLI
17	1931-08-01	M	VERCELLI
18	1931-08-01	M	VERCELLI
19	1980-08-01	M	VERCELLI
20	1931-05-01	M	VERCELLI

Data	Ora
26/09/2011	12:26:38
01/02/2013	13:28:29

Figura 4.3: Vista elenco pazienti

Cognome	Nome	Data di nascita	Codice Fiscale	Sesso	Comune di residenza	Utenza
1 VERDI						

Data	Ora
nessun follow-up trovato	

Figura 4.4: Vista elenco pazienti

Nella nuova versione del progetto GUCH la prima pagina su cui arriva il medico a seguito dell'accesso è proprio la vista elenco pazienti registrati ordinati per ordine alfabetico; da qui il medico può cercare un paziente utilizzando l'apposito campo, aggiungerne un nuovo e accedere alla scheda del paziente richiesto. Tutte le azioni possono essere eseguite senza dover cambiare pagina e le istruzioni messe a disposizione nella sezione apposita guidano il medico nel completamento delle diverse procedure.

4.2.3 Inserimento di una nuova utenza

Per l'inserimento di una nuova utenza è necessario cliccare sul bottone in alto a sinistra; una volta cliccato viene aperta una nuova sezione che presenta un form da completare. Tranne per la data che è possibile selezionare, gli altri campi devono essere compilati dal medico senza poter effettuare scelte o essere aiutato da meccanismi di autocompletamento, inoltre tutti i campi sono obbligatori e pertanto devono essere necessariamente inseriti al fine di procedere. Non è possibile creare la sola utenza (in vista magari di un'imminente visita del paziente) in quanto già in questa fase mi sono richieste delle informazioni (pressione, peso ecc...) di carattere medico.

Figura 4.5: Registrazione nuovo paziente

Osservando la modalità con cui è possibile aggiungere un nuovo paziente in questa versione, si è preferito inserire il bottone "Aggiungi paziente" in modo che fosse sempre ben visibile (posizionato da solo, senza altri elementi cliccabili intorno a se). La registrazione di un nuovo paziente non implica l'inserimento di voci relative alla clinica, ma in prima battuta vengono richieste solo informazioni anagrafiche, ritenute fondamentali ed in qualunque momento modificabili. Si è prestata particolare attenzione ai messaggi di errore, all'organizzazione e dimensione dei campi e agli strumenti di autocompletamento per facilitare e velocizzare la compilazione dei diversi campi.

4.2.4 La registrazione di nuove voci

Dopo aver registrato un nuovo paziente inserendo i suoi dati anagrafici e alcune informazioni di carattere medico è possibile passare ad una nuova sezione in cui possono essere registrate altre notizie di carattere medico maggiormente specifiche. Le differenti sezioni a cui è possibile accedere sono rappresentate nella figura sottostante.

Per accedere ad una delle tre voci principali è necessario aprire i menu a tendina e selezionare la voce desiderata.

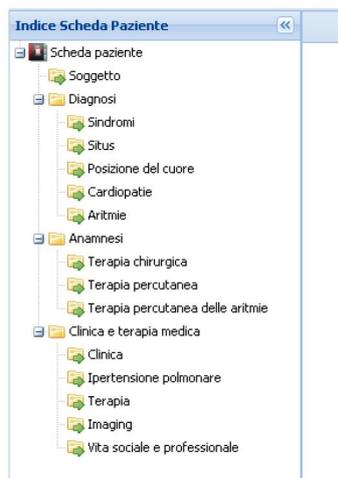


Figura 4.6: Menu della scheda paziente

Una volta selezionata la voce di interesse (ad esempio selezionata la voce sindrome nella cartella diagnosi) si può procedere alla sua compilazione. Quello che appare al medico è un elenco delle voci presenti nel database fra cui è possibile scegliere utilizzando un radio-button; tutti i valori sono di default selezionati su No; il medico deve scorrere l'elenco per ricercare la voce di interesse.

Nel nuovo progetto il medico non è più costretto a scegliere quale voce desidera inserire utilizzando dei radio button, al contrario inserisce la voce di interesse all'interno di un campo di input; l'autocompletamento guida e facilita l'operazione di scelta dell'elemento corretto.

Alcune voci una volta selezionato il radio button d'interesse richiedono l'inserimento della data e nel caso in cui la data non venga inserita il sistema non permette il salvataggio. Questa scelta è stata mantenuta anche nell'attuale versione dove ad ogni voce è associata la data di riferimento, al fine di avere sempre un'indicazione temporale in cui è stata registrata la nuova informazione.

4.2.5 Gestione del follow-up

Per un paziente già presente a sistema è possibile inserire e consultare i dati di un follow-up riferito alle visite successive alla prima. Per inserirne uno nuovo è necessario selezionare il paziente di interesse e nell'apposita sezione dedicata al follow-up cliccare sul bottone Nuovo. I bottoni nuovo follow-up e nuovo paziente sono molto simili, di forma e di colore, sono unicamente posizionati in due punti diversi. La compilazione di un nuovo follow-up è costruita come in precedenza, anche in questo caso, è necessario selezionare la voce di interesse dal menu a tendina.



Figura 4.7: Follow-up del paziente

Questa sezione, fondamentale per un medico per tenere sotto controllo le informazioni di salute dei propri pazienti e aggiornare periodicamente il suo decorso clinico, è stata completamente rivoluzionata; partendo proprio dalla precedente versione e dalle difficoltà riscontrate dai medici si è principalmente puntato a garantire:

- velocità di registrazione di un nuovo follow-up;

- rapida accessibilità ai follow-up precedenti;
- facilità di modifica di eventuali informazioni non corrette o incomplete già inserite;
- chiarezza espositiva dei dati, puntando ad un'impaginazione grafica delle informazioni essenziale e semplice;
- organizzazione visiva delle informazioni.

Per una più esplicativa spiegazione della struttura fare riferimento al paragrafo 'L'entità medico' (4.3.6)

4.2.6 Filtri e ricerca

Una volta visualizzata la griglia dei pazienti è possibile compiere delle operazioni sulle colonne visualizzate con operazioni di ordinamento e introduzione di filtri di ricerca.

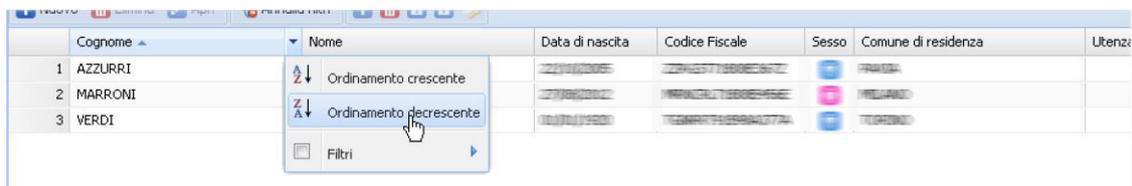


Figura 4.8: Ordinamento voci in tabella

Il meccanismo è complesso e prevede necessariamente che l'utente operi due click; i filtri permettono di ridurre il numero di colonne e di righe visualizzate, mentre attraverso i filtri di ricerca sulla voce Cognome è possibile operare una ricerca, mettendo in evidenza tutti i cognomi che iniziano, ad esempio con una determinata lettera.

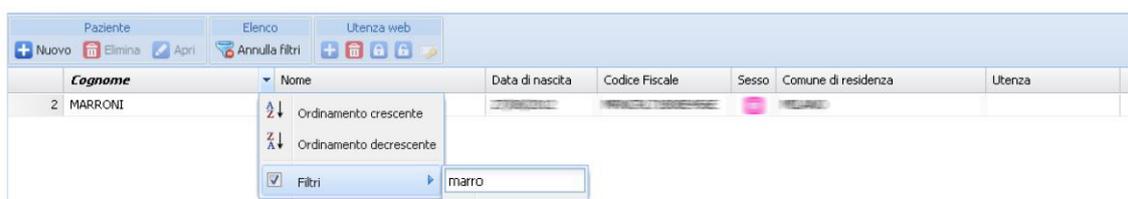


Figura 4.9: Ordinamento voci in tabella

Ancora una volta però l'operazione non è facilmente raggiungibile e richiede che il medico acceda ad un menu a tendina, ne selezioni la voce filtri e successivamente digiti le lettere di interesse. La ricerca, a maggior ragione se il numero di voci all'interno della tabella è alto, è fondamentale ed è importante che sia accessibile in qualunque momento e per queste ragioni nella nuova versione di GUCH il campo di ricerca è presente e ben visibile all'interno di tutte le pagine. Per mantenere la massima coerenza tale strumento è sempre posizionato nel medesimo punto.

Di seguito è riportato un piccolo estratto di codice per analizzare come è stata implementata l'operazione di ricerca; per visionare invece il codice che ha permesso l'integrazione dell'operazione di ordinamento è possibile guardare il listato 2.2 .

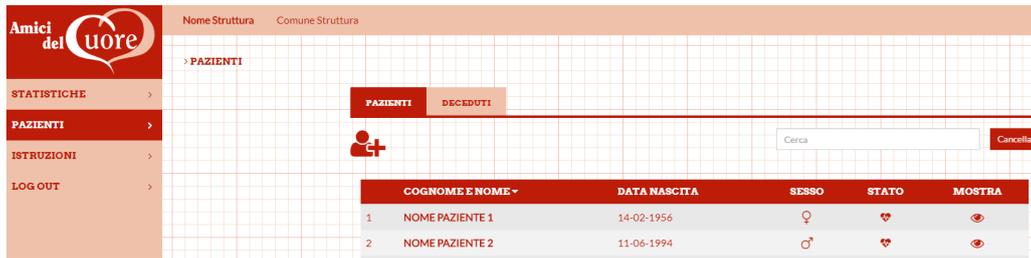


Figura 4.10: Campo ricerca

```

1 <div class="toolbar">
2   <div title="Aggiungi nuovo paziente" class="aggiungi-nuovo">
3     <a ng-click="modalNewPaziente()"><i class="fa fa-user-plus fa-3x" aria-
4       -hidden="true"></i></a>
5   </div>
6   <div>
7     <form class="search-box">
8       <input type="text" placeholder="Cerca" class="form-control" ng-
9         model="searchPaziente" id="search-paziente" size="40">
10      <button type="button" class="btn btn-primary" ng-click="
11        searchPaziente = ''" > Cancella </button>
12    </form>
13  </div>
14 </div>
15 $scope.searchPaziente = "";
16 $scope.getDataPazienti = function() {
17   return $filter('filter')(self.pazienti, $scope.searchPaziente);
18 }
19 $scope.getDataPazientiDeceduti = function() {
20   return $filter('filter')(self.pazienti, $scope.searchPazienteDeceduto);
  }

```

Listing 4.1: Campo ricerca - HTML e Javascript - estratto di codice

4.2.7 Le statistiche

Uno dei principali scopi per cui è stato pensato il progetto GUCH è quello di poter ricavare delle informazioni statistiche al fine di poter portare avanti studi e ricerche mirate fondamentali per aiutare e sostenere i pazienti, ma con il vecchio progetto, questa operazione è diventata progressivamente più complessa. I pochi dati statistici che si potevano estrarre derivavano da un file Excel di ampie dimensioni e dovevano pertanto essere rimaneggiati per poter estrarre le sole informazioni di interesse. Con l'aumentare del numero di pazienti registrati all'interno dell'applicazione il processo di estrazione è diventato progressivamente più complicato; facendo crescere il numero di utenti registrati automaticamente si è visto crescere la dimensione del file, arrivando ad avere un documento di Excel troppo grande (per numero di colonne e di dati) per essere aperto e consultato.

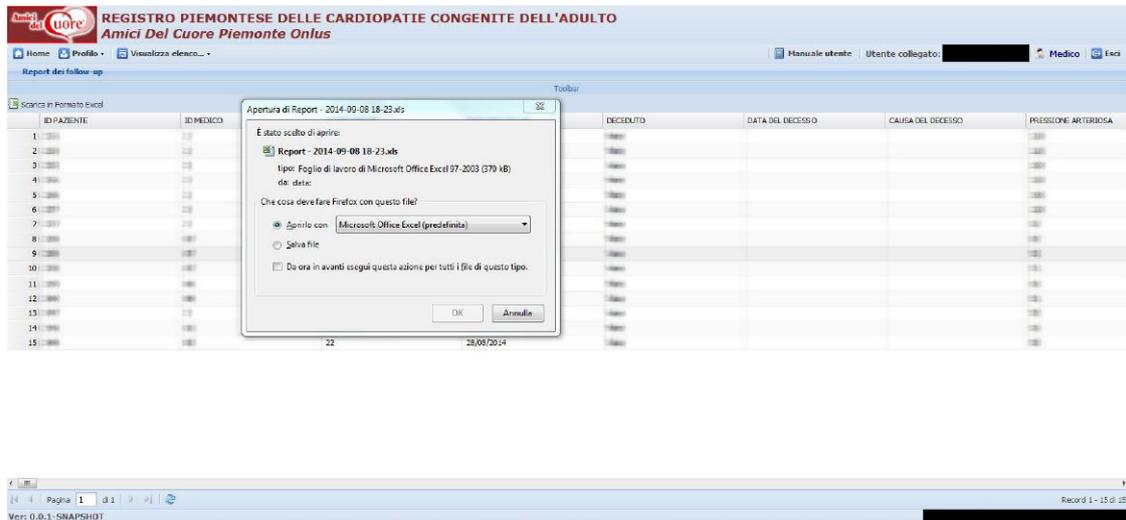


Figura 4.11: Estrazione delle statistiche

Nel nuovo GUCH, le statistiche occupano una sezione rilevante ed è stata loro dedicata un'area specifica all'interno delle tre distinte entità (Nazionale, Regionale, Medico). Esse assumono la medesima struttura in tutti e tre i livelli; quello che si modifica è il numero di dati che vengono considerati per generarle. Per implementare i grafici è stata utilizzata la direttiva Angular Chart derivata dalla libreria Chart.js che permette di realizzare rappresentazioni grafiche differenti, dinamiche e responsive. Per ogni rappresentazione tabellare dei dati viene generato un grafico (circolare o a barre orizzontali), per farsi che sia maggiormente evidente la distribuzione sui pazienti di ogni singola voce. Di seguito viene presentato il codice HTML e Javascript utilizzato per la generazione dei grafici.

```

1 <md-tab id="tabAritmie">
2   <md-tab-label>Aritmie</md-tab-label>
3   <md-tab-body>
4     <div class="table-container">
5       <div class="jumbotron">
6         <div class="title-sezione">
7           <p class="title-sezione-grafo">Aritmie</p>
8           <hr class="riga-divisione">
9         </div>
10
11        <div id="graficoAritmia">
12          <div>
13            <canvas id="bar-1" class="chart-horizontal-bar"
14              chart-data="datasAritmiaCategoria" chart-labels="
15              labelsAritmiaCategoria"
16              chart-series="aritmieXcategoria">
17            </canvas>
18          </div>
19          <div ng-repeat="item in countsXcat" class="row tortaAritmia">
20            <h3>{{ item[2] }}</h3>
21            <hr class="riga-statisticaCat">
22            <div class="col-sm-8" ng-if="item[3] == true">

```

```

22     <canvas id="pie-{{$index}}" class="chart chart-pie" chart-
data="item[0]"
23     chart-labels="item[1]" chart-options="legendaOptions">
24     </canvas>
25     </div>
26
27     <div class="datiTable">
28     <div class="titolo">
29     <ul ng-repeat="titolo in item[1]">
30     <li>{{titolo}}</li>
31     </ul>
32     </div>
33     <div class="count">
34     <ul ng-repeat="count in item[0] track by $index">
35     <li>{{count}}</li>
36     </ul>
37     </div>
38     </div>
39     </div>
40     </div>
41     </div>
42     </div>
43 </md-tab-body>
44 </md-tab>

```

Listing 4.2: Grafico aritmie - HTML - estratto di codice

```

1 restStatistiche.getStatisticaNazionale()
2 .then(
3     function(response) {
4         if (response != null) {
5             self.statisticaNazionale = response;
6             self.statisticaAritmia = response.aritmia;
7             self.statisticaCardiopatia = response.cardiopatia;
8             self.statisticaClinica = response.clinica;
9             self.statisticaDecesso = response.decesso;
10            self.statisticaFarmaco = response.farmaco;
11            self.statisticaSindrome = response.sindrome;
12            self.statisticaTerapie = response.terapia;
13            self.graficiCategorie();
14            self.graficoXCategorie();
15            preloaderManager.removePreloader();
16        }
17    },
18    function(response) {
19        console.log(response);
20        preloaderManager.removePreloader();
21    }
22 );
23
24 //Grafici a barre per categorie
25 self.graficiCategorie = function(){
26     angular.forEach(self.statisticaAritmia, function (obj, index) {
27         $scope.datasAritmiaCategoria.push(obj.count);
28         $scope.labelsAritmiaCategoria.push(obj.titolo);

```

```

29     });
30 }
31
32 //Grafico a torta per ogni specifica categoria
33 self.graficoXCategorie = function() {
34     var aritmieXcat = [];
35     $scope.datiTabella = [];
36     angular.forEach(self.statisticaAritmia, function (obj, index) {
37         aritmieXcat.push(obj.aritmie);
38     });
39     $scope.countsXcat = [];
40     angular.forEach(aritmieXcat, function(obj, index) {
41         $scope.countsXcat[index] = [];
42         $scope.countsXcat[index][0] = [];
43         $scope.countsXcat[index][1] = [];
44         $scope.countsXcat[index][2] = "";
45         $scope.countsXcat[index][3] = false;
46
47         angular.forEach(obj, function(obj2, index2) {
48             $scope.countsXcat[index][0][index2] = obj2.count;
49             $scope.countsXcat[index][1][index2] = obj2.titolo;
50             $scope.countsXcat[index][2] = obj2.categoria;
51             if(obj2.count != 0){
52                 $scope.countsXcat[index][3] = true;
53             }
54         });
55     });
56 }

```

Listing 4.3: Grafico aritmie - Javascript - estratto di codice

Le statistiche si presentano con questa veste grafica, di seguito alcune immagini di riferimento

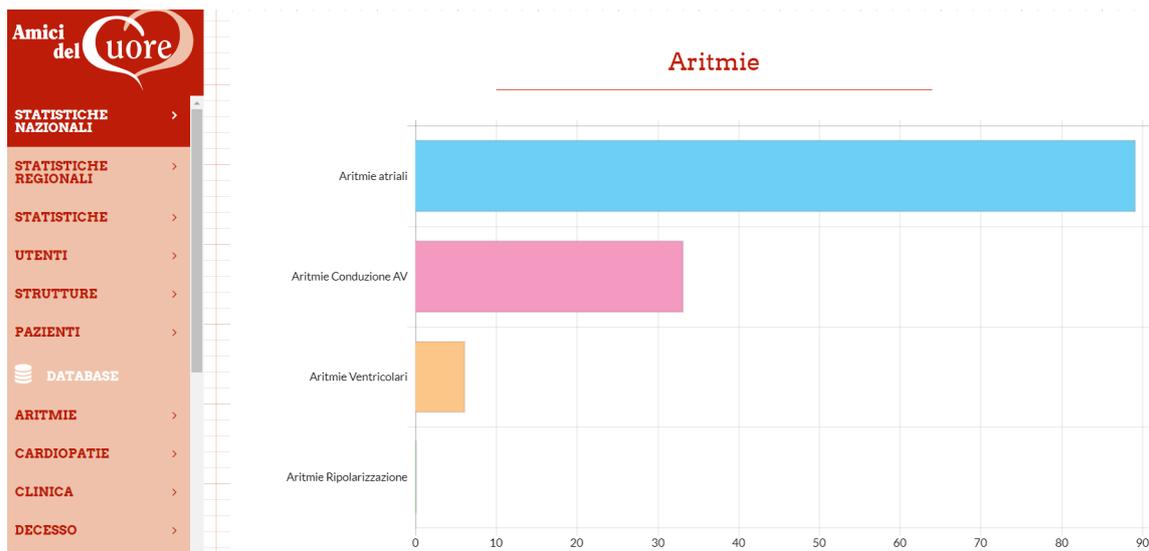


Figura 4.12: Statistiche con grafico a barre - Aritmie

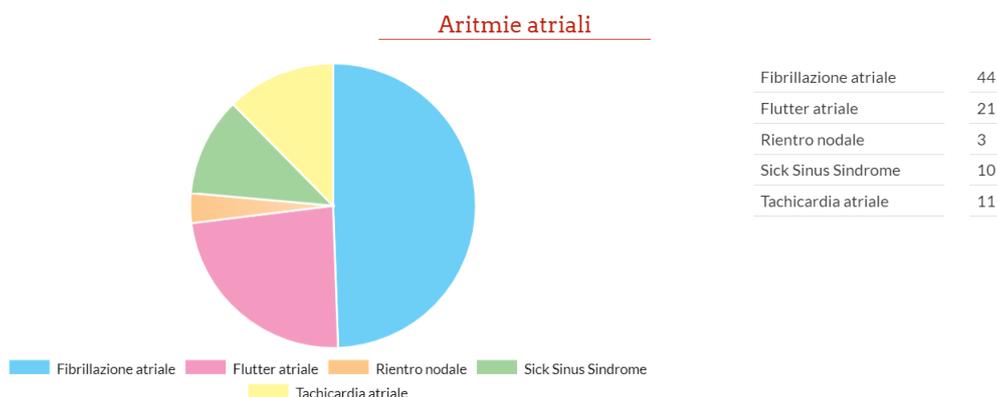


Figura 4.13: Statistiche con grafico a torta - Aritmie

4.2.8 Il manuale

Concludiamo questa piccola analisi valutando il manuale di spiegazione che accompagna il progetto. Vuole essere un aiuto e una guida per tutti gli utenti dell'applicativo che si interfacciano per la prima volta con questo strumento; è ben organizzato nelle diverse parti, ma è molto lungo; il risultato che si rischia di ottenere è quello che nessun medico legga il vademecum e sia quindi in grado di sfruttare il prodotto in tutti i suoi aspetti.

Nella nuova versione di GUCH non è stato redatto un manuale su carta, ma è stata aggiunta una nuova voce al menù laterale denominata "Istruzioni"; qui l'utente può trovare una piccola guida in cui vengono illustrati, utilizzando immagini e poco testo, i passi da eseguire per portare a termine una specifica operazione. In questo modo, nel caso si presentassero dei dubbi, non è necessario andare a consultare un documento esterno.



Figura 4.14: Sezione istruzioni

```

1 <div class="col-lg-8 col-lg-offset-2 col-md-12">
2   <div class="table-responsive">
3     <md-tabs md-dynamic-height md-no-ink>
4       <md-tab id="tab1">
5         <md-tab-label>Log in / Log out</md-tab-label>
6         <md-tab-body>
7           <div class="table-container">
8             <div uib-accordion>
9               <div uib-accordion-group heading="Non ricordo la mia
password">
10                <div class="row">

```

```

11         <div>
12             <div uib-carousel active="logIn.active" template-url="
app/assets/carouseltpl/carousel.html">
13                 <div uib-slide ng-repeat="slide in slidesLogIn track
by slide.id" index="slide.id">
14                     
15                 </div>
16             </div>
17         </div>
18     </div>
19 </div>
20 </div>
21 </div>
22 </md-tab-body>
23 </md-tab>
24 <md-tabs>
25 </div>
26 </div>
27

```

Listing 4.4: Istruzioni - HTML - estratto di codice

```

1  $scope.slidesLogIn = [];
2  $scope.logIn = {};
3  $scope.logIn.active = 0;
4
5  getImages.getImages()
6  .then(
7      function (response) {
8          console.log(response);
9          if (response != null) {
10             $scope.slidesLogIn = response.login;
11         }
12     preloaderManager.removePreloader();
13     },
14     function (response) {
15         console.log(response);
16         preloaderManager.removePreloader();
17     }
18 );

```

Listing 4.5: Istruzioni - Javascript - estratto di codice

4.3 La navigazione in GUCH

Affrontiamo ora la navigazione all'interno di GUCH; scopo di questa sezione è quello di analizzare nel dettaglio le azioni che possono essere portate a termine da ogni entità e le differenti relazioni che intercorrono fra di esse. Per illustrare e facilitare l'analisi della navigazione, ho preparato degli schemi ⁵ che permettono di mettere in luce alcuni meccanismi e funzionalità dell'applicativo.

⁵Tutti gli schemi e le mappe che seguiranno sono state realizzate con Illustrator

4.3.1 Le entità

All'interno del progetto GUCH sono identificabili quattro differenti entità; esse presentano quattro specifici compiti e funzionalità differenti le une dalle altre. È possibile rappresentare l'interazione che intercorre fra i quattro ruoli utilizzando una piramide.



Figura 4.15: Le entità

Come si può vedere dall'immagine, alla base della piramide abbiamo l'entità nazionale che gestisce e amministra il sistema; è la sola entità preposta all'aggiungere voci all'interno del database che saranno poi disponibili a tutte le altre ed eventualmente modificarle. Il ruolo nazionale ha accesso a tutte le statistiche a livello nazionale, filtrate per singole regioni, ma non ha alcun accesso alla clinica del paziente appartenente ad una singola struttura. Successivamente analizzeremo nello specifico i compiti di questa entità entrando maggiormente nel dettaglio.

Il ruolo regionale, al contrario, ha il compito di gestire le strutture appartenenti alla sua regione, ha accesso alle statistiche riferite unicamente al suo territorio e collabora, insieme al livello nazionale, alla gestione del registro abilitando le strutture e i medici che vi possono far accesso. Il ruolo regionale vuole essere un punto di riferimento per tutte le strutture e i medici operanti in una specifica regione, al fine di far arrivare eventuali richieste al livello nazionale. È utile che per ogni regione, l'entità nazionale definisca un referente, ma nel caso in cui tale figura non sia stata identificata, per quella specifica regione il referente sarà la stessa entità nazionale, fino al suo riconoscimento. Anche in questo caso il ruolo regionale non ha alcuna possibilità di accedere alle schede dei singoli pazienti affiliati ad una struttura operante in quella regione.

L'entità medico è quella predisposta alla registrazione del paziente e fa riferimento ad una specifica struttura ospedaliera; i suoi compiti sono quelli di creare, modificare e aggiornare le schede dei pazienti. Il punto di riferimento del medico per eventuali richieste, abilitazioni ad utilizzare l'applicativo è la regione. Ogni medico ha accesso ai soli pazienti registrati che appartengono alla medesima struttura ospedaliera in cui egli opera e può visionare le statistiche che si riferiscono unicamente ai dati di quell'ospedale.

I pazienti sono legati ad una struttura e possono solo interfacciarsi con essa; possono accedere all'applicativo se ne hanno fatto richiesta, possono visionare la loro scheda aggiornata e modificata dal medico, ma non hanno modo di modificare dati clinici. Le specifiche funzionalità dell'entità paziente saranno approfondite di seguito.

4.3.2 Il login

Il login rappresenta il punto di partenza per tutte le entità; per accedere all'applicativo è necessario inserire e-mail e password corrette. In funzione dell'utente che vi accede il sistema automaticamente è in grado di sapere a quale ruolo appartiene, presentandogli quindi il corretto ambiente con le funzionalità che caratterizzano lo specifico ruolo. La fase di login quindi rappresenta l'unico punto in comune per tutte le entità. A seguito dell'accesso due soggetti differenti (esempio paziente e regione) si muoveranno all'interno di un ambiente diverso per le funzionalità messe a disposizione.

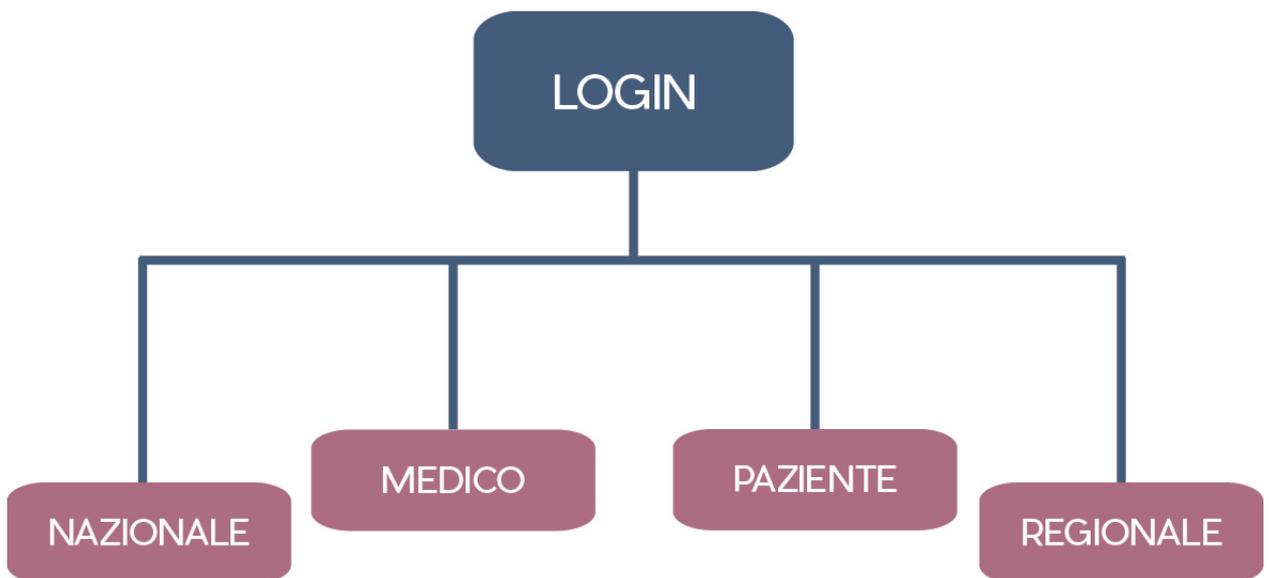


Figura 4.16: Mappa del login

4.3.3 Mappa complessiva dell'applicativo

Di seguito presentiamo la mappa complessiva dell'applicazione che permette di avere a colpo d'occhio un'idea della complessità e dell'ampiezza del progetto. Nell'introduzione di questo capitolo si è analizzato come l'utente non sia in grado di comprendere e il più delle volte non abbia idea delle proporzioni dell'applicativo che sta utilizzando, facendo

pertanto difficoltà a tracciarne i confini e di conseguenza, ad orientarsi all'interno di essa. Per questa ragione, ho preparato una piccola mappa complessiva, che illustri, seppur in modo schematico, le possibili interazioni che sussistono fra le quattro entità analizzate in precedenza ⁶.

⁶Riferimento - 4.1.1 Le entità

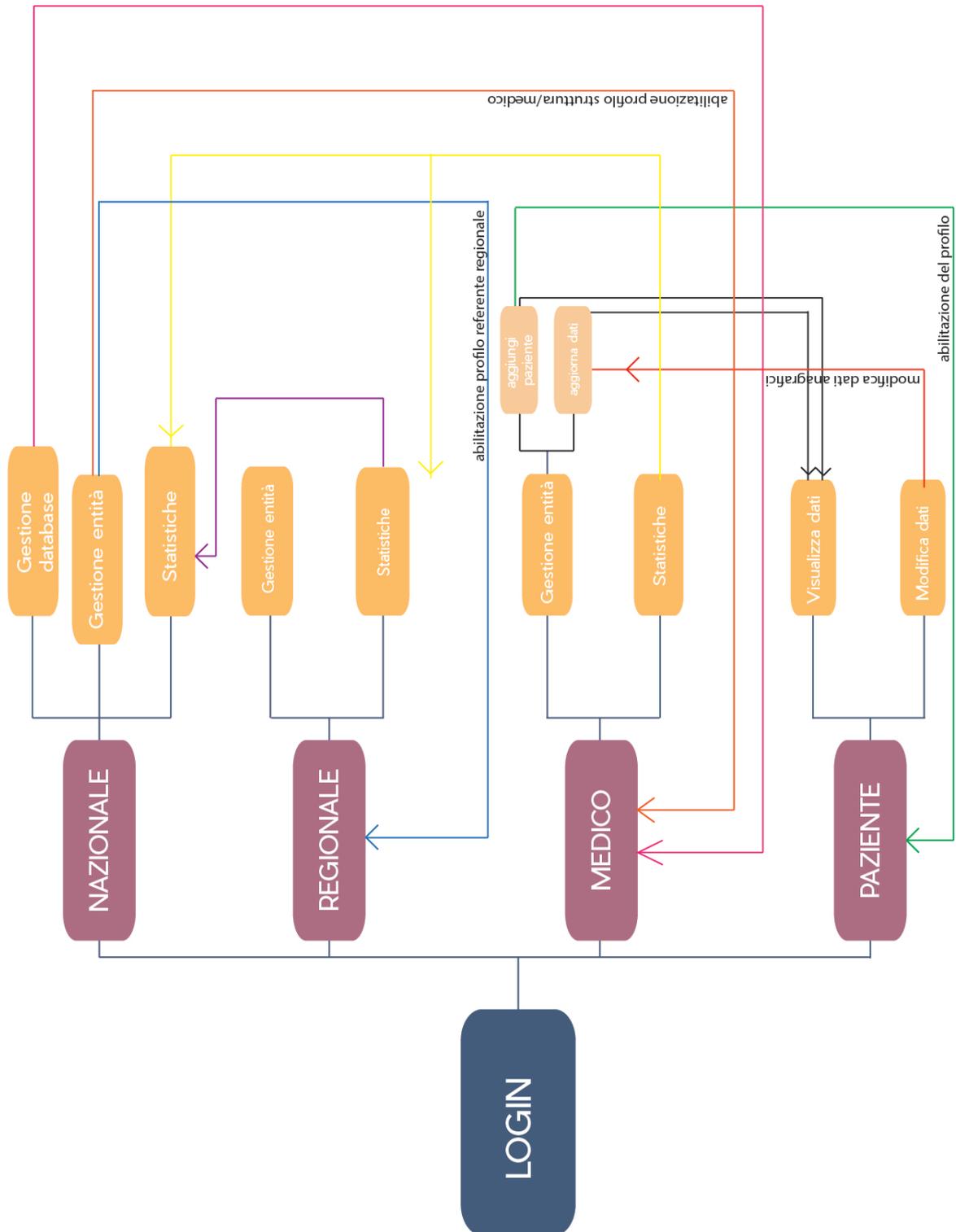


Figura 4.17: Mappa complessiva dell'applicazione

4.3.4 L'entità nazionale

Analizziamo ora nel dettaglio tutte le possibili azioni che può compiere l'entità nazionale. Le principali funzionalità sono:

- statistiche. L'entità nazionale ha la possibilità di accedere alla sezione statistiche organizzate in due sotto sezioni. Ha la possibilità di osservare statistiche di tipo globale (a livello nazionale) o a livello regionale (riferendosi ad una delle regioni iscritte al progetto);
- aggiungere entità. Il ruolo nazionale ha la possibilità di aggiungere e creare nuove entità di tipo struttura o medico; il suo ruolo non permette di creare entità di tipo paziente. Inoltre alla sola entità nazionale spetta il compito di gestire i referenti regionali; essa può assegnare (creandone una nuova), rimuovere e modificarle l'utenza referente regionale.
- visualizzare elenco delle entità. È possibile visualizzare l'elenco di tutte le entità create. Il ruolo nazionale ha la possibilità di osservare in tabella tutte le regioni e i rispettivi referenti, tutte le strutture affiliate al progetto e i medici attivi. In qualunque momento può modificare il referente di una specifica regione;
- gestione del database. Al solo livello nazionale spetta la gestione del database. È la funzione principale e fondamentale di questo ruolo. Il ruolo nazionale ha infatti la possibilità di aggiungere e modificare le voci presenti all'interno del database. Per ogni sezione (aritmie, cardiopatie, clinica, decesso, esami/imaging, farmaci, posizione, sindrome, situs, stili di vita, terapie, titoli di studio) il ruolo nazionale può aggiungere, attraverso una specifica modale una nuova voce in elenco o modificarne una già precedentemente inserita; ogni nuova voce aggiunta, viene resa disponibile a tutti i medici. Facendo un esempio, il livello nazionale può aggiungere alla sezione farmaci un nuovo medicinale di nome "nuovo farmaco prova". Dal momento in cui la voce è stata caricata con successo, tutti i medici potranno registrare tale farmaco ad uno dei pazienti; al medico sarà sufficiente digitare le prime tre lettere del nome della nuova voce per avere il farmaco denominato "nuovo farmaco prova" fra i suggerimenti. Le voci delle differenti sezioni sono organizzate in tabelle; alcune di esse sono a loro volta suddivise in categorie e per ogni specifica classe è possibile visualizzare le voci che sono ad essa accomunate.

Un campo input, preposto alla ricerca facilita l'operazione ricerca di una voce, nel caso sia necessario controllarne la presenza per evitare l'inserimento di una doppia voce o dover accedere ad un elemento per modificarlo.

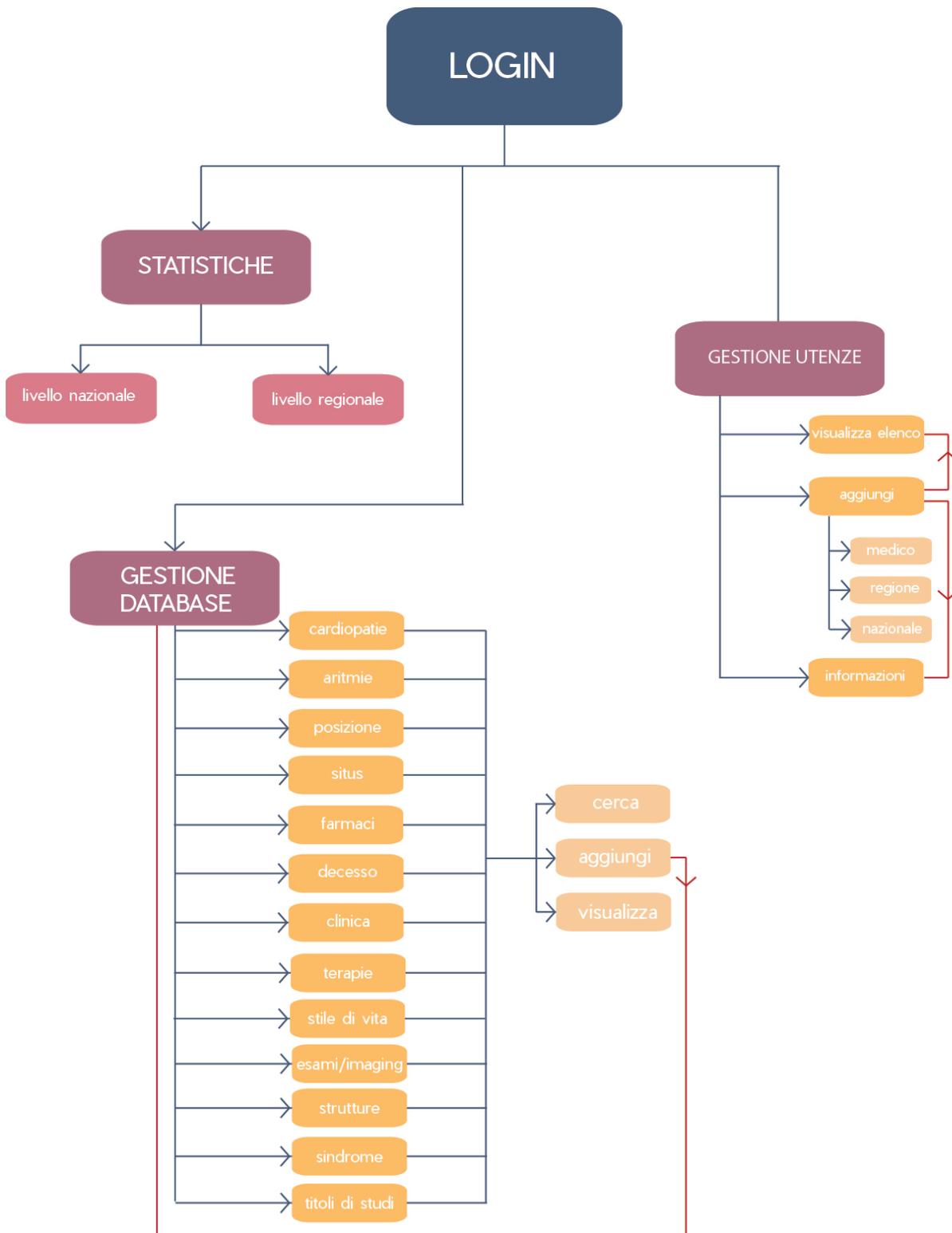


Figura 4.18: Mappa riferita all'entità nazionale

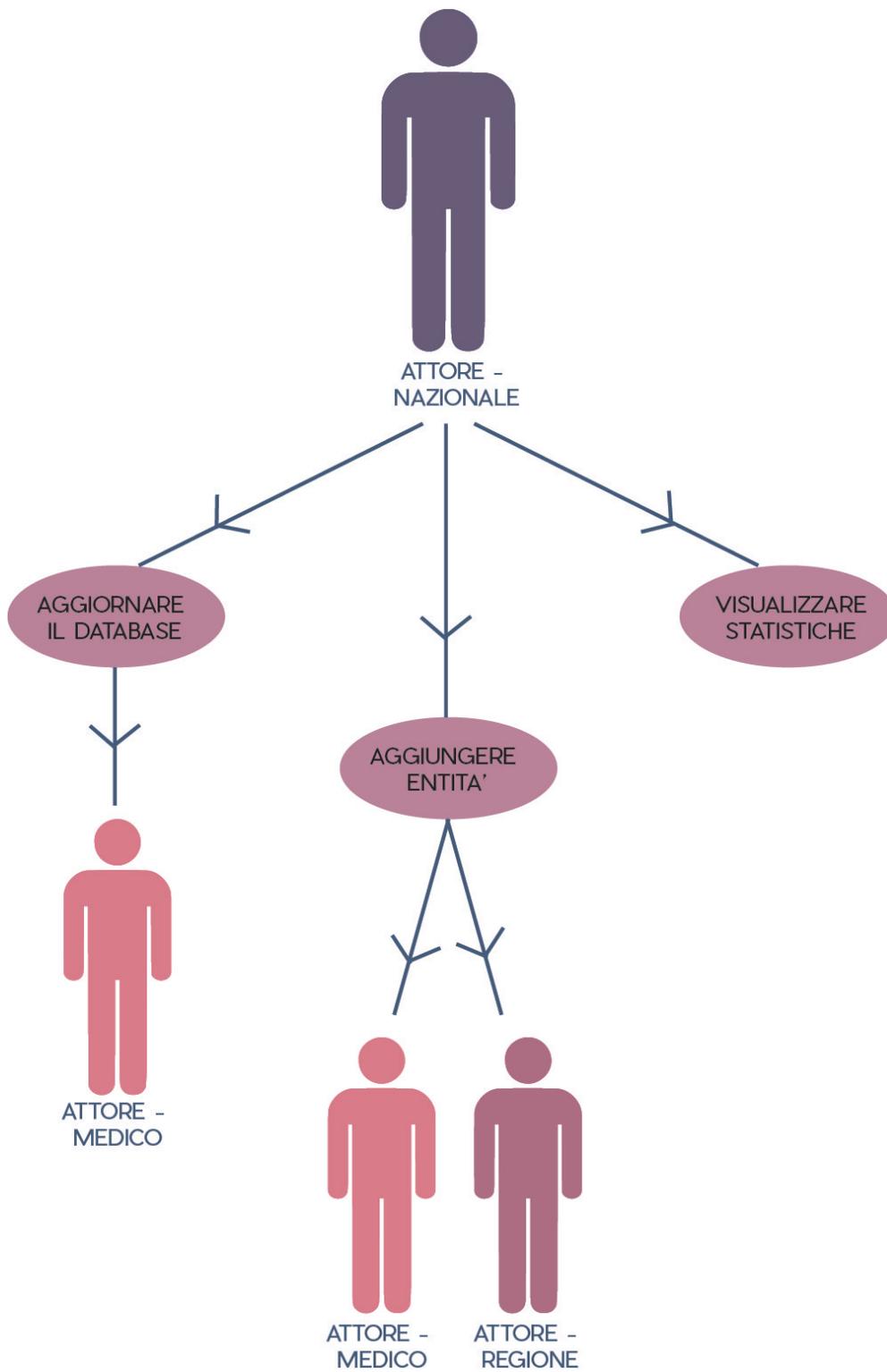


Figura 4.19: Diagramma entità nazionale

4.3.5 L'entità regionale

L'entità regionale è il ruolo più semplice dal punto di vista delle funzionalità specifiche, analizziamole nel dettaglio. La sua entità può essere unicamente creata dall'entità nazionale. A seguito del login tre sono le azioni caratteristiche che spettano a questo ruolo:

- statistiche. Anche in questo caso il referente regionale può accedere alla sezione statistiche per poter estrarre le informazioni di carattere statistico che maggiormente lo interessano. Le statistiche possono essere estratte facendo riferimento all'intera regione oppure in rapporto ad una specifica struttura;
- aggiungi struttura. Attraverso questa specifica funzionalità il referente regionale ha la possibilità di creare una nuova entità ospedaliera operante all'interno della sua regione;
- aggiungi entità. Ultima funzionalità specifica del ruolo regionale è quella di poter aggiungere e creare una nuova entità medico. All'atto di creazione di una nuova utenza medico, sarà premura del referente regionale controllare la presenza nel database della struttura ospedaliera in cui egli opera e in caso contrario aggiungerla. Ogni medico infatti fa riferimento obbligatoriamente ad una sola struttura.

Il referente regionale, così come il livello nazionale, non hanno modo di accedere a nessuna informazione medica riferita ad uno specifico paziente.

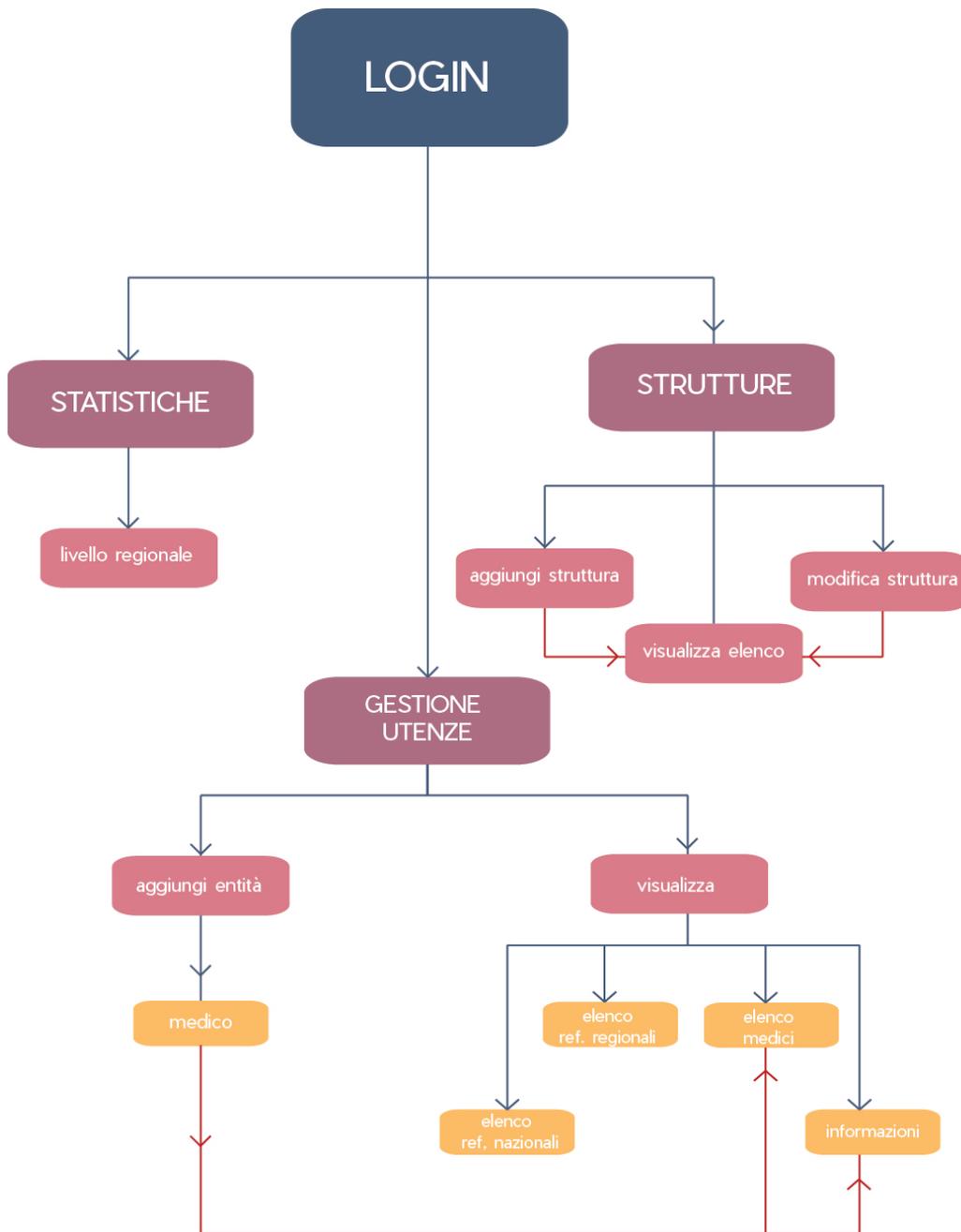


Figura 4.20: Mappa riferita all'entità regionale

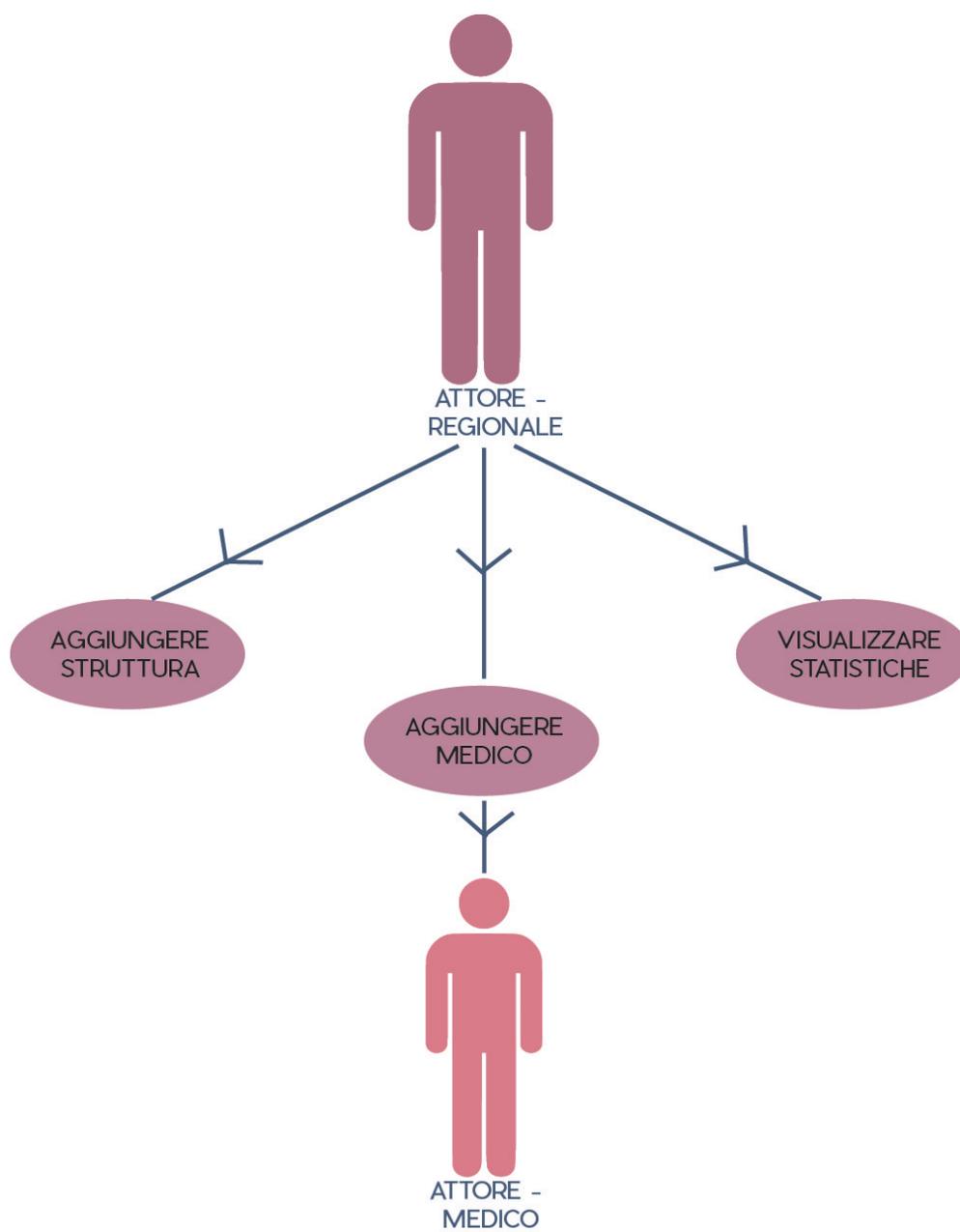


Figura 4.21: Diagramma entità regionale

4.3.6 L'entità medico

L'entità di tipo medico è a tutti gli effetti il ruolo più complesso e dalla struttura più articolata. Esaminiamone l'organizzazione.

- statistiche. Ancora una volta la prima funzionalità che analizziamo riguarda le statistiche. La possibilità di estrarre report è uno degli scopi principali per cui è stato pensato il progetto GUCH ed in questo modo ogni medico è in grado di ottenere dei dati statistici utilizzabili per eventuali studi ed approfondimenti. Nel caso di questa entità, il medico ha la possibilità di analizzare dati che si riferiscono unicamente alla struttura ospedaliera in cui opera. Nel caso si abbia necessità di ampliare i propri report ed avere informazioni di più ampio spettro (esempio livello regionale) è necessario contattare il referente regionale.

- pazienti. Seconda funzionalità chiave di questo ruolo è proprio il paziente; ogni medico ha la possibilità di accedere alla lista dei pazienti registrati presso quella specifica struttura, aggiunti personalmente o da altri medici che operano all'interno. L'elenco dei pazienti è suddiviso in: pazienti vivi e in pazienti deceduti; un paziente può essere ricercato nella lista attraverso il campo di input e se necessario l'intera lista è ordinabile in senso alfabetico in modo decrescente o crescente.

Nel caso di un paziente deceduto, il medico può accedere alla sua scheda personale, visualizzarne i dati organizzati in tab ed eventualmente modificarne le informazioni nel caso siano presenti degli errori, ma non è possibile registrare un nuovo follow-up. Nel caso invece in cui il paziente sia vivo, il medico, accedendo alla sua scheda personale, oltre ad operare le medesime azioni descritte nel caso di un paziente deceduto, può aggiungere un nuovo follow-up; con tale termine, nell'ambito medico si intende una nuova visita (esempio: visita di controllo). All'atto di registrazione di un nuovo follow-up il medico compila un form che gli permette di inserire una serie di dati relativi a: pressione (sistolica e diastolica), frequenza cardiaca, saturazione, peso e altezza. Salvando i dati inseriti nel form il medico può procedere, passando alla seconda parte, che gli consente invece di aggiungere, se lo ritiene necessario: cardiopatie, aritmie, terapie, imaging, clinica, farmaci e modifiche relative alla vita sociale. Il form, in cui è possibile inserire rilevazioni generali del paziente è sempre precompilato con tutti i dati riferiti all'ultima visita effettuata dal degente; in questo modo il medico ha la possibilità, nel caso non ci fossero rilevanti variazioni, di confermare i dati precedenti e procedere più rapidamente alla seconda parte.

Ogni follow-up è caratterizzato da una data che fa riferimento al giorno in cui è stata effettuata la visita; accedendo al tab, denominato "follow-up", il medico può scorrere tutte le visite effettuate dal paziente ordinate a partire dalla più recente. Per ogni visita il medico può scegliere se aprirla in modalità "visione" (in questa modalità non è possibile modificare nessun dato inserito, al più visionare la scheda) oppure in modalità "modifica". Con quest'ultima modalità il medico è in grado di andare a modificare ogni singolo dato inserito nell'eventualità che siano stati commessi degli errori di compilazione.

Ogni nuova voce aggiunta in un nuovo follow-up o modificata in uno precedente viene a sua volta automaticamente aggiunta e nel caso modificata nelle diverse tab specifiche; ogni medico è in grado di sapere inoltre, chi ha modificato o aggiornato uno

specifico follow-up, in questo modo tutto è tracciato ed è data la possibilità ad ogni medico di richiedere eventualmente un chiarimento ad un proprio collega.

I diversi tab sono:

- Highlights: in questo tab, che è il primo che il medico visiona nel caso acceda alla scheda del paziente, sono presenti tutte le voci derivanti dal progressivo aggiornamento dello scheda clinica che il medico ritiene importante tenere ben vista. Nel caso in cui ad esempio nell'ultimo follow-up il medico abbia prescritto al proprio paziente un nuovo farmaco, egli può scegliere di inserire l'aggiornamento all'interno del tab highlight validando una checkbox. Da quel momento la voce sarà inserita all'interno di questa sezione, fino al momento in cui il medico la riterrà un'informazione importante.
- Diagnosi: all'interno di questa tab sono presenti informazioni importanti rispetto alla diagnosi del paziente. Sono riportate rilevanti informazioni cliniche che caratterizzano lo stato di salute di un paziente affetto da cardiopatia (situs, sindrome e posizione). I dati presenti all'interno di questa tab vengono compilati la prima volta nel momento in cui viene registrato un nuovo paziente, possono essere modificati in una qualunque circostanza, ma non è possibile registrarne una seconda.
- Stile di vita: all'interno di questa tab è possibile osservare l'evolversi dello stato del paziente dal punto di vista sociale (il tipo di lavoro, il suo grado di attività sportiva ecc...).
- Rilevazioni: vengono inseriti qui tutte gli aggiornamenti dei valori di: pressione (sistolica, diastolica e media), frequenza cardiaca, saturazione, altezza, peso, massa corporea e superficie inseriti nel momento in cui viene registrato un nuovo follow-up. Tali informazioni vengono presentate all'utente attraverso un grafico, al fine di aiutare il medico nella valutazione di eventuali cambiamenti ed oscillazioni nei dati.
- Imaging: all'interno di questa sezione vengono riportati tutti i dati che fanno invece riferimento alla registrazione di un nuovo esame di imaging. Tre sono i possibili esami fra cui è possibile scegliere: TAC, risonanza magnetica ed ECO; per ognuno di essi vengono riportati tutti i dati che vi fanno riferimento.
- Cardiopatie: qui vengono riportati tutti i dati riferiti a cardiopatie di cui è affetto il paziente e che sono state diagnosticate.
- Aritmie: vengono riportate all'interno della tab tutte le aritmie riscontrate nel paziente
- Farmaci: vengono riportati nella sezione tutti i dati che si riferiscono ai farmaci somministrati al paziente.
- Terapie: vengono riportate tutte le terapie, chirurgiche e non, subite dal paziente nel decorso della sua patologia cardiaca.
- Clinica : vengono riportate all'interno della tab l'insorgere di altre complicanze, non necessariamente collegate alla cardiopatia diagnosticata, ma che devono essere tenute in considerazione nel percorso di assistenza clinica del paziente.

- Diario clinico: in questa sezione è data possibilità al medico di inserire delle note affinché sia possibile registrare dati clinici significativi o annotazioni di cui è necessario tenere traccia e che non è possibile inserire all'interno delle altre voci specifiche. Le informazioni inserite all'interno di questo tab non possono essere visualizzate dall'account del paziente.

Il medico, aggiornando progressivamente lo stato di salute del paziente attraverso i follow-up, concorre automaticamente a costruire nelle differenti sezioni la storia clinica del degente. In questo modo la consultazione di un determinato aspetto clinico del paziente risulta essere estremamente più facile e veloce, in quanto è sufficiente selezionare la tab corretta e scorrere le voci ricercando le informazioni di interesse. Nel caso di errori o incompletezze è possibile risalire al follow-up in quanto ogni voce registrata si riferisce ad una data e recandosi nella tab follow-up cercare e modificare l'informazione non corretta.

Nel caso in cui un degente non sia ancora stato registrato è possibile aggiungerlo (cliccando sull'opportuna icona); l'operazione di registrazione si compone di tre fasi fondamentali:

- Registrazione delle informazioni anagrafiche. Cliccando sull'icona "Aggiungi un nuovo paziente" si apre automaticamente il form che permette al medico di inserire tutte le informazioni anagrafiche del paziente (nome, cognome, residenza, data di nascita ecc..). Le informazioni inserite in questa fase possono essere modificate dal medico in un qualunque momento accedendo alla scheda personale del paziente.
- Registrazione della diagnosi. Una volta compilate le informazioni anagrafiche il medico dovrà compilare un nuovo form che fa riferimento alla diagnosi del paziente. In questa sede bisogna inserire le informazioni relative a: sindrome, situs e posizione.
- Registrazione di un nuovo follow-up. Terminata e salvata la compilazione della sezione diagnosi per il medico è dunque possibile registrare un nuovo follow-up che in questo caso corrisponderà alla prima visita. A partire da questo momento avrà inizio la storia clinica del paziente.

Di seguito riportiamo gli schemi che illustrano quanto spiegato all'interno di questa sezione.

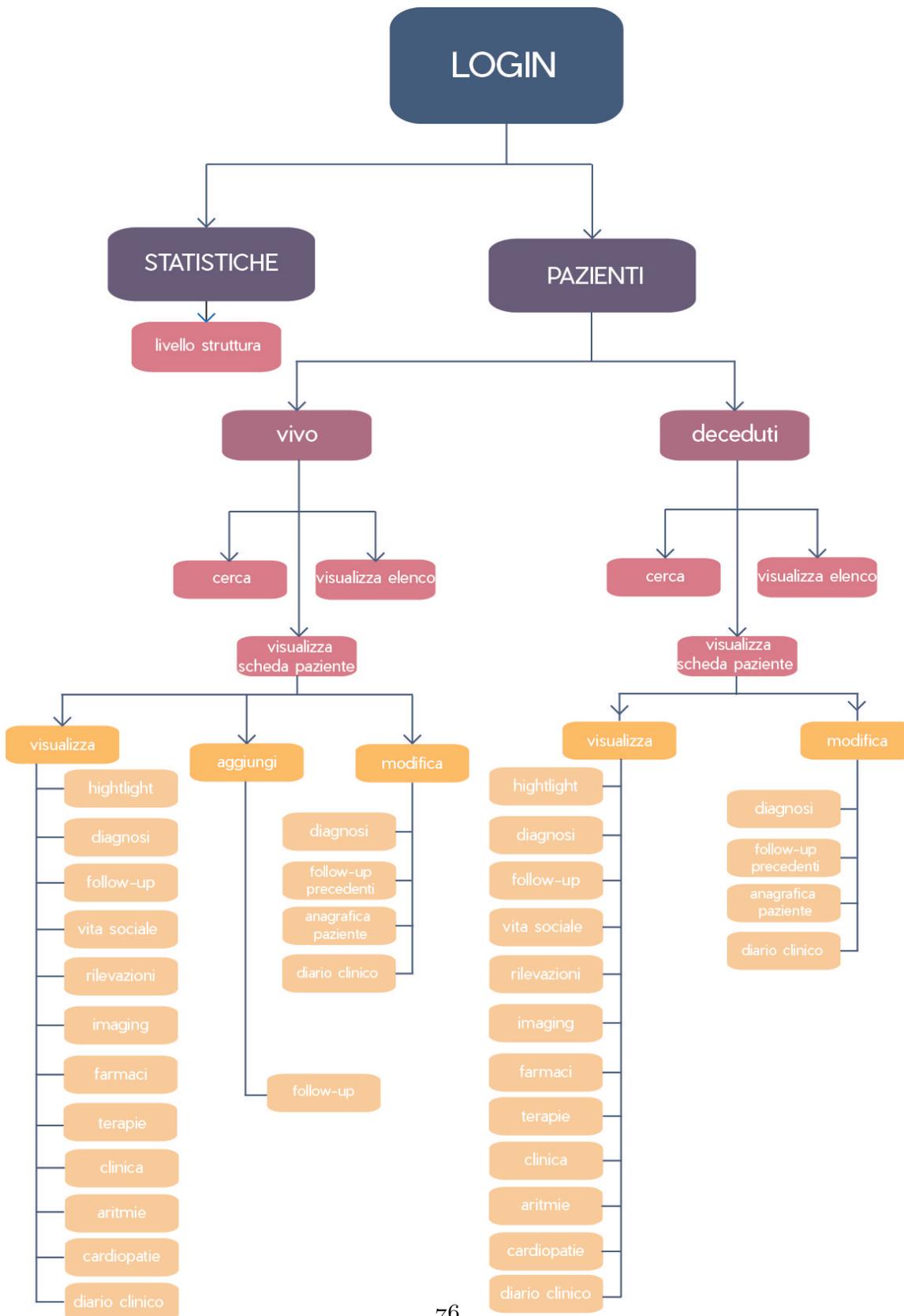


Figura 4.22: Mappa riferita all’entità medico

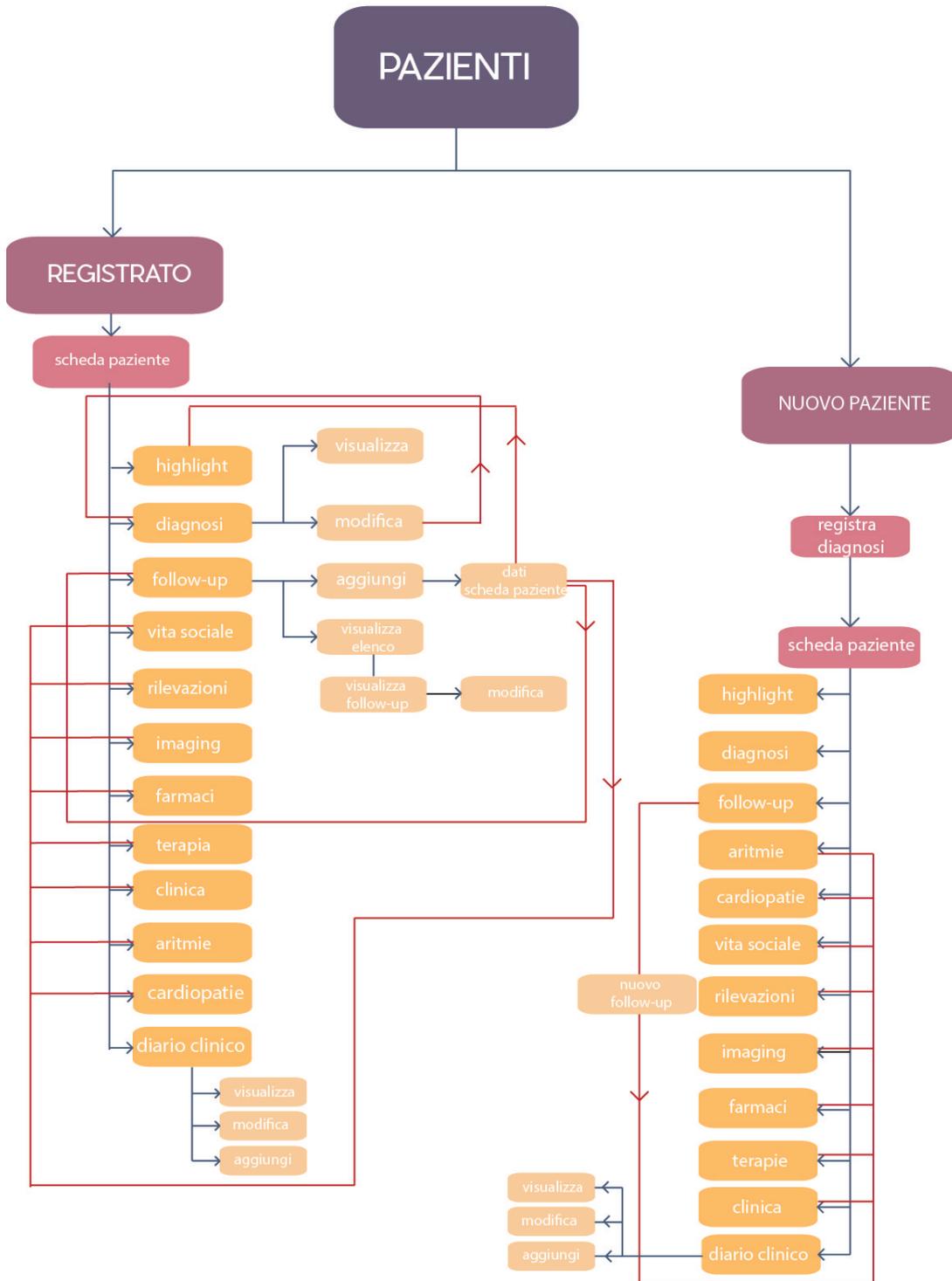


Figura 4.23: Seconda mappa riferita all'entità medico

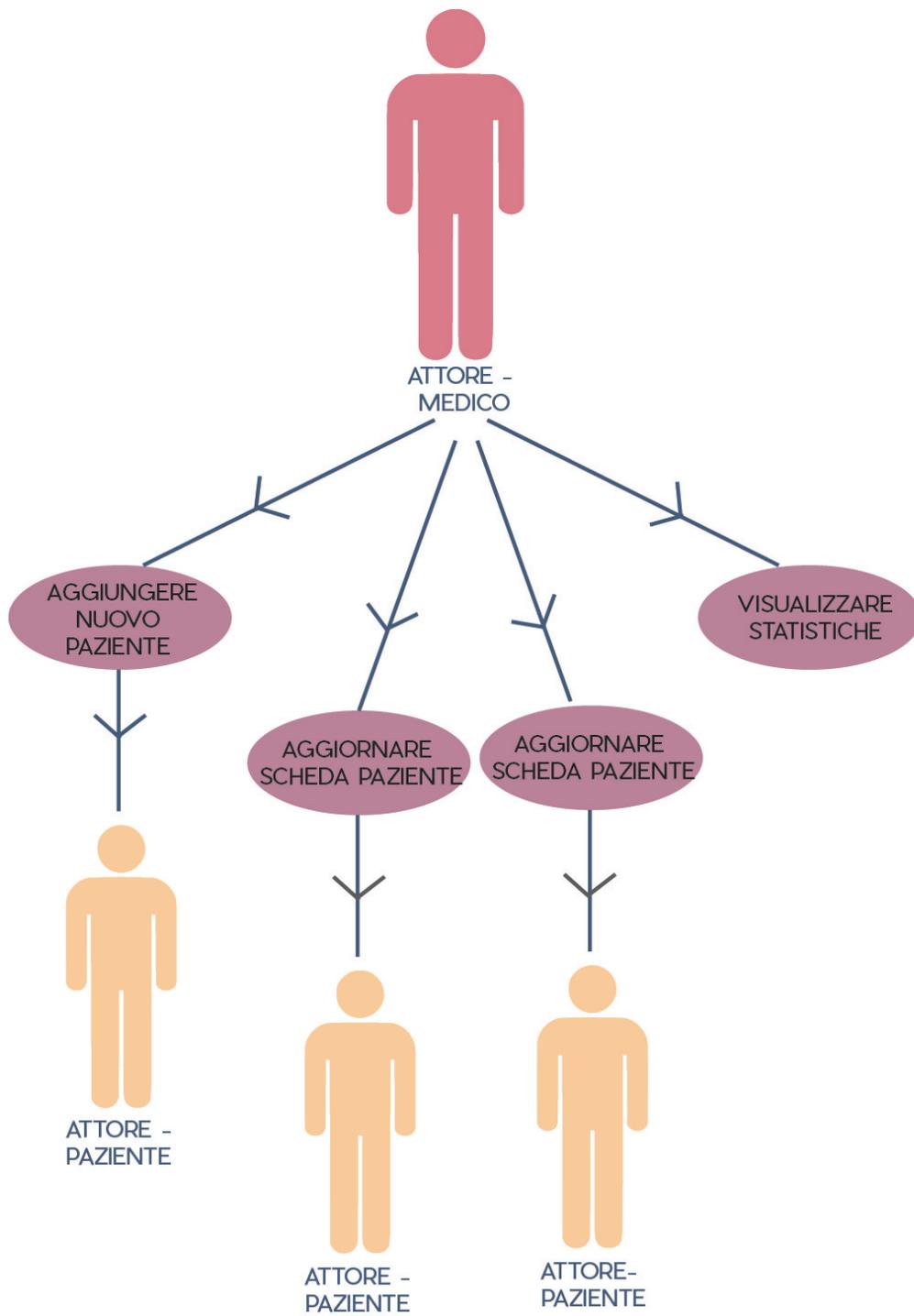


Figura 4.24: Diagramma entità medico

4.3.7 L'entità paziente

L'ultima entità che dobbiamo analizzare è l'entità paziente e questo profilo è quello che maggiormente è stato modificato rispetto al vecchio progetto. Sono state infatti aggiunte alcune funzionalità specifiche che sono state ritenute importanti per dare al paziente uno strumento efficace ed utile nella sua progressione clinica. L'entità paziente è per lo più un ruolo di presentazione dati e non sono molte le informazioni che lui stesso è in grado di modificare. Analizziamo la struttura e le funzionalità principali di questo ruolo.

- **Calendario/Agenda:** è stata messa a disposizione del degente una nuova funzionalità, rispetto al precedente progetto. All'interno di questa sezione egli può inserire a calendario eventi e appuntamenti clinici; in questo modo il paziente ha uno spazio personale per tenere traccia delle sue prossime visite. Per ogni evento il paziente può crearlo, modificarlo e visualizzarlo all'interno della sua agenda virtuale.
- **Diagnosi:** all'interno di questa tab vengono riportate tutte le informazioni che riguardano la situazione clinica del paziente e che il medico ha inserito all'interno del registro. È in questa sezione, dunque che il paziente può monitorare il suo decorso clinico e ritrovare le informazioni specifiche riguardanti la sua diagnosi. L'unica sezione che il medico non riporta al paziente è il diario clinico, che rimane quindi privata per il medico.
- **Taccuino medico:** anche questo rappresenta una nuova sezione. All'interno di questa tab il paziente, in una sorta di diario personale, può annotare dati e informazioni che ha bisogno di ricordare e riportare al medico nella successiva visita. In questo modo l'utente ha uno spazio dedicato a se e di aiuto per mantenere sotto controllo il proprio stato di salute.
- **Storia clinica/Timeline:** in questa sezione viene riportata graficamente su una timeline la storia clinica del paziente e attraverso dei filtri è possibile scegliere quale tipo di informazione mostrare sulla linea temporale. In questo modo il paziente è in grado di osservare il suo percorso clinico e l'evolversi del suo stato.
- **Anagrafica:** questa è l'unica sezione in cui le modifiche apportate dall'utente si ripercuotono anche sulle informazioni visualizzate dal medico nella scheda del paziente. In quest'area il degente ha la possibilità di modificare le sue informazioni anagrafiche e quelle di contatto (e-mail, telefono, indirizzo).

Le informazioni e i dati inseriti (tranne quelli relativi all'anagrafica) dall'utente sono esclusivamente visibili a lui, il medico non vi può in alcun modo fare accesso. La struttura e il tipo di layout che si è scelto per il profilo del paziente è identico a quello delle altre entità, al fine di mantenere la stessa impostazione grafica. L'abilitazione del profilo utente spetta al medico, in ogni momento il paziente può farne richiesta o vi può rinunciare. La prima volta, il paziente accede utilizzando l'e-mail fornita al medico e utilizzando come password il suo nome e cognome senza spazi; sarà premura dell'utente, a seguito del primo accesso procedere alla modifica della password seguendo la specifica procedura.

Le difficoltà di sviluppo di questa entità sono state per lo più causate dalla necessità di rendere ogni sezione e ogni elemento grafico responsive e accessibile da dispositivi mobile

(smartphone e tablet), al fin di garantire piena operabilità dell'applicativo indipendentemente dal mezzo utilizzato. Per questa ragione, è stato necessario in fase di sviluppo rivedere il progetto e modificarlo in buona parte. Di seguito vengono riportate alcune immagine a scopo illustrativo di come si presenta l'applicativo per l'entità paziente.

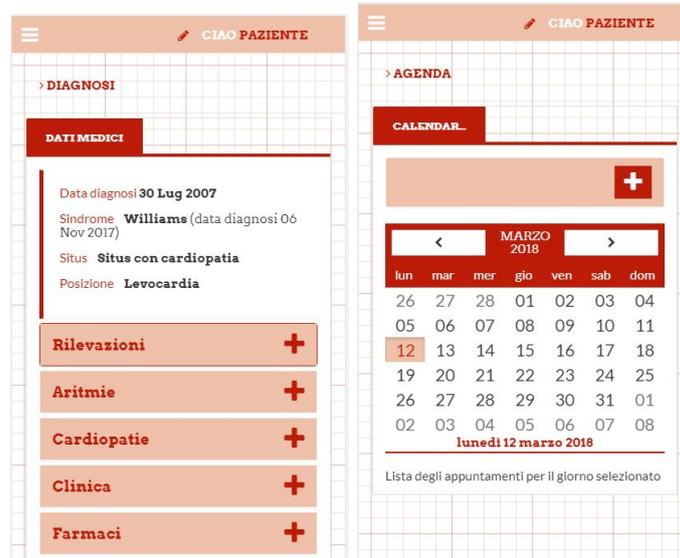


Figura 4.25: Sezione diagnosi e agenda - Dispositivo smartphone

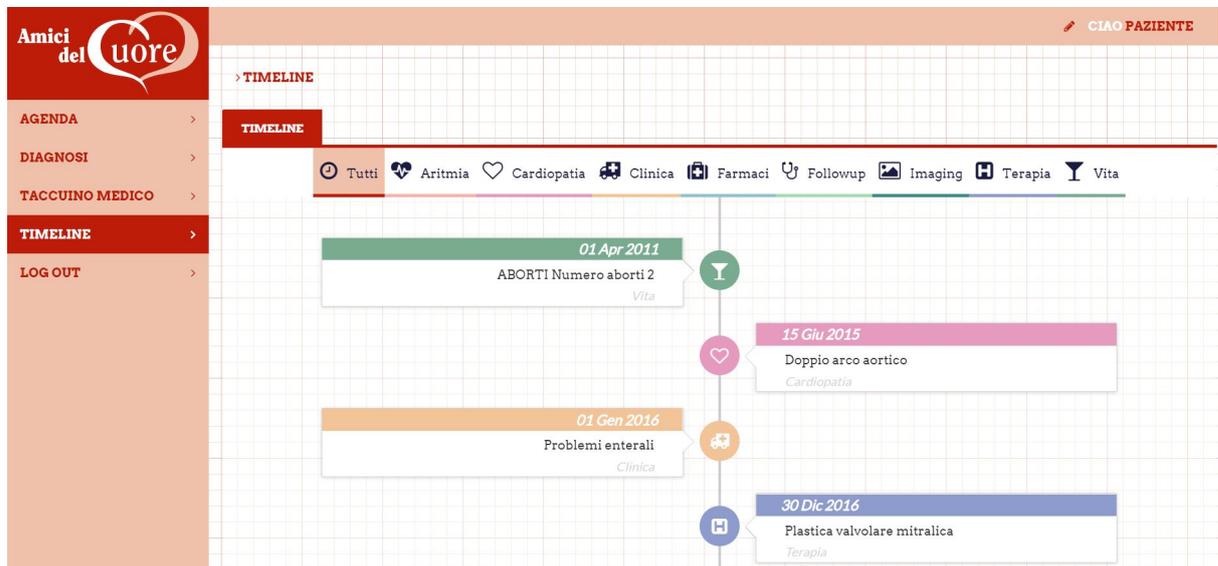


Figura 4.26: Sezione timeline - Dispositivo desktop

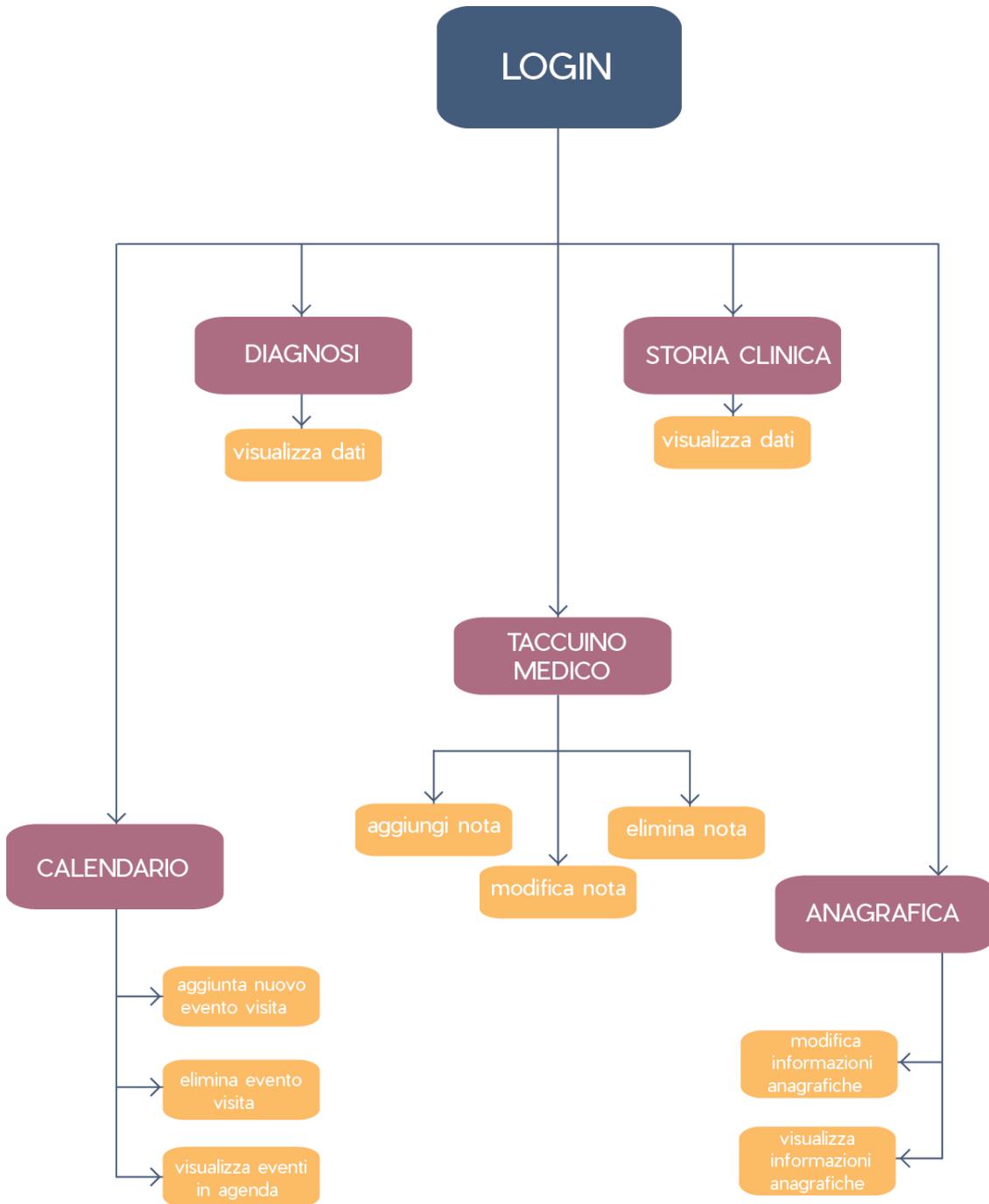


Figura 4.27: Mappa riferita all'entità paziente

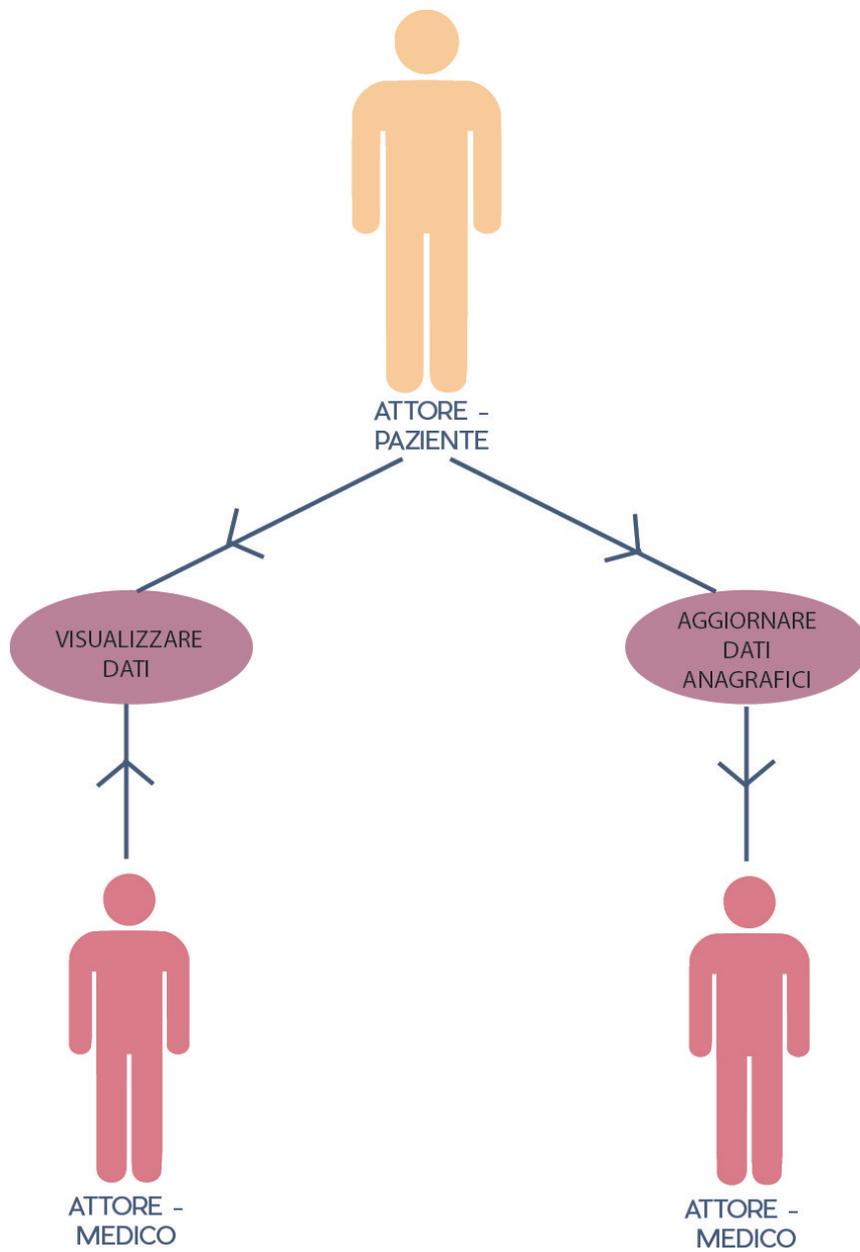


Figura 4.28: Diagramma entità Paziente

Capitolo 5

I test

5.1 Introduzione

Quando si ha a che fare con la progettazione Web è bene ricordare che il più delle volte non esistono delle vere e proprie risposte alla maggior parte delle domande e dubbi che possono emergere nel processo di progettazione e realizzazione. Come è stato possibile osservare ci possono essere delle linee guida che possono essere applicate al fine di limitare i problemi e aiutare quindi il progettista a trovare la soluzione migliore. Spesso è proprio l'esperienza del progettista, la sua creatività e il buon senso dell'intero gruppo di sviluppo che facilita e supporta il processo di ideazione di un'interfaccia grafica utile alle esigenze dell'utente. È giusto considerare e includere nel processo di sviluppo di un applicativo i futuri utenti che lavoreranno e sfrutteranno il prodotto ed è proprio qui che acquistano un'importante rilevanza i test di usabilità, utilizzati, secondo gli esperti, molto poco dagli sviluppatori, ma possono aiutare a superare dubbi e arginare inefficienze strutturali del proprio prodotto. I test possono essere considerati a tutti gli effetti degli strumenti fondamentali e pratici per migliorare un proprio prodotto per il Web e tutto quello che comporta un'interazione con l'utente.

In questo capitolo analizzeremo come sono stati utilizzati alcuni test al fine di migliorare e curare nel dettaglio ogni aspetto dell'applicativo GUCH; l'analisi partirà da una piccola introduzione del mondo dei test al fine di comprendere meglio quali vantaggi apportano e come è possibile sfruttarli.

5.1.1 I test di usabilità

Prima di poter procedere, è bene togliere ogni dubbio e dedicare alcune righe di questo elaborato a spiegare e definire cosa si intende con "test di usabilità". Alcuni esperti del settore, come ad esempio Jakob Nielsen, definiscono test di usabilità quell'azione in cui il progettista osserva le persone mentre provano ad utilizzare quanto lo sviluppatore ha progettato/costruito/creato al fine di comprendere come semplificare e migliorare il suo utilizzo (anche in un'ottica futura) o avere delle conferme in merito alle scelte progettuali portate avanti. Secondo l'approccio di Nielsen, i test non hanno la finalità di decidere con elevata affidabilità quale sia l'interfaccia migliore fra una versione e l'altra, al contrario il loro unico scopo di realizzazione è quello di mettere in luce i problemi precocemente, ovvero

quando è più conveniente intervenire con delle modifiche alla struttura dell'interfaccia. Si possono scegliere diverse tipologie di test, ma tutte prevedono comunque l'osservazione e l'analisi delle azioni e delle scelte compiute dall'utente per far fronte ad un piccolo compito. La differenza fra test di usabilità e sondaggio risiede nel fatto che in una valutazione di usabilità al soggetto fruitore è chiesto di svolgere praticamente un compito e di utilizzare concretamente un prodotto, non viene chiesta semplicemente una sua opinione.

Possiamo classificare i test di usabilità in due macrofamiglie.

- test quantitativi. L'obiettivo con cui si svolgono questi tipi di test è quello di dimostrare qualcosa. Per far ciò è quindi necessario fare delle misure, ad esempio valutazioni di successo che permettano di individuare quante persone riescono a portare a termine un determinato compito o misure relative al tempo di risposta, che permettono invece di avere una visione relativa di quanto tempo impiega la persona a terminare uno specifico compito. Per completare un test di tipo quantitativo in modo efficace e ottenere dei dati effettivamente adoperabili è necessario utilizzare un protocollo e far sì che tutti i partecipanti al test lo seguano attentamente. Questo tipo di test può essere paragonato ad un vero e proprio esperimento scientifico; è quindi necessario essere rigorosi e precisi al fine di non ottenere dati ingannevoli, e prestarci massima cura e attenzione. Altro parametro da tenere in considerazione è sicuramente legato al numero di utenti partecipanti al test; il bacino di utenza deve essere il più ampio possibile; è infatti fondamentale poter ricavare conclusioni statistiche significative, in grado di rappresentare, a grandi linee, il maggior numero possibile di futuri utenti; l'estrazione di dati applicabili e generalizzabili concorre al buon successo del test e comporta necessariamente un miglioramento significativo della qualità dell'applicativo testato.

- test qualitativi. Il loro principale scopo non è quello di dimostrare una tesi, ma ottenere informazioni utili e pratiche al fine di migliorare quanto si sta realizzando. Questo fa sì che, questo tipo di test sia strettamente più informale della precedente tipologia e decisamente meno scientifico. I test qualitativi possono essere portati avanti utilizzando come bacino di utenti un numero inferiore di persone e il protocollo non è vincolante, ma all'occorrenza può essere modificato anche in corso d'opera in qualsiasi momento.

Il funzionamento di questo tipo di test è molto semplice; di norma un facilitatore, ovvero colui che aiuta la persona a comprendere quali siano i suoi obiettivi e a raggiungerli, si posiziona all'interno di una stanza con il partecipante al test e gli assegna alcuni semplici compiti da portare a termine. Alla persona impegnata nel test è chiesto di parlare ad alta voce, spiegare quello che sta facendo, il perché delle sue scelte e i suoi dubbi. Gli sviluppatori osservano e prendono nota dei gesti e delle affermazioni dell'utente; il loro obiettivo è mettere in luce i dubbi ed eventuali esitazioni della persona.

Al termine del test avviene quello che gli esperti definiscono debriefing ed è una fase fondamentale, che non deve essere sottovalutata. Con il debriefing gli osservatori si confrontano sulle annotazioni prese e decidono quali sono i problemi che dovranno

essere in secondo momento valutati e risolti. È in questa fase che normalmente emergono nuove prospettive che richiedono di mettere nuovamente in discussione alcune scelte prese.

5.1.2 I test di usabilità e i risultati

Apparentemente, leggendo questa piccola e schematica classificazione possiamo essere spinti a pensare che i test di usabilità siano estremamente difficili da portare a termine, inoltre ci si può convincere che l'investimento economico e temporale richiesto non sia ripagato dai dati ottenuti. I molteplici studi condotti confermano però l'effettivo successo dell'approccio. Analizziamo allora i motivi per cui, anche i più semplici test di usabilità hanno avuto successo:

- funzionano perchè tutti gli applicativi portano con se dei problemi.
- funzionano perchè i problemi importanti e gravi sono facilmente identificabili, ma per farlo, spesso è necessario affidarsi ad occhi diversi da quelli di un progettista e sviluppatore. Un problema di usabilità viene sottovalutato o, addirittura ignorato da colui che ha progettato l'applicativo perchè egli sa come funziona e conosce tutto il processo di sviluppo.
- funzionano perchè osservare le persone che portano a termine un compito migliora la capacità di progettazione dello stesso progettista e sviluppatore. È infatti importante per la figura professionale che si occupa dell'ideazione di un'interfaccia lavorare non facendo riferimento ad un'immagine utopica e distante dell'entità utente, nè tanto meno è vantaggioso per lui proiettare le sue idee a costruzione di un'utente tipico, incorrendo nell'errore di definire quindi uno stereotipo troppo simile a se stesso. Inoltre è importante sottolineare come le capacità di osservazione e deduzione possano essere allenate e sarà quindi, per il progettista, progressivamente più facile con l'esperienza comprendere determinate dinamiche di ragionamento dell'utente andando ad arginare o risolvere eventuali problemi in minor tempo.

5.1.3 Scegliere che cosa testare

Per poter ottenere dei dati utilizzabili e utili è bene identificare che cosa testare, da tale scelta dipenderanno automaticamente i possibili problemi e, ciò che si dovrà risolvere. È quindi importante comprendere per prima cosa se vi possono essere all'interno dell'applicativo dei nodi cruciali che devono trovare conferme e che hanno bisogno di essere provati dall'utente allo scopo di comprendere se in fase di progettazione si sono operate delle scelte corrette. Oltre a capire che cosa testare è importante individuare anche il periodo in cui è necessario iniziare ad eseguire i primi test. Gli esperti consigliano di iniziare le valutazioni il prima possibile, infatti è meglio risolvere subito determinati problemi piuttosto che aspettare di costruire un applicativo che li contenga tutti; il tempo diventa una componente fondamentale, talvolta alcuni problemi possono essere troppo complicati per essere poi risolti in un secondo momento quando il progetto ha già assunto una struttura più complessa e maggiormente articolata. In base a cosa si decide di testare è necessario definire la corretta strategia e identificare il metodo più efficace per portare a termine dal test i dati e le informazioni di interesse. Ricapitolando è quindi necessario identificare:

- che cosa testare e quando;
- cosa voglio ottenere dal test;
- quale strumento utilizzare.

Per scegliere "che cosa testare", come già affermato in precedenza, è bene per prima cosa concentrarsi sui punti critici e cruciali. La scelta, si può basare su più fattori:

- Quali possono essere i compiti critici che l'utente sarà poi chiamato a portare a termine? Per compiti critici possiamo intendere tutte quelle operazioni che le persone dovranno essere in grado di svolgere e nel caso non vi riuscissero, allora lo scopo dell'applicativo sarebbe fallito.
- Quali sono i nodi cruciali in cui lo sviluppatore ha bisogno di trovare delle conferme? È utile testare quindi tutte quelle funzionalità su cui lo stesso sviluppatore nutre delle perplessità, perchè non sono chiare o perchè potrebbero confondere l'utente.
- A seguito dei precedenti test sono emerse delle problematiche e sono state risolte? Sono migliorate? È importante rivalutare e quindi rimettere nuovamente in discussione scelte compiute a seguito di precedenti test che hanno portato ad evidenziare problematiche risolte in un secondo momento.

5.1.4 Il debriefing

La fase successiva a quella di test, in cui gli utenti sono protagonisti, viene chiamata debriefing ¹. È fondamentale per andare a discutere quanto emerso dal precedente stadio di test e valutare i problemi che devono essere risolti al fine di migliorare l'applicativo. La discussione avviene immediatamente al termine dei test, in quanto è importante che tutti gli sviluppatori e i progettisti dibattano ricordando ancora bene l'interazione utente-applicativo. La riunione di debriefing deve far emergere:

- un elenco dei problemi di usabilità. Per aiutare una successiva valutazione delle azioni da intraprendere, può essere utile classificarli.
- un elenco dei problemi che devono essere immediatamente risolti; qui diventa fondamentale essere in grado di valutare la portata di un problema identificando quelle che sono le questioni principali che hanno causato nell'utente partecipante al test maggiore difficoltà. L'analisi deve pertanto partire sempre dall'osservazione della persona mentre svolge i compiti chiave dell'applicativo.

È importante essere in grado di identificare i problemi di usabilità principali perchè progettisti e sviluppatori si scontrano con altri due problemi il tempo e le risorse. A maggior ragione se le fasi di test sono iniziate tardi sarà molto probabile che le tempistiche di consegna del progetto si facciano progressivamente più strette. Non lavorando per priorità

¹Il termine indica la valutazione finale di un processo. Deriva dal linguaggio militare e può essere tradotto in: andare a rapporto al termine di una missione.

si corre il rischio di consegnare un progetto in cui permangono problemi gravi, mentre i dettagli risultano invece curati, incorrendo in una situazione incongruente. Per comprendere quali sono i problemi gravi tendenzialmente basta osservare con attenzione come gli utenti del test si muovono all'interno dell'applicativo; si determina un problema grave se molti utenti segnalano o incorrono nel medesimo errore e se coloro che sono incappati nel problema hanno conseguentemente avuto serie difficoltà.

Differenziare un problema comporta necessariamente che venga espresso, da parte dello sviluppatore, un giudizio. Per far sì che tale valutazione non comporti un'errata classificazione è bene concentrarsi unicamente sull'osservazione dell'utente nella sua interazione con l'applicativo; solo in questo modo si eviterà di prendere in considerazione casi limite e non realistici.

Dopo aver analizzato i problemi, bisogna definire le strategie di risoluzione; quando si pensa a come modificare un elemento all'interno del progetto è sempre necessario valutare come migliorarlo, al fine di risolvere il problema incorrendo nel minor numero possibile di modifiche. Spesso è sufficiente operare tendendo a fare il meglio possibile, limitando uno specifico problema anziché puntare alla sua completa risoluzione, infatti non necessariamente ad un problema complesso e rilevante corrisponde una soluzione difficile nella sua identificazione e implementazione. Fondamentale per risolvere un problema di usabilità è quello di puntare a modifiche semplici e immediate tenendo sempre a mente quale è il problema da risolvere e l'obiettivo ad esso collegato. Trovare una soluzione non deve coincidere necessariamente con il fatto che un futuro utente non incappi più nell'errore, semplicemente significa evitare che il maggior numero di persone possibili incorrano nel medesimo problema.

5.2 I test di GUCH

Analizziamo ora insieme come sono stati sfruttati e portati avanti i test in GUCH. In questa sezione esamineremo i principali passaggi utilizzati concentrandoci, per lo più su quanto ricavato e sulle successive modifiche apportati con lo scopo di migliorare il prodotto finale.

5.2.1 Prima fase di test

Testare il mock-up

Il primo test effettuato ha permesso di valutare il mock-up realizzato. Per realizzare il mock-up, definito anche wireframe, è stata fatta una prima riunione con la dottoressa referente del progetto al fine di:

- mettere in luce gli obiettivi e gli scopi stessi dell'applicativo;
- chiarire dubbi sulla struttura e presentazione dei dati;
- ricevere impressioni e richieste rispetto alla struttura dell'interfaccia.

Solo a seguito del primo incontro si è passato all'effettiva realizzazione del wireframe attraverso l'utilizzo di un programma specifico, *Balsamiq*². Con il termine mock-up o wireframe, ci si riferisce ai diagrammi schematici di una pagina i quali aiutano lo sviluppatore a identificare dove devono essere posti i differenti contenuti, a definire una scala di priorità e determinare l'importanza relativa dei differenti elementi (quali ad esempio: menu di ricerca, strumenti di navigazione titoli ecc...) da posizionare. Partire dalla progettazione e dalla realizzazione di un wireframe aiuta a rimanere concentrati sull'essenziale e a focalizzare l'attenzione unicamente sulla struttura stessa dell'interfaccia, piuttosto che su aspetti importanti, ma considerabili secondari come possono essere colori, icone e affinamenti vari. I wireframe non sono solo utili allo sviluppatore, ma permettono anche al cliente finale di farsi un'idea più precisa e definita di quello che sarà il prodotto alla sua conclusione. Se il cliente ha un visione maggiormente definita e concreta dell'applicativo sarà per lui, di conseguenza, molto più facile effettuare una valutazione ed una più puntuale analisi. Durante il test della prima versione del mock-up che ho realizzato ci si è concentrati maggiormente sull'analisi:

- della navigazione dell'applicativo;
- della struttura e la disposizione dei differenti contenuti;
- dell'accessibilità alle informazioni cliniche dei pazienti registrati;
- dei nomi utilizzati e delle icone.

Principalmente, si è testato insieme alla dottoressa lo schema e la struttura e si è cercato di comprendere se tutti gli elementi fondamentali fossero sufficientemente chiari e di facile accesso; allo stesso tempo era premura comprendere e valutare se era chiaro il funzionamento degli strumenti di navigazione. Questa prima fase di test è stata utile per comprendere come migliorare l'organizzazione del sito e il suo organigramma, contemporaneamente per valutare e discutere insieme l'impatto delle modifiche rispetto al precedente progetto e le nuove idee proposte (ad esempio sezione highlights per il personale medico oppure sezione agenda o storico clinico per il paziente) al fine di fornire ai medici e ai pazienti solo funzionalità utili e non superflue.

La fase di valutazione del wireframe ha permesso di porre l'attenzione su alcune considerazioni e ha messo in evidenza alcune modifiche da apportare in particolar modo a:

- sezione imaging. La prima idea si è rilevata non funzionale alle esigenze del personale medico ed è stato quindi necessario modificare l'idea di partenza per permettere un più facile caricamento dei dati associati ad un referento, a seguito di un esame diagnostico per immagini (esempio TAC, risonanza magnetica o ecografia).
- registrazione di un nuovo paziente e presentazione lista dei malati già inseriti. Si è provveduto a modificare i dati richiesti, l'ordine e la loro presentazione, mettendo in evidenza tutti i campi obbligatori. Si sono inoltre aggiunte in tabella le informazioni

²Balsamiq è un tool grafico pensato per sviluppatore, designer e progettisti che permette di "schizzare" interfacce utente e schermate (wireframe) per siti Web e applicazioni.

desiderate dalla dottoressa (oltre a nome e cognome, anche data di nascita e sesso, eliminando il luogo di residenza.) e corretti alcuni termini specifici.

- gestione del follow-up. A lungo si è discusso sulla struttura pensata per la registrazione di un nuovo follow-up e per la sua consultazione. È sicuramente la sezione più importante della scheda personale di un paziente ed era premura definirne limiti e punti di forza della proposta fatta al fine di modificarla e renderla maggiormente funzionale alle specifiche di un medico. Al termine della prima fase di test sono state apportate alcune modifiche che riguardavano in particolar modo la modalità di aggiornamento e modifica dei dati inseriti e chiariti alcuni dubbi.
- è stato richiesto di porre attenzione ai messaggi di conferma a seguito di una qualsiasi operazione, ad esempio sulla scheda del paziente e all'invio di email a garanzia dello svolgimento di una specifica azione.
- è stato possibile definire insieme al medico referente del progetto come gestire il trasferimento di un paziente ad nuova struttura e di conseguenza discutere in modo più preciso e completo la creazione di una nuova entità medico, la sua abilitazione all'utilizzo dell'applicativo e l'associazione ad una struttura di riferimento già presente nell'elenco o ancora da aggiungere.
- analizzati nel dettaglio, sono stati molto apprezzati i tab e la complessiva organizzazione dei contenuti.

A seguito del primo incontro che è servito per l'appunto a definire e chiarire meglio alcuni meccanismi con l'aiuto del mock-up, ne è stato organizzato un secondo, con lo scopo, questa volta di rivalutare le modifiche pensate al fine di risolvere i problemi riscontrati in precedenza e valutare il procedere del progetto.

Nel secondo appuntamento non solo si è nuovamente testato il wireframe preparato, ma è stato possibile anche valutare insieme la parte di progetto già sviluppata e riferita alla gestione del database.

Seconda presentazione del progetto

Ad inizio dicembre il progetto è stato presentato ad un gruppo di medici cardiologi dell'Ospedale Molinette di Torino; a tale presentazione ha partecipato unicamente il responsabile dell'azienda, Roberto Maiocco.

Lo scopo dell'incontro era quello di fare nuovamente il punto sullo stato di avanzamento del progetto e poter definire le successive scadenze. È stata anche un'occasione utile per presentare l'idea e il progetto a nuovi medici interessati ad utilizzare il nuovo l'applicativo non appena terminato e a tutti coloro che già ne utilizzavano la precedente versione. Durante la riunione sono state raccolte nuove impressioni, consigli di sviluppo da applicare e accortezze da tenere in considerazione. In sede di riunione, con qualche inconveniente tecnico è stato possibile mostrare il mock-up e la parte di interfaccia utente già sviluppata e funzionante.

L'incontro tenutosi è stato importante ed utile, non solo al fine di ricevere, per l'appunto, nuovi punti di vista sempre utili in un'ottica di progressiva ottimizzazione del prodotto finale, ma anche per incuriosire nuovi medici nel suo utilizzo e incoraggiare l'apertura del

bacino di utenza. Ricordiamo infatti che uno dei principali obiettivi del progetto è proprio quello di ricavare dati statistici utili per alimentare la costante ricerca in questo settore ed è pertanto indispensabile poter contare su di un numero consistente di dati per poter trarre considerazioni fondate e non approssimative.

Di seguito alcune immagini di riferimento estratte dal mock-up realizzato.

The screenshot shows a web application interface for a medical entity. At the top, there is a navigation bar with a search icon and a menu icon. Below the navigation bar, the user is logged in as 'Benvenuto _____' with a profile picture and 'Data e ora' next to it. The main content area is divided into several sections:

- Left Sidebar:** Contains a search icon, a list of 'Pazienti' and 'Statistiche', and a profile picture of a patient. Below the profile picture, the patient's name 'NOME COGNOME' and contact information 'Telefono', 'email', and 'Indirizzo' are displayed. Further down, medical conditions 'Cardiopatia', 'Situs', 'Posizione', and 'Sindrome' are listed, along with icons for a trash can and a lock.
- Top Navigation:** A horizontal menu with tabs: 'Highlights', 'Diagnosi', 'Follow-Up', 'Vita sociale', 'Rilevazioni', 'Imaging', 'Aritmie', and 'Farmaci'.
- Main Content Area:** This section is for registering a diagnosis. It includes:
 - 'Data diagnosi' with a date input field and a calendar icon.
 - 'Sindrome' with a text input field.
 - 'Data diagnosi sindrome' with a date input field and a calendar icon.
 - 'Situs' with a text input field.
 - 'Posizione' with a text input field.
 - 'Salva' and 'Annulla' buttons at the bottom right.

Figura 5.1: Mock-up: vista entità medico. Registrazione diagnosi

The screenshot shows the same web application interface, but now focused on registering a follow-up. The 'Follow-Up' tab in the top navigation is highlighted. The main content area is titled 'Primo follow-Up' and contains:

- 'Data visita: 26-10-2017'.
- 'Vita sociale' with a large text input area.
- 'Rilevazioni' section with several input fields:
 - 'Altezza: 180' with a spin button.
 - 'Pressione Sistolica 75' with a spin button.
 - 'Diastolica 120' with a spin button.
 - 'Media: 100'.
 - 'Peso: 69' with a spin button.

Figura 5.2: Mock-up: vista entità medico. Registrazione di un nuovo follow-up

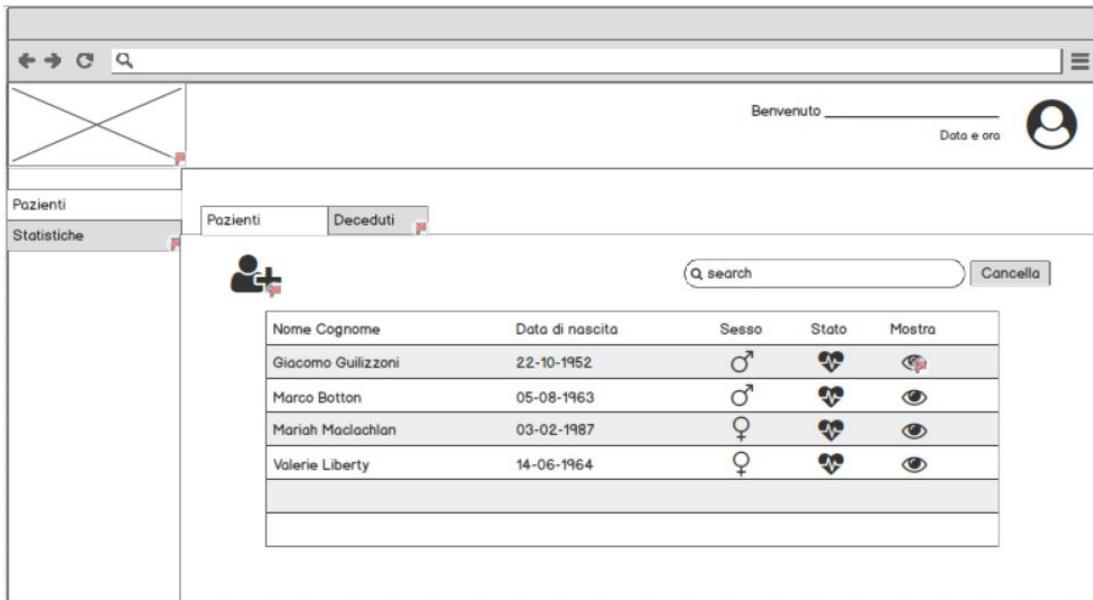


Figura 5.3: Mock-up: vista entità medico. Vista elenco pazienti

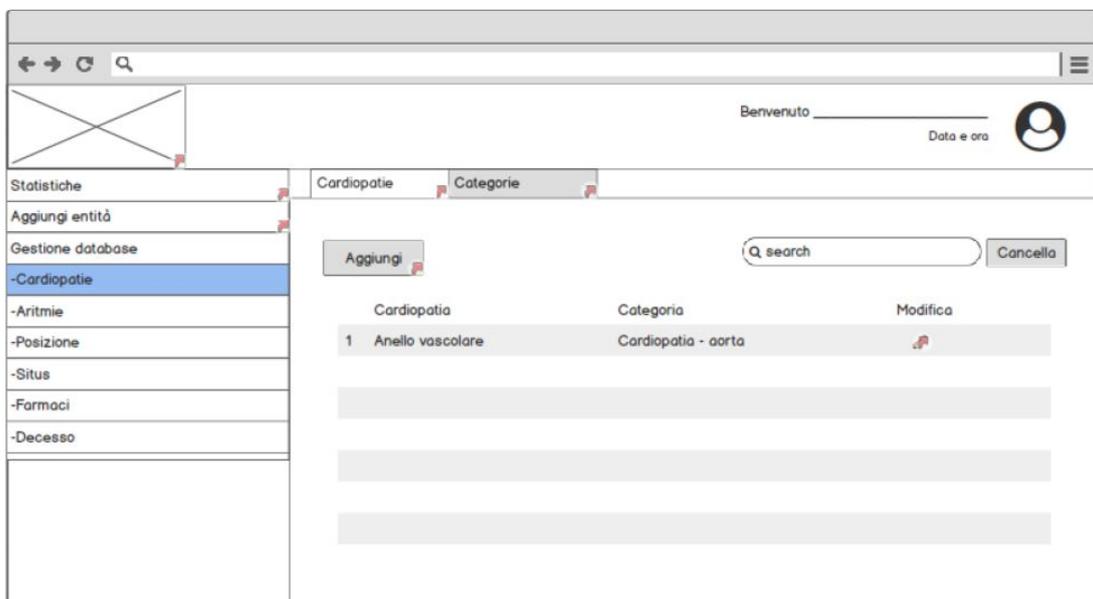


Figura 5.4: Mock-up: vista entità nazionale. Gestione database

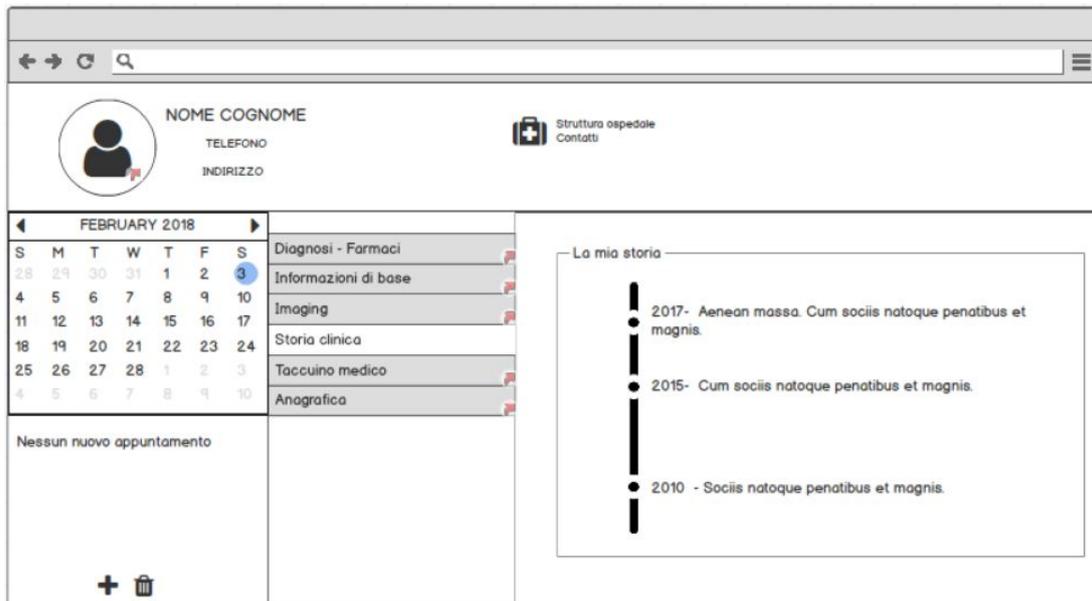


Figura 5.5: Mock-up: vista entità paziente.

5.2.2 Seconda fase di test

La seconda fase di test ha avuto come scopo principale la verifica e l'analisi dell'usabilità dell'interfaccia coinvolgendo direttamente un gruppo di medici, che saranno poi chiamati ad utilizzare quotidianamente il prodotto non appena terminato nel suo sviluppo. Trovando difficoltoso organizzare la giornata di test a causa dei molteplici impegni del personale medico, si è preferito optare per una mattinata di presentazione e di prova del progetto in cui fosse comunque possibile avere medici provenienti da differenti realtà ospedaliere piemontesi, affinché loro stessi potessero poi riportare le spiegazioni e i chiarimenti anche ai propri colleghi. Il gruppo era costituito da medici degli ospedali Maria Vittoria, Molinette di Torino, Ospedale Maggiore di Novara e dalla referente del progetto.

La fase di test e di analisi vera e propria è stata preceduta da una breve spiegazione delle differenze strutturali (concentrandosi maggiormente sul consistente cambiamento apportato alla struttura del database) rispetto al precedente applicativo, al fine di far comprendere la complessità del lavoro portato avanti nei mesi precedenti e rendendo maggiormente consapevoli gli stessi utenti del nuovo strumento messo a loro disposizione. La sostanziale differenza, proprio a livello di database risiede nel fatto che, nella nuova versione si è strutturato il progetto sfruttando la flessibilità di un database di tipo relazionale. Un database di tipo relazionale può essere definito come insieme di tabelle e relazioni fra di esse, che permettono la memorizzazione di una serie di dati opportunamente modellati ed organizzati; tale modello pone le sue basi proprio sulle relazioni di tipo matematico (nello specifico il concetto di relazione proviene proprio dalla teoria insiemistica) e la corrispondenza fra i dati è realizzata attraverso i dati stessi (meccanismo basato sui valori). Le tabelle organizzano i dati e le informazioni specifiche in entità che a loro volta sono identificate da un numero fissato di caratteristiche (chiamati attributi) in cui, ad ogni nome viene fatto corrispondere uno specifico valore. Utilizzando un database di questo tipo è possibile avere

massima flessibilità nell'accesso ai dati ed elevata efficienza rispetto ad altre tipologie di basi dati. Sfruttando le potenzialità di questa basi dati è possibile pertanto generare una struttura flessibile e scalabile, nella quale risulta essere facile il recupero dei dati. L'interrogazione del database permette infatti di andare ad ottenere un sottoinsieme limitato di dati che possono a loro volta essere modellati e ripresentati a livello visivo in modo differente.

Al termine della spiegazione più tecnica, si è passati ad una breve panoramica delle diverse entità e funzionalità specifiche; ci si è soffermati in particolar modo sulle entità paziente e medico. In questa fase è stato chiesto ai medici di provare l'applicativo e seguire la breve illustrazione delle funzionalità principali non solo osservando il monitor messo a loro disposizione, ma contemporaneamente provando ad eseguire delle piccole azioni. Ci si è concentrati in particolar modo nell'analisi e nei test sulle operazioni:

- crea nuovo paziente;
- modifica informazioni anagrafiche di un paziente;
- registra nuovo follow-up;
- visualizza follow-up precedente;
- modifica un follow-up;
- organizzazione scheda del paziente;

In questa fase sono state raccolte molte impressioni e numerose sono state le domande relative ad alcune scelte implementative; in particolar modo si è puntato a stimolare i medici ad osservare e concentrarsi nell'analisi di tutte le scelte implementative dipendenti fortemente dal loro ruolo medico. Non essendo infatti operatori del settore e non potendo fare altro che immedesimarsi ed immaginare situazioni, molto spesso anche eccessivamente estreme, sono state operate decisioni di compromesso di cui era bene valutare il loro reale impatto.

Estremamente utile è stato osservare le difficoltà riscontrate ad orientarsi nel nuovo applicativo, in particolare nei medici che hanno utilizzato per un lungo periodo la precedente versione e allo stesso tempo guardare come essi ponessero poca attenzione ai suggerimenti, ai messaggi di conferma e di errore. Inoltre, effettuare i test, ci ha permesso di comprendere come i medici si interfacciassero con l'applicativo; alcuni di essi infatti pur riconoscendo l'importanza di avere uno strumento che permette loro di seguire il decorso clinico ed approfondire gli studi attraverso la raccolta di dati statistici utili, non solo per gli studi clinici ma anche per la scrittura di report, ha anche evidenziato la difficoltà di farne uso all'interno dell'ambiente ospedaliero, relegando il suo utilizzo ai soli ritagli di tempo. Questa osservazione ha posto ancor di più l'attenzione sulla necessità di valutare ogni singola operazione (ad esempio il numero click richiesti per eseguire una particolare funzionalità), ma allo stesso tempo ha confermato le scelte grafiche ed implementative effettuate al fine di garantire una chiara ed ordinata presentazione dei dati.

Al termine della riunione, durata all'incirca due ore e mezza è seguita una piccola fase di debriefing che ha permesso di fare il punto sulle osservazioni fatte dai medici e classificarle suddividendole in due categorie: quelle che non determinano un sostanziale cambiamento

al database e quelle che, al contrario ne determinano una modifica profonda. Operare infatti sul database, ora che la struttura dell'applicativo è pressochè completa risulta essere complesso ed una modifica necessita necessariamente di un'attenta valutazione per poter valutare a pieno il suo impatto sulla restante struttura.

Analizziamo ora le modifiche classificate come poco impattanti sulla struttura del database:

- inserimento automatico della diagnosi e le cardiopatie registrate nella sezione highlights del paziente. I medici hanno apprezzato l'idea di potersi personalizzare, aggiungendo tutte le informazioni ritenute importanti, all'interno della sezione highlights, ma hanno ritenuto necessario che le informazioni relative a diagnosi e cardiopatie fossero sempre presenti ed automaticamente inserite all'interno del tab.
- è stato chiesto di rinominare la sezione terapie in procedure;
- è stato chiesto di accorpare due tab, diagnosi e cardiopatie. I medici hanno ritenuto infatti che queste informazioni non debbano essere separate, in quanto la specifica cardiopatia diagnosticata al paziente concorre alla sua caratterizzazione in ambito clinico; da essa dipendono fortemente le altre sezioni presenti e il suo decorso clinico.
- è stato richiesto di cambiare l'ordinamento dei tab per facilitare l'accesso alle informazioni più importanti. Il nuovo ordine proposto è: highlights, diagnosi e cardiopatie, follow-up, clinica, aritmie, imaging, farmaci, terapie (rinominato procedure), stili di vita;
- i medici si sono interrogati e confrontati durante la fase di test sulle politiche di privacy da tenere in considerazione in merito al trattamento dei dati sensibili del paziente. Hanno pertanto proposto di inserire, all'interno dell'applicativo pronto al download, un documento che deve essere fatto obbligatoriamente firmare e valido per tutti i pazienti di qualunque ospedale, comune e condiviso, al fine di concorrere ad adottare la medesima politica ed uniformare la procedura.

Di seguito invece vengono riportati i cambiamenti ritenuti di maggiore impatto sulla struttura del database:

- è stato proposto di aggiungere un campo che permetta di indicare la data in cui il farmaco è stato terminato, ovvero un'indicazione temporale che permetta di specificare quando ne è stata interrotta la somministrazione. Nella versione dell'applicativo presentato ai medici infatti non è possibile indicare nessuna data di termine assunzione ne eventuali informazioni relative al dosaggio; i medici hanno valutato però che la data di termine non sia un'informazione trascurabile e di conseguenza che abbia bisogno di uno specifico campo, affinché non debba essere inserita all'interno della sezione note.
- continuando l'analisi della sezione farmaci è stato chiesto di mettere in evidenza quelli che sono i medicinali che il paziente sta assumendo, inserendoli all'interno di uno specchio riassuntivo posto all'inizio del tab farmaci. Tale modifica è implementabile solo andando ad inserire due campi data, uno che si riferisca esplicitamente alla data di inizio assunzione ed uno che faccia invece riferimento al termine della prescrizione.

- il personale medico presente nella fase di test si è interrogato a lungo sulla possibilità di spostare la cartella di un paziente da una struttura ad un'altra. Al momento infatti i medici operanti all'interno di un ospedale possono unicamente visionare l'elenco dei pazienti registrati presso quella medesima struttura, ma numerose sono le situazioni in cui un paziente è costretto a dover eseguire una specifica terapia presso un ospedale differente da quello in cui è stato registrato. È stato quindi richiesto un controllo che permetta ai medici di sapere se un dato utente è già stato schedato all'interno del sistema in un'altra struttura in modo da evitare l'inserimento di un duplicato che falserebbe le statistiche. Per ovviare al problema sono state ipotizzate alcune soluzioni, che prevedono l'utilizzo di un codice identificativo univoco per ogni singolo paziente, ma nessuna sembra essere funzionale; fra le possibilità valutate si è ipotizzato di utilizzare il codice fiscale rendendolo un campo obbligatorio in fase di registrazione del paziente.

Ai medici è stato fornito un account e messo a disposizione su di un server interno all'azienda una copia del database originale (datato Luglio 2017) affinché sia loro possibile accedere all'elenco dei pazienti e ad ogni cartella clinica, registrata utilizzando il vecchio applicativo. In questo modo è stata data loro la possibilità di continuare la prova dell'applicazione, familiarizzare con essa, valutare miglioramenti e segnalare eventuali mal funzionamenti all'interno di un ambiente di prova, in cui ogni modifica ai dati non comporta la perdita di dati originali. Si è in fine concordato che eventuali segnalazioni e proposte di miglioramento siano da far pervenire entro metà marzo, allo scopo di procedere all'ultimazione del progetto e valutare tutte le modifiche da implementare.

Capitolo 6

Conclusioni e sviluppi futuri

In questo elaborato è stato analizzato e seguito l'intero processo di sviluppo di un'applicativo Web in ambito clinico. Il progetto iniziato ad ottobre, si è concentrato inizialmente sullo studio dell'interfaccia utente e delle principali linee guida utilizzate all'interno della User Experience andando a realizzare dei mock-up. Solo successivamente si è passati alla vera e propria fase di sviluppo che ha impegnato circa tre mesi (da novembre a gennaio) e che ha richiesto un approfondimento nello studio di diversi linguaggi di programmazione quali HTML, CSS-Less, Javascript e in particolar modo AngularJS. Lo sviluppo è poi proseguito con la fase di testing che ha permesso di valutare eventuali mal funzionamenti o elementi da migliorare con lo scopo di realizzare un prodotto funzionale alle esigenze del personale medico.

La realizzazione di una buona interfaccia è stata un tassello fondamentale di questo progetto; la sua progettazione e la sua successiva implementazione hanno richiesto molto tempo con continue modifiche e rivalutazioni delle diverse scelte implementative al fine di poter realizzare un prodotto finale soddisfacente e vicino alle esigenze dei futuri fruitori. In fase di sviluppo, si è dovuto più volte valutare soluzioni di compromesso, in particolar modo fra necessità progettuali e il bisogno di garantire la piena operabilità dell'applicativo su browser e dispositivi differenti per risoluzione di schermo e/o per dimensione.

Il mese di marzo è stato interamente dedicato ad ultimare il progetto; sono stati infatti messi a punto alcuni dettagli, valutate tutte le implementazioni delle diverse proposte avanzate nella fase di test dai medici, in particolar modo di quelle che comportano una modifica della struttura del database e sono continuate le fasi di testing puntando a coinvolgere profili medici appartenenti a diverse realtà ospedaliere.

Nello specifico, le prime settimane di marzo sono state dedicate, non solo nell'implementazione di tutte le modifiche, ma anche allo sviluppo della landing page.

Con il termine landing page (o pagina di atterraggio) si intende una pagina specificamente strutturata che permetta di presentare il progetto ed invogliare il personale medico, che non è a conoscenza della rete GUCH, ad entrarne a far parte, contattando e raccogliendo informazioni su di essa. Uno degli obiettivi primari che si vogliono raggiungere è quello di far in modo che, ad utilizzare l'applicativo non siano solo poche realtà ospedaliere pubbliche, ma estendere il progetto il più possibile affinché vada a coinvolgere ospedali su tutto il territorio italiano, comprese strutture cliniche private, allo scopo di ampliare

sempre di più il bacino di utenti. Per far sì che sia loro possibile scoprire il progetto si sta dunque valutando la possibilità di realizzare, oltre alla landing page, anche una piccola presentazione scaricabile, che sinteticamente permetta di chiarire gli obiettivi del progetto e come entrare a farne parte; l'intento è quello di stimolare la condivisione all'interno degli ambienti ospedalieri di questo applicativo e fornire ai medici un semplice strumento utile per far conoscere il progetto ai loro colleghi. Realizzare una pagina di atterraggio efficace e funzionale presenta alcune difficoltà ed è importante, al fine di non commettere errori di progettazione, concentrarsi sullo studio dei contenuti e degli elementi grafici da inserire. Per far questo il percorso di sviluppo è partito da una prima fase di analisi di altre landing page, in particolar modo visionando quelle relative a prodotti sanitari o onlus impegnate nel sociale. Si è intrapresa la realizzazione di alcune bozze, riflettendo in particolar modo sui contenuti, sugli elementi visivi e grafici (font, immagini) e iniziando uno studio dei colori da utilizzare per poter redigere una pagina accattivante, ma al contempo in grado di veicolare tutte le informazioni richieste. Solo dopo aver concluso la definizione dei testi e di tutti gli elementi stilistici si potrà realizzare la presentazione in modo che sia possibile richiamarne la medesima veste grafica.

Sviluppi futuri

Per quanto riguarda gli ulteriori sviluppi futuri, l'obiettivo è far in modo che il progetto, l'idea dell'applicativo e le sue funzionalità di analisi dei dati possano essere applicate anche in altri settori sanitari. Lo scopo sarà quindi quello di adattare la struttura base del database relazionale utilizzato in questo progetto ad altri contesti clinici, andando eventualmente ad aggiungere ulteriori specificità e rendere la parte di front-end flessibile e generalizzabile alle eventuali modifiche effettuate sul back-end. Si cercherà quindi di realizzare un database e un'interfaccia utente modulabili rendendo il più facile possibile l'ampliamento delle funzionalità del sistema.

All'interno di un progetto così ampio non mancano sicuramente le possibilità per ulteriori modifiche allo scopo di migliorare progressivamente l'esperienza dell'utente e le sue funzionalità specifiche. Il lavoro svolto, insieme ai feedback positivi e le impressioni raccolte da parte dei medici saranno utilizzate come punto di partenza per portare a termine il lavoro.

Bibliografia

- [1] J. Nielsen, H. Horanger *Web Usability 2.0. L'usabilità che conta*, Apogeo, Milano, 2010.
- [2] J. Nielsen, M.. Tahir *Homepage Usability. 50 siti Web analizzati*, Apogeo, Milano, 2002.
- [3] J. Veen *Web Design. Arte e scienza.*, Apogeo, Milano, 2002.
- [4] S. Krug *Don't make me think. Un approccio di buon senso all'usabilità Web e mobile.*, Tecniche nuove, Milano, 2016.
- [5] S. Krug *Usabilità. Individuare e risolvere i problemi.*, Tecniche nuove, Milano, 2016.

Siti consultati

- [6] Nielsen Norman Group - *www.nngroup.com*
- [7] AngularJS - Documentazione ufficiale - *angularjs.org*
- [8] AngularJS - Documentazione ufficiale - *docs.angularjs.org/guide/introduction*
- [9] Less - Documentazione ufficiale - *lesscss.org*
- [10] Javascript - Documentazione ufficiale - *devdocs.io/javascript*

Ringraziamenti

Grazie all'Azienda CGM Consulting per la fiducia data, per l'aiuto e per avermi dimostrato ancora una volta che "un sorriso fa fare il doppio della strada".

Grazie Elena, una splendida e paziente insegnante.

Grazie a chi, con pazienza, mi ha aiutato a correggere ogni parola, virgola ed errore di questa tesi.

Grazie ai fratelli e al loro indispensabile aiuto.

Grazie a chi è stato di supporto.

Grazie agli Amici.