

ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA  
CAMPUS DI CESENA

---

DIPARTIMENTO DI INFORMATICA – SCIENZA E INGEGNERIA  
Corso di Laurea in Ingegneria e Scienze Informatiche

SVILUPPO DI UN SISTEMA DI  
TELEMEDICINA PER LA GESTIONE  
DELL'IPERTENSIONE ARTERIOSA

*Elaborato in*  
SISTEMI EMBEDDED ED INTERNET OF THINGS

*Relatore*  
Prof. ALESSANDRO RICCI

*Presentata da*  
RICCARDO OMICCIOLI

*Corelatore*  
Prof.ssa SARA MONTAGNA

Anno Accademico 2020 – 2021



*“If you get up in the morning and think the future is going to be better, it is a bright day. Otherwise, it’s not.” - Elon Musk*



# Indice

<b>Introduzione</b>	<b>vii</b>
<b>1 Sistemi telemedicina e Chatbot nel dominio medico</b>	<b>1</b>
1.1 Sistemi di telemedicina e chatbot . . . . .	1
1.2 Chatbot nel mondo healthcare . . . . .	3
1.3 Il problema dell'ipertensione . . . . .	5
1.4 Sistemi di telemedicina per la gestione dell'ipertensione arteriosa	6
<b>2 Progetto AI4Hypertension</b>	<b>9</b>
2.1 Analisi dei requisiti . . . . .	9
2.2 Architettura del sistema . . . . .	11
2.3 Tecnologie utilizzate . . . . .	14
2.4 Sviluppo . . . . .	15
<b>3 Hypertension Chatbot</b>	<b>17</b>
3.1 Analisi dei requisiti . . . . .	17
3.2 Progettazione e tecnologie scelte . . . . .	18
3.3 Implementazione . . . . .	22
<b>4 Hypertension Website</b>	<b>29</b>
4.1 Requisiti e tecnologie scelte . . . . .	29
4.2 Sviluppo e dettagli implementativi . . . . .	30
<b>5 Discussioni finali sul sistema</b>	<b>43</b>
5.1 Considerazioni sul sistema . . . . .	43
5.2 TensioBot . . . . .	44
<b>Conclusioni</b>	<b>49</b>
<b>Ringraziamenti</b>	<b>51</b>



# Introduzione

E-health è un termine recente che si riferisce all'uso della tecnologia per migliorare la salute e il benessere umano attraverso una vasta gamma di strumenti informatici e di telecomunicazione che sono ora facilmente a disposizione grazie ad un rapido ed intenso avanzamento dei settori elettronico-informatico negli ultimi decenni. Uno dei campi più sviluppati dell'e-health è quello della telemedicina la quale include tutte le tecniche che permettono la cura e l'assistenza al paziente a distanza con l'idea di riuscire, quando possibile, a muovere informazioni invece delle persone in modo da fornire un servizio più rapido, efficiente ed accessibile. Oltre a permettere al medico di svolgere meglio il proprio lavoro, la telemedicina favorisce quello che viene chiamato patient engagement, ovvero il coinvolgimento del paziente facendolo passare da attore passivo nel suo percorso di cura ad elemento attivo rendendolo più autonomo e consapevole nella gestione del proprio stile di vita. Le capacità della telemedicina sono già state dimostrate in passato con una serie di numerosi successi e, proprio negli ultimi anni, sta acquisendo un ulteriore strumento utile soprattutto per abilitare il coinvolgimento del paziente, i chatbot. I chatbot sono programmi informatici che, facendo spesso uso di tecniche di intelligenza artificiale, assistono un utente fornendogli informazioni o aiutandolo nel raggiungimento di uno scopo. L'obiettivo di questo progetto è proprio quello di utilizzare un chatbot e altre tecniche di telemedicina per la gestione dell'ipertensione arteriosa, una delle malattie più diffuse al mondo, ed anche, purtroppo, una tra le più trascurate. L'applicazione di tecnologie come quella dei chatbot permetterebbe di creare uno strumento economico ed efficace per stimolare il paziente ad effettuare nuove misurazioni e alla corretta assunzione della propria terapia facendo uso di una conversazione motivazionale e un sistema di notifiche attive. Partendo da uno studio generale del contesto applicativo e servendosi di un prototipo di chatbot già sviluppato, il contributo di questo progetto è stato quello di migliorare le parti già implementate, rendendole più affidabili e aggiungendo nuovi requisiti elaborati in collaborazione con esperti di questo settore medico. Oltre al potenziamento del sistema già esistente è stata sviluppata una nuova piattaforma web fondamentale per aiutare il personale sanitario nella supervisione e cura dei propri pazienti. L'organizzazione di que-

sto documento cerca di ricalcare quanto più possibile il percorso fatto durante il progetto partendo in primo luogo da un'analisi relativa alle tecnologie già sviluppate nel settore medico e dei problemi che questa malattia presenta per cercare di progettare la soluzione più efficace. Verrà poi esposto il funzionamento della parte già implementata che è stata oggetto di studio e di successivo utilizzo come fondamento per i nuovi sviluppi, per poi passare a discutere delle modifiche e delle nuove aggiunte fatte al sistema spiegando le decisioni prese per costruire una soluzione che si adattasse alle esigenze degli utilizzatori. Infine, dopo un riepilogo degli obiettivi raggiunti, verrà fatta un'analisi critica sul sistema cercando di capire la correttezza delle scelte effettuate e gli aspetti che invece richiederebbero un miglioramento futuro confrontando inoltre i risultati ottenuti con soluzioni simili già sviluppate. In particolare il flusso del progetto in base agli argomenti trattati o alla loro importanza è stato diviso in cinque capitoli nel seguente modo:

- Nel primo capitolo verranno introdotti quelli che sono i sistemi di telemedicina e in particolare i chatbot, spiegandone i principi generali e di come essi siano attualmente diffusi nel settore medico. Saranno poi analizzate le problematiche relative all'ipertensione arteriosa in modo da comprendere i problemi relativi a tale malattia e di quali azioni sono necessarie per migliorare la situazione. Verrà anche evidenziato un caso particolare di un sistema di telemedicina che ha portato ad un significativo miglioramento nel trattamento dell'ipertensione dimostrando di come utilizzando tecnologie in questo settore sia possibile ottenere ottimi risultati.
- Nel secondo capitolo verrà spiegata la struttura e il funzionamento del progetto AI4Hypertension che è stato alla base del lavoro svolto e sul quale è stato comunque dedicato diverso tempo per capirne il funzionamento e i possibili miglioramenti da sviluppare.
- Il terzo capitolo sarà dedicato invece ad un riepilogo del lavoro svolto da quando è stato preso in carico il progetto e delle successive modifiche che sono state effettuate in seguito a confronti con uno dei medici responsabili del progetto.
- Il quarto capitolo riguarda lo sviluppo di una piattaforma web pensata soprattutto per assistere, oltre al paziente, il medico che si prende cura di esso durante lo svolgimento del proprio lavoro. Verranno esposti i problemi che si volevano risolvere e gli strumenti che sono stati implementati per rispondere a tali esigenze.
- Nel quinto capitolo verrà riassunto il lavoro svolto e gli obiettivi raggiunti prendendo in analisi un progetto molto simile a quello qui sviluppato per



capire le differenze tra essi, i risultati che è possibile ottenere e di come le decisioni prese in questo progetto presentino elementi in comune o in contrasto rispetto alle altre soluzioni disponibili sviluppate per affrontare lo stesso problema.



# Capitolo 1

## Sistemi telemedicina e Chatbot nel dominio medico

In questo capitolo verrà spiegato cosa sono i sistemi di telemedicina e che ruolo svolgono nel dominio medico per migliorare la qualità dei servizi forniti. Saranno anche introdotti i chatbot, una delle tecnologie fondamentali di questo progetto e verrà analizzato di come essi, negli ultimi anni, si stiano inserendo sempre più in vari campi della medicina come nuovo strumento funzionale a supporto sia del paziente che del medico. Il tema di questo progetto rimane comunque l'applicazione di queste tecnologie per la gestione dell'ipertensione arteriosa e quindi, dopo un riassunto delle caratteristiche di questa malattia, verranno riportati alcuni esempi dei lavori già svolti e dei risultati positivi che si possono ottenere.

### 1.1 Sistemi di telemedicina e chatbot

“La telemedicina è la fornitura di servizi sanitari, attraverso l'uso delle tecnologie dell'informazione e della comunicazione, in situazioni in cui l'operatore sanitario e il paziente (o due operatori sanitari) si trovano in posti diversi. Essa implica la trasmissione sicura di informazioni e dati medici attraverso testi, suoni, immagini o altri formati necessari per la prevenzione, diagnosi, trattamento e follow-up dei pazienti.”

Questa è la definizione di telemedicina data dalla commissione europea in un documento da loro pubblicato riguardo ai benefici della telemedicina per il paziente, il sistema sanitario e la società in generale [1]. Come si può intuire, sotto questa definizione rientrano quindi un vasto numero di tecnologie e di loro utilizzi che possono andare da una semplice teleconsulenza tra il paziente

e il medico ad un avanzatissimo sistema di telechirurgia. L'interesse di questa tesi però ricade sull'uso delle tecnologie per la raccolta dati, monitoraggio e supporto del paziente attraverso strumenti che permettano di seguire le loro condizioni con una frequenza maggiore, di essere informati tempestivamente delle situazioni di emergenza e di controllare i pazienti senza che essi debbano recarsi presso le strutture mediche cosa che, in questo periodo di emergenza sanitaria per l'epidemia di COVID-19, è diventata particolarmente problematica. Ovviamente i sistemi di telemedicina non hanno lo scopo di sostituire completamente le visite mediche soprattutto quando esse trattano casi delicati o richiedono l'uso di strumenti specializzati ma si pongono l'obiettivo di rendere più efficaci le visite dove è sufficiente una semplice conferenza con il medico o un monitoraggio costante della propria condizione. Una tra le tecnologie più promettenti per applicazioni di telemedicina è quella dei chatbot che negli ultimi anni hanno subito una rapida diffusione nelle applicazioni di uso quotidiano ma, prima di analizzare come essi vengano usati nel campo medico e dei vantaggi che possono portare, è bene comprendere cosa essi siano e quello che sono in grado di fare. Un chatbot è un programma creato per simulare una conversazione con un umano nel modo più naturale possibile facendo spesso uso di intelligenze artificiali per interpretare gli input forniti da un utente e, dato il loro scopo di conversare con una persona, vengono spesso chiamati anche agenti conversazionali. Esistono varie tipologie di chatbot e, mentre alcuni di loro si limitano ad interpretare solamente una serie di comandi predefiniti, altri sono capaci di interpretare le richieste anche se queste vengono presentate con delle variazioni tra loro o addirittura in modi che il chatbot non aveva mai visto in passato. Alcuni di essi possono interagire con gli utenti solo attraverso parole scritte mentre altri sono in grado di interpretare anche dialoghi parlati o entrambe le cose ma, solo i chatbot più avanzati, sono in grado di scegliere in modo autonomo come formulare risposte mentre, generalmente, in quelli più semplici le risposte vengono decise a priori dagli sviluppatori. La capacità di un chatbot di poter stimare e sviluppare una risposta in modo autonomo pone spesso numerose sfide dove, con l'aumento di complessità del sistema, si rischia di andare in contro a quello che nel settore delle intelligenze artificiali viene chiamato "effetto scatola nera", ovvero, quando i ragionamenti dietro la scelta delle risposte dei chatbot diventano troppo complessi per essere verificati e validati dagli sviluppatori del sistema. Nonostante le importanti ricerche effettuate non si è raggiunto ancora l'obiettivo di avere un chatbot con le stesse capacità di comprensione di un essere umano, tuttavia, il loro utilizzo si è già rivelato generalmente utile anche nel campo healthcare dove sono stati ottenuti risultati positivi. E' possibile dividerli in tre grandi categorie in base alle loro capacità le quali spesso vengono successivamente divise in sotto categorie a seconda, per esempio, del metodo di interazione con esso, della piattaforma

utilizzata e dello scopo per cui viene utilizzato.

**Scripted / rule-based** Questa è la tipologia di chatbot più semplice dato che sono in grado di riconoscere solamente comandi specifici che, spesso, vengono attivati tramite l'uso di menu personalizzati o pulsanti e soddisfano la richiesta di un utente facendogli compiere dei passi successivi in quello che può essere considerato l'albero decisionale del programma. Questa tipologia di chatbot la si trova maggiormente nei servizi di supporto offerti ai clienti dove, con una serie di domande e selezioni successive, è possibile rispondere ad alcune delle richieste più frequenti fatte dagli utenti senza la necessità di intervento umano.

**Natural Language Processing (NLP) Chatbots** Chiamati anche keyword recognition-based chatbots essi fanno uso di intelligenze artificiali per determinare le richieste degli utenti e agire di conseguenza. Spesso è possibile interagirci attraverso testo scritto o vocalmente anche se a volte vengono accompagnati da menu/pulsanti per fornire all'utente un metodo alternativo per l'interazione nel caso il riconoscimento non fornisca i risultati desiderati. Anche essi, come quelli che utilizzano menu e pulsanti personalizzati, sono adatti per rispondere ad un numero limitato di richieste predeterminate.

**Chatbots contestuali** Sono la categoria più avanzata ed utilizzano una combinazione di intelligenza artificiale e machine learning per cercare di imparare a rispondere alle varie richieste generali che un utente può porgli invece di basarsi su una serie di risposte predeterminate. A differenza delle altre categorie, le interazioni passate con l'utente possono essere usate per migliorare e personalizzare le interazioni successive con esso.

## 1.2 Chatbot nel mondo healthcare

Grazie all'incredibile e rapida diffusione degli smartphone e alla crescita esponenziale nell'uso di chatbot negli ultimi anni, il settore dell'healthcare si è ritrovato con un prezioso strumento da utilizzare per migliorare la qualità dei propri servizi. Uno dei principali vantaggi degli agenti conversazionali è la loro capacità di poter essere utilizzati con semplicità da moltissime fasce della popolazione sia essa più giovane che anziana richiedendo solo conoscenze basilari nell'uso della tecnologia. Nel campo medico esistono già da alcuni anni svariati chatbot con diversi obiettivi che vanno dalla semplice educazione dell'utente, al monitoraggio, trattamento e diagnosi di malattie. Secondo alcune ricerche il 60% delle visite mediche vengono effettuate per malattie semplici

che, nell'80% dei casi, possono essere curate in casa anche senza l'intervento di un medico [8] e questo significa che, se si riescono ad utilizzare chatbot per risolvere questi casi, il personale medico potrà dedicare maggior tempo e risorse a casi più complessi e delicati. Inoltre, alcuni utenti, hanno riportato di trovarsi più a loro agio nel fare domande ad un chatbot o ad interagire con un medico attraverso strumenti informatici soprattutto quando la natura della visita riguarda aspetti che possono mettere a disagio il paziente. Anche dei semplici chatbots che riescano ad educare gli utenti rispondendo alle domande più frequenti su un certo tema hanno la loro importanza e la loro efficacia è stata comprovata da alcuni studi. Un esempio lo si ha in una ricerca effettuata su 927 donne riguardo al tema della fertilità e contraccezione dove, seppur con risultati non ottimali, si è potuto notare un notevole miglioramento delle conoscenze sul tema nei partecipanti che hanno utilizzato un chatbot per educarsi rispetto al gruppo di controllo [7].

*Conversational Agents in Health Care: Scoping Review and Conceptual Analysis* è un interessante articolo che fornisce utili statistiche prese da un'analisi effettuata su 47 report di applicazione di agenti conversazionali riguardanti l'ambito healthcare su obiettivi diversi come: trattamento e monitoraggio, servizi di supporto, educazione, studio dei comportamenti e diagnosi. Analizzando i risultati e le modalità dei test effettuati è possibile farsi un'idea generale di come vengono utilizzati i chatbot nel mondo health care, dei risultati ottenuti da essi e delle tecnologie utilizzate per svilupparli ed interagire con gli utenti. Negli studi considerati l'interazione con gli agenti conversazionali veniva effettuata per la maggior parte tramite applicazioni smartphone (23 su 47), tramite assistenti integrati (come, ad esempio, Siri, Google assistant e Alexa) (6 su 47) oppure tramite applicazioni web based (5 su 47). La piattaforma di messaggistica Telegram è stata utilizzata in 2 dei report considerati ma è comunque degna di nota dato che, come verrà spiegato in seguito, è stata utilizzata in questo progetto per la conversazione con l'utente. Quasi la metà (19) degli agenti conversazionali utilizzati accettava input dall'utente senza vincoli mentre una buona parte (10) necessitavano di input predeterminati. Le risposte venivano fornite in gran parte (30) solamente tramite testo e solo una piccola parte attraverso testo e voce o solamente voce (rispettivamente 5 e 4). Tutti gli agenti conversazionali considerati erano goal oriented, ovvero utilizzati per una specifica funzione e con degli obiettivi ben determinati e, in particolare, metà di essi (23) erano orientati al raggiungimento di obiettivi a breve termine quali fornire all'utente risposte o suggerimenti per educarlo sugli argomenti trattati. L'altra metà (22) è stata utilizzata con in mente obiettivi a lungo termine come supporto all'utente durante il percorso per smettere di fumare, affrontare il problema dell'obesità, per affrontare problemi psicologici e comportamentali. Secondo i report pubblicati la maggior parte degli agenti conversazionali sono

quindi forniti tramite applicazioni specializzate utilizzando messaggi di testo per l'interazione con l'utente e servendosi di tecniche di machine learning per rispondere alle richieste. Molti dei risultati pubblicati dai report si basano su casi di studio che considerano un piccolo numero di utenti ma i risultati sono stati comunque generalmente positivi considerando l'efficacia, l'accuratezza e il grado di accettazione nell'uso dei chatbot. Anche se l'uso di agenti conversazionali nel mondo healthcare si sta sviluppando solamente negli ultimi anni il loro uso si sta velocemente spostando dal fornire semplici risposte al raggiungimento di obiettivi a lungo termine come la gestione di malattie croniche. Come già detto la maggior parte dei chatbot considerati nell'analisi utilizza testi sia per rispondere agli utenti che per ottenere informazioni da essi anche se, l'interazione vocale, potrebbe rivelarsi essere un modo migliore per interagire con l'utente soprattutto se si considera la parte più anziana della popolazione che potrebbe avere problemi nel leggere testi piccoli o utilizzare le tastiere degli smartphone per digitare il testo da inserire. Dalla ricerca viene anche fuori che l'uso di una app specializzata porta generalmente ad un drop-out maggiore nell'uso del sistema, dovendo l'utente installare e aggiornare regolarmente l'applicazione, cosa che sembra presentarsi in maniera molto minore nei chatbot che utilizzano piattaforme di messaggistica già conosciute ed utilizzate dagli utenti. Manca ancora, purtroppo, un sistema ben definito per valutare gli agenti conversazionali applicati al dominio healthcare soprattutto per quanto riguarda i loro possibili effetti negativi e, specialmente per quei chatbot che fanno uso di machine learning per decidere le risposte da dare all'utente, è fondamentale accertarsi che esso sia in grado di gestire correttamente la situazione e non fornire informazioni sbagliate al paziente mettendolo a rischio. Anche le questioni relative alla privacy degli utilizzatori sono molto delicate in questi usi degli agenti conversazionali dato che i pazienti condividono con essi informazioni sensibili spesso necessarie per l'utilizzo delle funzionalità messe a disposizione. Considerando questi aspetti non ancora del tutto definiti al momento la soluzione ottimale sembra quella di utilizzare i chatbot in un sistema ibrido dove essi vengono affiancati ai servizi già esistenti anche se, con sviluppi futuri e linee guida più definite, gli agenti conversazionali potrebbero, almeno per situazioni meno delicate, assumere un ruolo molto simile a quello di un vero e proprio professionista del settore medico [11].

### 1.3 Il problema dell'ipertensione

L'ipertensione arteriosa è un disturbo della pressione sanguigna caratterizzato da una frequente ed eccessiva forza esercitata dal sangue sulle pareti arteriose rispetto ai valori fisiologici normali. Questo comporta un lavoro

aggiuntivo effettuato dal cuore durante il suo ciclo di contrazione (sistole) e rilassamento (diastole) e aumenta notevolmente il fattore di rischio per infarti, ictus, insufficienza cardiaca e aneurismi. Quasi la totalità dei casi di ipertensione sono classificati come “ipertensione primaria”, ovvero, la condizione di ipertensione si verifica in modo indipendente e non è causata dalla presenza di altre malattie. Anche se spesso è necessario intervenire attraverso un trattamento farmacologico, una dieta equilibrata e uno stile di vita sano possono notevolmente contribuire al controllo della pressione sanguigna e alla riduzione del rischio di presentare complicazioni nel proprio stato di salute. Secondo alcune stime circa un quarto della popolazione mondiale soffre di ipertensione dei quali più di metà non è consapevole della sua condizione in quanto raramente sono presenti dei sintomi evidenti. Sicuramente sia l’ampia diffusione che l’inconsapevolezza e il mancato trattamento sono fattori importanti che hanno fatto diventare l’ipertensione la causa principale di mortalità cardiovascolare. Secondo la SIIA (Società Italiana dell’Ipertensione Arteriosa), in Italia, mediamente il 33% degli uomini e il 31% delle donne soffre di ipertensione arteriosa mentre il 19% degli uomini e il 14% delle donne presenta valori di pressione che indicano una condizione di possibile rischio di ipertensione [5]. Dato che la sensibilità all’ipertensione aumenta esponenzialmente con l’età, visto il generale aumento dell’aspettativa di vita e un invecchiamento medio della popolazione italiana si capisce come sia fondamentale potenziare le azioni preventive e agire per ridurre i fattori di rischio. Le malattie cardio-cerebrovascolari rappresentano la prima causa di morte nel mondo occidentale e, a seguito di un evento acuto, fatale nel 40% dei casi, si hanno comunque complicanze e notevoli ripercussioni sulla qualità della vita del soggetto [4]. L’assenza di sintomi è il motivo per cui effettuare misurazioni periodiche e visualizzare l’andamento medio delle proprie misure è fondamentale per monitorare la propria condizione, agire tempestivamente quando necessario e fornire un riscontro sull’efficacia di cambiamenti sulla dieta o sullo stile di vita quotidiano. La mancata assunzione da parte del paziente della propria terapia e la dimenticanza nell’effettuare misurazioni sono i principali problemi che i medici riscontrano oltre, in alcuni casi, all’utilizzo di metodi inadeguati o inefficaci per visionare i dati dei propri pazienti.

## 1.4 Sistemi di telemedicina per la gestione dell’ipertensione arteriosa

Vi sono anche esempi di sistemi di telemedicina utilizzati nello specifico per il controllo dell’ipertensione dove un sistema di invio automatico delle misurazioni e una linea telefonica dedicata venivano messi a disposizione di un



gruppo di pazienti selezionati. Tramite appuntamenti telefonici venivano chieste informazioni generali al paziente circa la sua condizione e venivano forniti consigli nel caso fosse necessario migliorare alcuni aspetti come la sua dieta o l'esercizio fisico. I pazienti inoltre potevano anche contattare il personale medico in qualsiasi momento semplicemente facendo una chiamata al servizio messo a disposizione nel caso essi presentavano dei sintomi o avevano delle domande da porre. I dati raccolti da questi appuntamenti telefonici insieme a quelli rilevati con un dispositivo per la misurazione della pressione in casa venivano analizzati da specialisti e, sulla base dei loro riscontri, le terapie dei pazienti potevano essere modificate se necessario. Al termine di tale servizio i pazienti appartenenti al gruppo di controllo hanno riscontrato valori di pressione mediamente più bassi del 4,5% circa sia per quanto riguarda la pressione sistolica che diastolica mentre, il gruppo che ha utilizzato i servizi di telemedicina, ha ottenuto una riduzione pari a circa il 15% sia per la sistolica che per la diastolica. Questo significa che, dei 94 pazienti appartenenti al gruppo di controllo, l'81% e il 62% presentavano ancora valori di pressione superiori al limite per quanto riguarda rispettivamente i valori di pressione sistolica e diastolica. Dei 74 pazienti appartenenti al gruppo che faceva uso del sistema di telemedicina solo il 26% per la sistolica e 8% per la diastolica presentava alla fine del servizio valori fuori dalla norma. Questo miglioramento significativo lo si può attribuire a cambiamenti più frequenti nel gruppo di intervento relativamente ai medicinali da loro utilizzati dove il 70% di essi si è visto modificare la propria terapia a seguito dei riscontri avuti durante le interazioni telefoniche. Al contrario solo il 23% dei pazienti del gruppo di controllo ha subito modifiche alla propria terapia a seguito delle visite mediche da loro effettuate [2]. A seguito di alcune ricerche non sono stati trovati molti esempi di chatbot dedicati alla gestione dell'ipertensione arteriosa e in effetti solamente due esempi sono stati scoperti. Il primo, sviluppato e testato per oltre due anni tra il 2018 e il 2020, viene chiamato TensioBot ed è nato da un progetto svolto dall'università dei paesi Baschi portato avanti da tre ricercatori aderenti ai dipartimenti di fisiologia, linguaggi e sistemi computazionali e medicina. Si tratta di un sistema che utilizza la piattaforma Telegram per l'interazione con il chatbot attraverso una serie di menu e pulsanti messi a disposizione dell'utente e con i quali può scegliere quale funzionalità utilizzare tra quelle messe a disposizione. Dato che TensioBot ha diversi punti in comune con il chatbot sviluppato durante questo progetto una descrizione dettagliata è stata inserita a seguito della spiegazione del progetto in modo da analizzarne similitudini e punti di differenza e studiare le loro osservazioni in funzione dei risultati da loro ottenuti durante le prove. Il secondo tra i chatbot per l'ipertensione è MEGI, sviluppato durante la collaborazione tra una start-up e una clinica entrambe di origine croate, il quale è entrato in uso dopo un periodo di prova

della durata di un anno svolto su alcuni pazienti. Anche per MEGI è stato scelto di utilizzare Telegram come piattaforma grazie alla sua flessibilità, sicurezza e facilità di sviluppo. Non sono stati trovati molti dettagli relativi al chatbot MEGI ma sono presenti comunque alcuni commenti riguardo alla sua utilità e, in particolare, è stato fatto notare come i medici siano stati in grado, grazie al suo utilizzo, di apportare modifiche alle terapie dei pazienti in sole tre settimane invece dei sei mesi richiesti tra due visite successive [9].

# Capitolo 2

## Progetto AI4Hypertension

In collaborazione con il dottor Martino Pengo dell’Istituto Auxologico di Milano è stato sviluppato, durante la tesi dello studente Matteo Castellucci un chatbot pensato per l’interazione con i pazienti affetti da ipertensione [3]. Durante questa collaborazione sono state definite le specifiche del sistema e le esigenze a cui avrebbe dovuto rispondere ed è stato poi sviluppato un prototipo funzionante per poter effettuare delle prove interne e validare i risultati ottenuti. Il tempo trascorso a studiare il funzionamento del progetto AI4Hypertension è poi risultato in una serie di informazioni utili per capire il funzionamento generale del sistema e per valutare in che modo fosse possibile migliorare il chatbot aggiungendo nuove funzionalità. In questo capitolo verrà esposto brevemente il lavoro svolto da Matteo Castellucci che è stato poi studiato e modificato per adattarsi alle nuove esigenze che verranno esposte successivamente.

### 2.1 Analisi dei requisiti

Il progetto AI4Hypertension è nato con lo scopo di implementare un chatbot user-friendly che possa essere usato da utenti di varie fasce di età per assisterli nella gestione della loro condizione di ipertensione ricordandogli di assumere le proprie terapie e di effettuare misurazioni, fornendo informazioni generali e memorizzando i valori di pressione inseriti dagli utenti per poterne poi visualizzare l’andamento nel tempo. Per interagire con il chatbot è stato scelto di utilizzare la piattaforma di messaggistica Telegram, una delle principali applicazioni del suo genere che, secondo statistiche del 2020, vanta ben 500 milioni di utenti [10] e che da sempre ha dato molta importanza allo sviluppo dei chatbot sulla piattaforma. Il chatbot doveva essere in grado di rispondere in modo adeguato ad un insieme limitato di richieste fatte attraverso messaggi in linguaggio naturale per poi elaborare e fornire la risposta

corretta in un tempo ragionevolmente breve. I messaggi che il sistema invia agli utenti non devono essere ambigui e devono guidare quanto più possibile l'utente durante l'utilizzo delle funzioni del sistema. Oltre a messaggi contenenti informazioni generali su come misurare in modo corretto i propri valori di pressione il sistema doveva essere in grado di registrare le misurazioni degli utenti memorizzando i valori di pressione sistolica, diastolica ed opzionalmente anche il valore della frequenza cardiaca. Il paziente doveva essere in grado di inserire nuove misure semplicemente scrivendo un messaggio contenente in linguaggio naturale i dati della misurazione e il sistema doveva guidare l'utente durante la procedura e informarlo su come inserire correttamente i valori in caso di errori che impedivano la corretta interpretazione. Dopo ogni misura inserita il sistema avrebbe dovuto notificare l'utente che l'operazione era andata a buon fine e, nel caso il valore inserito fosse sopra i valori limite di un paziente iperteso, informarlo di contattare il proprio medico o rassicurarlo nel caso la sua media risulti comunque nei valori corretti. Il sistema doveva essere anche in grado di poter fornire la media giornaliera, settimanale o mensile di un utente e di visualizzare tutte le misure inserite insieme ad un grafico con tutti i dati raccolti. Un'altra funzione molto importante del chatbot era quella di mandare notifiche al paziente per ricordargli di prendere la propria terapia o di effettuare nuove misure nel caso non siano stati inseriti nuovi valori negli ultimi giorni. Le notifiche dovevano essere inviate dopo due giorni dall'ultima misurazione, dopo tre giorni dalla prima notifica e infine dopo quattro giorni dall'ultima notifica in modo da non infastidire l'utente con un numero elevato di promemoria. E' comunque molto importante ricordare all'utente di inserire nuove misure in quanto, come detto precedentemente, solo attraverso uno storico dei valori di pressione è possibile studiare l'andamento della condizione di ipertensione e capire se la terapia prescritta sia adatta al paziente e stia dando i risultati attesi.

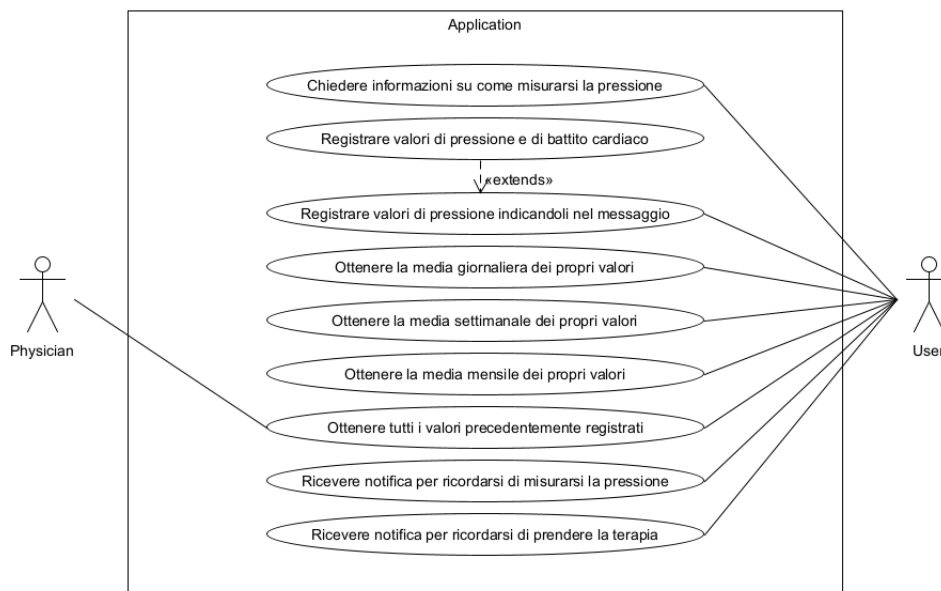


Figura 2.1: Diagramma dei casi d’uso del progetto AI4Hypertension

## 2.2 Architettura del sistema

In figura 2.2 è riportato il diagramma delle classi che descrive la struttura del sistema sviluppato durante il progetto AI4Hypertension. Il progetto prevedeva anche una parte di riconoscimento dei valori di pressione da immagini scattate allo schermo delle apparecchiature di misura della pressione ma tale funzionalità è stata poi rimossa in quanto non sufficientemente affidabile nel riconoscimento e valutata in seguito come non importante rispetto alle altre implementate. Il diagramma originale è stato quindi modificato rimuovendo ogni elemento relativo al riconoscimento tramite immagini per rispecchiare l’effettiva struttura del progetto utilizzato.

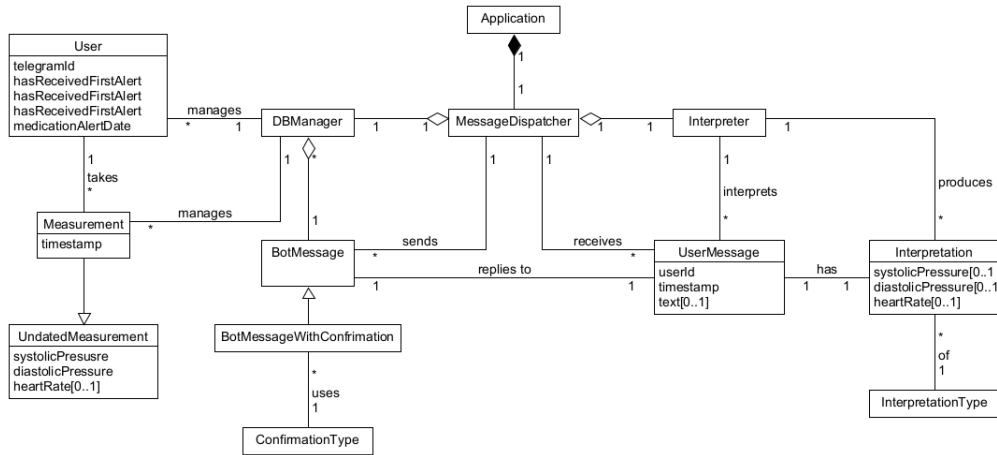


Figura 2.2: Diagramma delle classi del progetto AI4Hypertension

Il componente “Application” rappresenta la radice del sistema dato che è il primo che viene messo in esecuzione e ha lo scopo principale di costruire e avviare tutti gli altri componenti necessari. Tra i vari componenti quello che probabilmente è il più importante per l’applicazione “MessageDispatcher” che, come è intuibile dal nome, riceve tutti i messaggi provenienti dagli utenti per poi gestirli e fornire a loro una risposta. MessageDispatcher a sua volta dispone di due componenti che sono “DBManager” usato per l’interazione con il database e “Interpreter” che si occupa dell’interpretazione dei messaggi testuali inviati dagli utenti. Ogni volta che il sistema riceve un messaggio viene costruito uno “UserMessage” il quale modella un messaggio inviato dall’utente. Un messaggio Telegram contiene al suo interno numerosi parametri ma, per gli scopi del sistema, sono sufficienti solamente quelli presenti in UserMessage dove vengono riportati il numero identificativo Telegram dell’utente, la data e l’ora del messaggio e il testo inviato. Lo UserMessage viene poi passato all’ “Interpreter” per essere effettivamente interpretato ma, prima di procedere nell’analisi del sistema, è utile soffermarsi su come effettivamente un messaggio venga interpretato. Fornendo all’intelligenza artificiale il testo inviato dall’utente essa stimerà, con un certo grado di precisione, l’ “intent” corrispondente ovvero, come suggerisce la parola, all’intenzione che l’utente voleva esprimere attraverso il messaggio. Trattandosi di una intelligenza artificiale “semplice” e specifica per il dominio di applicazione il numero di intent riconoscibili da essa sarà limitato e deciso dal programmatore ed è possibile considerare gli intent come rappresentazione delle funzioni che il sistema mette a disposizione e che l’utente vuole utilizzare. A differenza delle intelligenze artificiali più evolute, quella utilizzata in questo sistema non restituisce direttamente la risposta da fornire all’utente ma fornisce invece l’intent più probabile

a seconda del messaggio che si vede arrivare ed è poi compito del programmatore associare ad ogni intent la funzione richiesta e rispondere all'utente con un messaggio predeterminato. E' possibile notare nel diagramma come ogni "Interpretation" abbia associato un "InterpretationType" ovvero un tipo di interpretazione che, a livello effettivo, rappresenta il tipo di intent restituito dall'intelligenza artificiale. Opzionalmente, nel caso l'utente stia inviando una nuova misura, l'interpretazione presenta dei parametri opportuni per memorizzare i dati necessari che, nel progetto AI4Hypertension, venivano direttamente estratti e restituiti al programma dall'intelligenza artificiale. La risposta che il sistema invia all'utente è rappresentata nello schema da "BotMessage" dove può trattarsi di una semplice risposta o di un "BotMessageWithConfirmation", ovvero di un messaggio che necessita di conferma. I messaggi che necessitano di conferma vengono usati, per esempio, quando un utente inserisce una nuova misura e si attende quindi, dopo un riepilogo dei valori inseriti, la conferma che i dati siano corretti prima di essere memorizzati. Il compito di ricordarsi da quali utenti si sta attendendo una conferma è svolto anch'esso dal Message-Dispatcher. DBManager è il componente che viene utilizzato ogni volta che è necessario interagire con il database per recuperare dati o inserirne di nuovi e, come si può notare dal diagramma, gestisce i componenti "User" e "Measurement" derivati secondo l'organizzazione degli utenti e delle misure all'interno del database utilizzato. Per gli User vengono memorizzati, oltre al proprio identificativo Telegram, informazioni su quali notifiche sono state inviate ad esso e, in particolare, quali avvisi per l'inserimento di nuove misure ha ricevuto e la data dell'ultima notifica per quanto riguarda la terapia che l'utente deve prendere. Measurement contiene intuitivamente tutte le informazioni necessarie a memorizzare una misura come l'identificativo utente, la data, i valori di pressione e opzionalmente di frequenza cardiaca. In figura 2.3 è riportato il diagramma che ricostruisce l'organizzazione del database utilizzato dal sistema e si può notare facilmente la corrispondenza con quanto appena detto riguardo al diagramma del sistema.

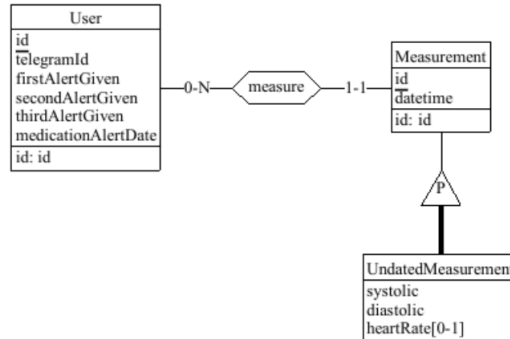


Figura 2.3: Diagramma Entity-Relationship del database utilizzato dal sistema

## 2.3 Tecnologie utilizzate

Come già detto precedentemente è stata utilizzata l'app Telegram per comunicare con il bot e questa scelta ha grandi vantaggi come la semplicità nell'utilizzo, la diffusione notevole dell'applicazione, uno sviluppo rapido e nessuna approvazione necessaria per rendere disponibile il sistema tramite gli app stores. Telegram ha anche a disposizione una gestione dei bot molto più matura e semplice da utilizzare rispetto ad applicazioni come Whatsapp, dove i chatbot sono meno diffusi e meno sviluppati e Facebook Messenger dove è necessario registrare la propria applicazione, avere un account Facebook ed utilizzare servizi che potrebbero scoraggiare gli utenti all'uso per questioni di privacy. Il programma del chatbot è sviluppato in Java che, oltre alla compatibilità con la maggior parte dei sistemi operativi, ha anche dato accesso ad una numerosa quantità di librerie e ha reso la comunicazione con le API di Telegram molto semplice. Con la libreria VertX per Java è stato realizzato il ciclo di esecuzione di base del programma e la creazione e gestione dei web client e web server per le comunicazioni con i vari servizi utilizzati. Per interpretare i messaggi testuali inviati dagli utenti è stato utilizzato "Wit.ai", un servizio di "Natural Language Processing" gratuito e sviluppato da Facebook che, attraverso una semplice interfaccia grafica, permette di configurare il riconoscimento dei vari intent da parte dell'intelligenza artificiale. Erano stati anche valutati altri servizi tra i quali "Dialogflow" di Google, "Bot Framework" di Microsoft e "Watson" di IBM che, tuttavia, sono stati ritenuti troppo complessi per l'utilizzo nel progetto o non adeguati in quanto non accessibili gratuitamente.



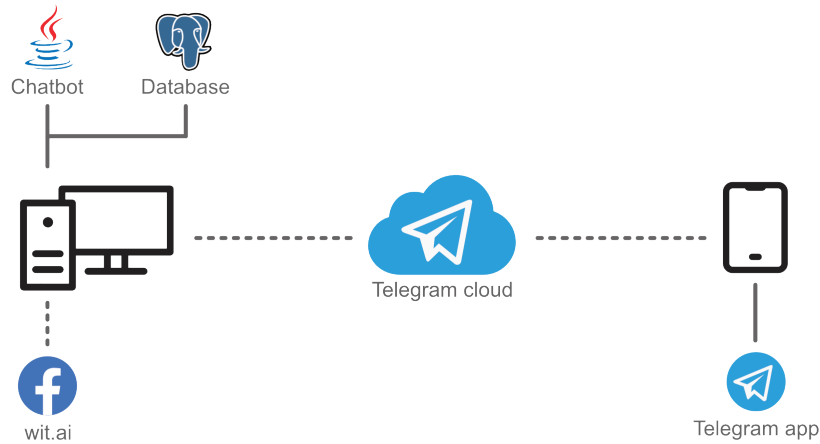


Figura 2.4: Funzionamento del sistema sviluppato durante il progetto AI4Hypertension

## 2.4 Sviluppo

L’implementazione del chatbot rispecchia in maniera accurata la struttura riportata precedentemente e si ha spesso una traduzione diretta dei componenti dell’architettura in classi Java nel codice ed è quindi per la maggior parte superfluo riportare dettagli implementativi specifici. Una delle cose degne di nota però è lo sviluppo di quello che nel diagramma viene identificato come BotMessage dove nel programma è stata creata una classe astratta “Abstract-BotMessage” che implementa elementi comuni a tutti i BotMessage e poi ne sono state create diverse sottoclassi ognuna relativa ad una tra le possibili risposte che il chatbot può fornire all’utente. L’interpretazione dei messaggi non avviene in locale ed è demandato ad un servizio esterno di Wit.ai al quale vengono effettuate richieste HTTPS che producono poi delle response il cui body contiene il risultato dell’interpretazione espresso in formato JSON. Per impostare i parametri necessari al funzionamento del chatbot e per semplificare l’utilizzo di una repository remota è stato messo a disposizione un file di configurazione esterno contenente informazioni quali i token dei servizi, i parametri e le credenziali di accesso al database. In questo modo, nel caso si voglia salvare il progetto in un servizio di hosting, è sufficiente escludere il file di configurazione dal caricamento per evitare di condividere dati sensibili o credenziali.



# Capitolo 3

## Hypertension Chatbot

Il chatbot sviluppato durante il progetto AI4Hypertension implementava con successo tutti gli obiettivi prefissati ed è alla base di tutti i miglioramenti e modifiche che sono state apportate in seguito sia per rispondere a nuove esigenze che per implementare ulteriori funzionalità che erano state tralasciate per mancanza di tempo. Il progetto è stato quindi preso come fondamento per le successive iterazioni e ampliato durante il periodo di tirocinio curricolare e di progetto per tesi. In questo capitolo verranno analizzate le nuove specifiche che sono state richieste e le soluzioni adottate e implementate durante il tempo a disposizione per rispondere alle nuove esigenze. Il codice sorgente del chatbot è disponibile al seguente indirizzo: [https://bitbucket.org/Riccardo\\_Omiccioli/hypertensionchatbot/src/master/](https://bitbucket.org/Riccardo_Omiccioli/hypertensionchatbot/src/master/)

### 3.1 Analisi dei requisiti

A seguito del confronto con il dott. Martino Pengo sono stati individuati i seguenti requisiti che avrebbero permesso di migliorare ed estendere il sistema già esistente:

- mandare in esecuzione in chatbot su macchine virtuali remote in modo da averlo sempre attivo in qualsiasi momento e non dover dipendere da un'esecuzione locale come invece avveniva precedentemente.
- migliorare il riconoscimento delle frasi e, soprattutto, dei valori inseriti dal paziente rendendo il sistema più robusto nei confronti delle possibili variazioni delle frasi che gli utenti potrebbero inserire.
- aggiungere nuove funzioni al chatbot pensate soprattutto per l'utilizzo da parte del personale medico per poter estrarre valori medi e misure in vari periodi temporali con una maggiore granularità ed essere in grado di

valutare i valori non solo nel periodo più recente ma anche in correlazione con dati misurati in un periodo di tempo più lungo.

- creare una dashboard per visualizzare le misurazioni ed alcune informazioni significative che possano permettere al medico di controllare con semplicità la condizione dei propri pazienti da remoto e in qualsiasi momento.

## 3.2 Progettazione e tecnologie scelte

A seguito della definizione di questi obiettivi sono state analizzate le possibili alternative a disposizione per l'implementazione e verranno ora spiegate le decisioni prese, sia a livello progettuale che di tecnologie utilizzate, motivandole in relazione ai requisiti che si vogliono soddisfare. Per quanto riguarda il primo punto, ovvero quello di poter avere il chatbot in esecuzione su una macchina remota, è stato subito chiaro che la scelta migliore sarebbe stata quella di utilizzare i servizi in cloud forniti da terzi. L'opzione di installare una macchina dedicata al chatbot era non solo troppo costosa ma anche meno affidabile rispetto alle garanzie che vengono fornite utilizzando alcuni dei servizi in cloud e quindi sono state inizialmente esplorate le alternative disponibili fornite dalle maggiori aziende del settore. Tra queste, le tre principali che sono state prese in considerazione, sono state Microsoft Azure, Google Cloud Services ed Amazon Web Services delle quali è stata poi scelta AWS sia per ragioni di costo che di semplicità e possibilità di configurazione dei loro servizi. La parte di riconoscimento delle frasi eseguita dall'intelligenza artificiale è sempre stata considerata come una delle parti critiche del sistema e questo perché, come già detto in precedenza, l'applicazione deve poter essere utilizzabile con semplicità da numerose categorie di utenti i quali spesso potrebbero non essere familiari con il mondo informatico. In particolare era stato notato, durante le prove eseguite sul chatbot sviluppato nel progetto AI4Hypertension, di come l'inserimento di nuove misure di pressione fosse particolarmente problematico quando si provavano ad usare formati diversi da quelli prestabiliti o di come non venisse effettuato alcun controllo nel caso i valori di pressione sistolica e diastolica venissero inseriti invertiti tra loro. Per quanto riguarda l'aggiunta di nuove funzioni al chatbot, alla fine del progetto AI4Hypertension era possibile:

- richiedere informazioni generali
- inserire una nuova misura
- richiedere la media giornaliera, settimanale e mensile

- visualizzare tutte le misure inserite nel sistema

A queste funzioni era necessario aggiungere la possibilità di:

- richiedere la media in un periodo di giorni a scelta
- richiedere tutte le misure inserite nell'ultimo giorno, settimana e mese
- richiedere tutte le misure inserite in un giorno a scelta
- visualizzare tutte le misure inserite in un periodo di giorni a scelta

Per aggiornare il chatbot con queste nuove funzionalità, tenendo conto della necessità di dover anche fornire al medico una dashboard per il controllo dei pazienti, si è reso necessario aggiornare il codice del chatbot e valutare come riorganizzare la struttura dei dati memorizzati nel database. Qui di seguito viene mostrato il diagramma dei casi d'uso delle nuove funzioni richieste che vanno aggiunte a quelle già presenti e delle quali si può osservare il diagramma in figura 2.1.

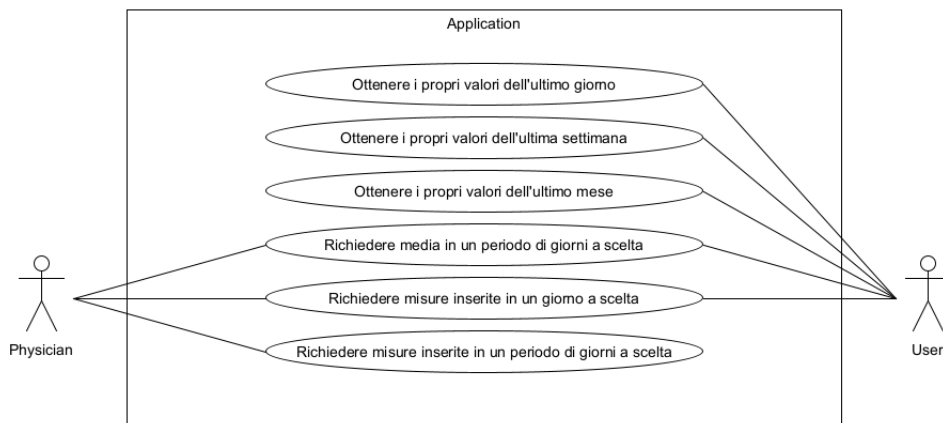


Figura 3.1: Diagramma dei casi d'uso delle funzioni aggiunte

Inizialmente il programma era stato pensato per ricevere come parametri dall'utente solamente i valori di pressione durante un nuovo inserimento, tuttavia, data la necessità di dover inserire anche, per esempio, date per specificare i giorni scelti è stata modificata la classe responsabile della modellazione di una interpretazione per accettare tipi di dati generali. Nella figura 3.2 è mostrato il diagramma delle classi del progetto iniziale e della versione modificata per adattarsi alle nuove specifiche. Si può notare nel diagramma di come i parametri di pressione sistolica, diastolica e frequenza cardiaca presenti nel componente "Interpretation" siano poi stati sostituiti da parametri generali.

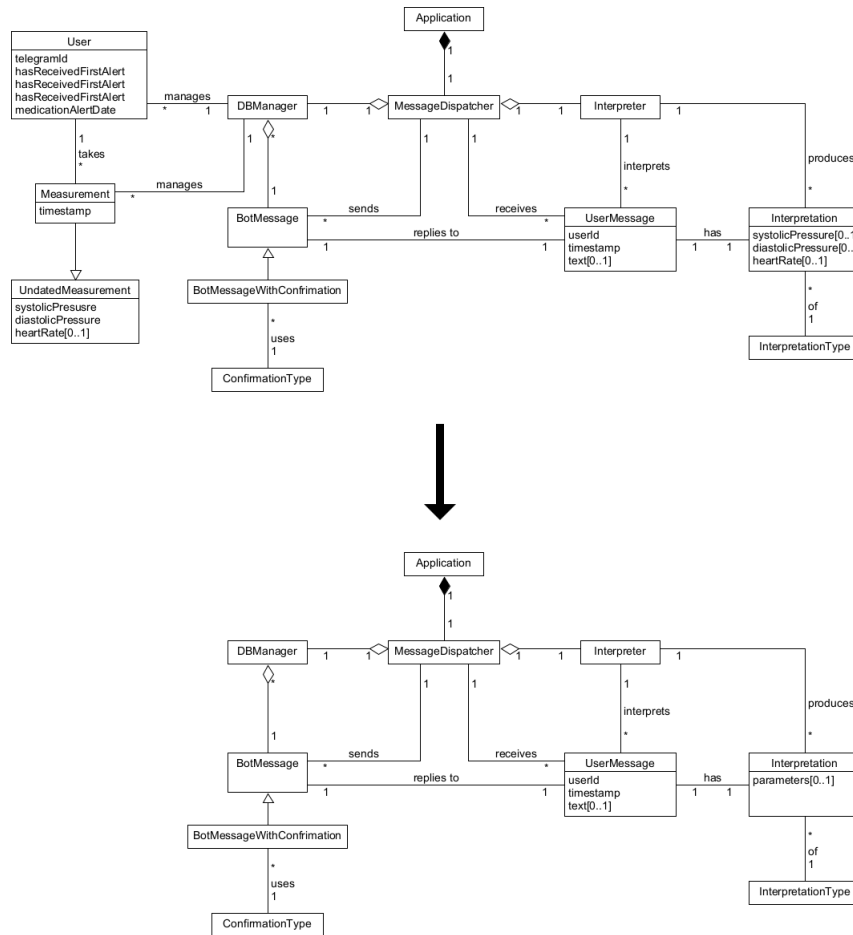


Figura 3.2: Modifiche effettuate alle classi del progetto

A seguito delle modifiche mancano anche i componenti “User”, “Measurement” e “UndatedMeasurement” che si sono resi non necessari e sono stati rimossi a seguito di alcune problematiche che verranno analizzati nel dettaglio nella parte di implementazione. Notevoli modifiche sono state apportate al database per adeguarlo ai nuovi requisiti cercando di prevedere anche possibili funzioni che potrebbero essere aggiunte in futuro. Nella prima versione il database memorizzava direttamente tutte le informazioni dell’utente nella tabella User dove comparivano infatti anche informazioni sulle notifiche date dal chatbot. Il modo in cui le misurazioni vengono memorizzate è rimasto quasi inalterato ma sono state rimosse da User tutte le informazioni non necessarie alla caratterizzazione di un utente spostandole in una tabella separata relativa alle notifiche. Sono state anche previste due tabelle, Admin e Doctor, con le quali è possibile gestire permessi forniti a medici e amministratori in previsione

della successiva aggiunta di un pannello di controllo per il personale medico. E' stata presa in considerazione anche la questione della privacy degli utenti visto che il sistema tratta quelli che vengono classificati come dati sensibili e quindi, è stato deciso di usare un semplice meccanismo per gestire il consenso al trattamento dei dati da parte degli utenti che si è tradotto, nel database, nell'aggiunta di una tabella dedicata a memorizzare i consensi forniti.

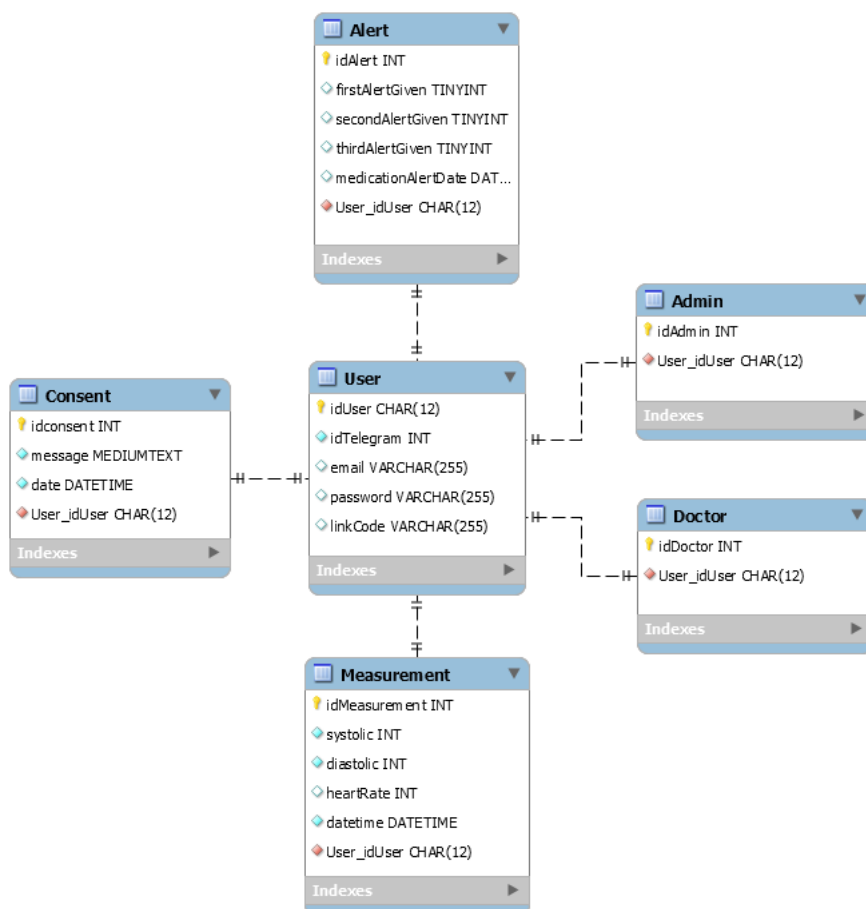


Figura 3.3: Modello del database modificato

Nel progetto AI4Hypertension gli utenti all'interno del database venivano riconosciuti attraverso il loro numero identificativo di Telegram ma, durante le modifiche effettuate, è stato scelto di identificare gli utenti del sistema utilizzando un codice separato, sia per anonimizzare gli utenti, dato che verranno poi identificati tramite una sequenza casuale non riconducibile ad alcuno dei loro dati, sia per rendere generale l'identificazione all'interno del database avendo in mente le aggiunte successive fatte al sistema dove possono essere usate altre

piattaforme diverse da Telegram. Come spiegato successivamente non dover più dipendere dall'identificativo di Telegram offre la possibilità di utilizzare il sistema senza la necessità di installare e creare un account Telegram se, per esempio, l'utente non si trova a proprio agio con la loro applicazione. Durante la progettazione è stato scelto di utilizzare un codice con una lunghezza di dodici caratteri in modo da avere molti più codici rispetto al possibile numero di utenti riducendo notevolmente la possibilità di collisione durante la creazione degli stessi migliorando le performance e garantendo allo stesso tempo una maggiore privacy agli utenti del sistema.

### 3.3 Implementazione

**Configurazione del chatbot in ambiente remoto** La configurazione su macchina virtuale remota del chatbot è stata effettuata durante il periodo di tirocinio curricolare durante il quale è stata dedicata una porzione di tempo allo studio del funzionamento generale del chatbot già sviluppato nel progetto AI4Hypertension in modo da comprendere quali componenti fossero utilizzati per poi essere configurati secondo le nuove esigenze. Come già detto, stati valutati diversi fornitori di servizi in cloud, ma tra loro è stato scelto di utilizzare i servizi offerti da AWS di Amazon considerando il costo adeguato per un possibile deployment, la flessibilità e scalabilità dei servizi offerti e l'importanza di AWS sia in generale nei cloud services che nel mondo IoT. In particolare è stato utilizzato il servizio di cloud computing EC2 e, una volta scelta la distribuzione Linux Ubuntu, è stata installata la JVM necessaria all'esecuzione del programma Java del chatbot e il servizio PostgreSQL per la gestione del database sulla macchina remota. Sono seguite ulteriori configurazioni nel programma del chatbot necessarie per utilizzare le API di Telegram e di Wit.ai i quali dettagli sono qui omessi.



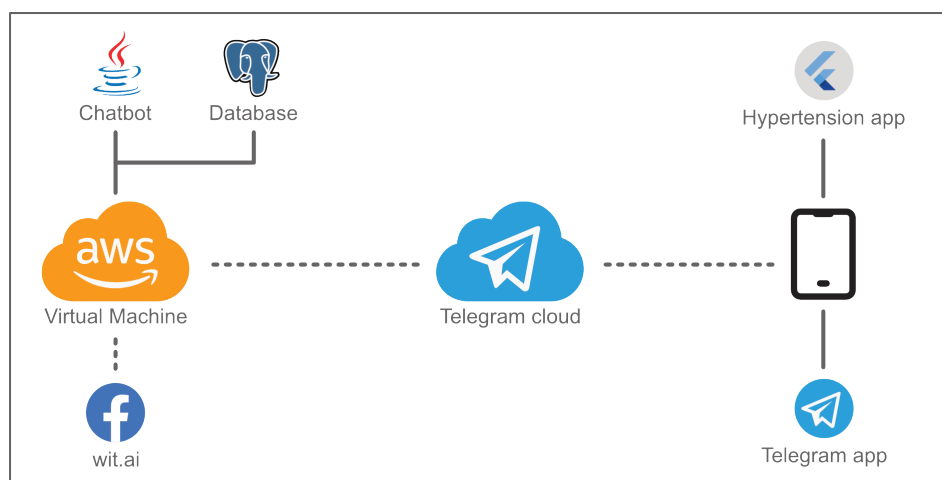


Figura 3.4: Funzionamento del sistema dopo la configurazione in ambiente remoto

**Miglioramento dell'interpretazione con IA** Nel progetto AI4Hypertension l'estazione dei valori di pressione sistolica, diastolica e di frequenza cardiaca veniva direttamente gestita dall'intelligenza artificiale attraverso quelle che vengono chiamate "entities" configurate per essere di tipo numerico e restituite direttamente al programma del chatbot insieme al tipo di intent interpretato. Questo ha causato numerosi problemi che risultavano nell'errata interpretazione dei dati soprattutto quando essi venivano scritti in un formato non accettato dalla parte di riconoscimento del linguaggio naturale fornita da wit.ai. Purtroppo la scelta dell'utilizzo di wit.ai si è rivelata, sotto questo punto di vista, sfavorevole dato che la sua semplicità nell'utilizzo attraverso un'interfaccia grafica minimale ha poi reso complesso personalizzare l'interpretazione corretta dei dati. Già le prime prove effettuate sul chatbot mettevano in evidenza di come, ad esempio, l'inserimento di una misura attraverso il messaggio "120-80" causasse problemi a causa della sbagliata interpretazione del tipo di entity che invece di essere di tipo numerico e associato ai ruoli di pressione sistolica e diastolica veniva considerato erroneamente come entity di tipo "date" o "phone-number". Non riuscendo a configurare il comportamento direttamente dall'interfaccia grafica è stato fatto un tentativo aggiungendo, tramite richieste dirette alle API, possibili inserimenti di valori di pressione con la corretta interpretazione in modo che l'intelligenza artificiale potesse usare tali esempi per migliorare l'affidabilità nel riconoscimento. Utilizzando richieste POST su protocollo HTTPS è possibile aggiungere quelle che vengono chiamate "utterances" ovvero frasi di esempio inserite dal programmatore che l'intelligenza artificiale utilizzerà per stimare in seguito possibili richieste simili a quelle già inserite. Il processo di apprendimento che, a partire da una serie di possibili

frasi, le associa ad uno dei possibili intent prende il nome di training ed è un'operazione di fondamentale importanza dato che, a seguito dell'inserimento di un numero adeguato di frasi, permette di migliorare gradualmente l'affidabilità nel riconoscere l'intent quando in futuro una frase simile verrà inserita dall'utente. Per l'inserimento e la memorizzazione viene usato il formato JSON e il listato 3.1 fornisce un esempio di come è formata una utterance. In particolare, l'utterance nel listato 3.1 è una di quelle che sono state utilizzate per cercare di migliorare il riconoscimento dei valori e nel campo "text" è presente il messaggio in linguaggio naturale che simula uno dei messaggi di inserimento che causava problemi. Nel campo "intent" viene segnato che si tratta appunto di una "pressure\_post" ovvero dell'inserimento di una nuova misura e il nome dell'intent è arbitrario e deciso dal programmatore, tuttavia, è necessario che sia il chatbot che l'intelligenza artificiale utilizzino lo stesso nome per riferirsi all'intent da interpretare. Come detto in precedenza i valori da estrarre vengono chiamati entity e nell'esempio ce ne sono appunto due di tipo numerico con ruolo rispettivamente di pressione sistolica e diastolica.

```
[{
  "text": "120-80",
  "entities": [
    {
      "entity": "wit$number:systolic_pressure",
      "start": 0,
      "end": 3,
      "body": "120",
      "entities": []
    },
    {
      "entity": "wit$number:diastolic_pressure",
      "start": 4,
      "end": 6,
      "body": "80",
      "entities": []
    }
  ],
  "traits": [],
  "intent": "pressure_post"
}]
```

Listato 3.1: Esempio di utterance in JSON

Attraverso un semplice programma scritto in linguaggio Python sono state aggiunte un numero considerevole di utterances, generando in modo casuale i

valori e utilizzando numerosi possibili formati nella scrittura dei dati numerici, ma, anche dopo questa procedura per migliorare il riconoscimento, i risultati non sono stati del tutto soddisfacenti. Ulteriori tentativi sono stati fatti utilizzando altre intelligenze per il riconoscimento testuale come, ad esempio, Lex di Amazon che però soffriva anch'essa degli stessi problemi dato che i valori venivano riconosciuti solamente quando erano inseriti con certi formati. Considerando che i dati da interpretare sono pochi e che utilizzare un sistema di intelligenza artificiale più sofisticato avrebbe richiesto una mole di lavoro notevole è stato scelto di non utilizzare più l'intelligenza artificiale per l'estrazione dei dati dal testo del messaggio e di demandare tale compito a codice specializzato all'interno del chatbot.

**Ridefinizione struttura chatbot e aggiunta nuove funzionalità** Rispetto alla versione iniziale del programma sono poi state richieste ulteriori funzioni soprattutto per quanto riguarda la possibilità di estrarre le medie e i valori in periodi di tempo arbitrari per aumentare il grado di libertà nella visualizzazione dei dati. Dato che ora non venivano solamente passati come input i parametri di pressione sistolica, diastolica e frequenza cardiaca presenti nel componente "Interpretation" essi sono poi stati sostituiti da parametri generali che, nel codice, sono di tipo `HashMap<String, Optional<?>>`. Utilizzando una `HashMap` è possibile adattare il codice, con poche semplici modifiche, a ricevere una quantità variabile di dati a seconda delle necessità delle varie interpretazioni e con le wildcards di Java il tipo di queste variabili può essere scelto per rappresentare, ad esempio, valori numerici per gli inserimenti o date per le richieste. Come già accennato mancano anche le classi "User", "Measurement" e "UndatedMeasurement" presenti, nel codice del programma, attraverso classi auto-generate necessarie al funzionamento di Jooq. Jooq fornisce un modo semplice di interagire con il database mappando la struttura del database con classi utilizzabili in Java rendendo le query e la manipolazione dei dati molto semplice e pulita. Tuttavia, durante la fase di test interno, ci si è accorti di come Jooq non fornisse consistenza nei dati inseriti nel database causando alcuni malfunzionamenti del sistema come la mancanza di dati inseriti nel database e il conseguente errato funzionamento di alcune richieste fatte al chatbot che necessitavano della lettura dei dati inseriti in precedenza. Le classi autogenerate sono quindi state completamente rimosse e al posto di Jooq sono state utilizzate direttamente le funzioni SQL messe a disposizione da Java attraverso JDBC che hanno permesso di risolvere i problemi riscontrati a scapito di una parziale complicazione del codice relativo all'interazione con il database. Per quanto riguarda le modifiche alla struttura del database esse sono state apportate anche alle richieste SQL effettuate nel chatbot che comunque, come appena detto, andavano comunque riscritte per risolvere gli

errori che si verificavano. Per la gestione della privacy degli utenti è stato implementato un semplice meccanismo di controllo che, quando il chatbot viene avviato per la prima volta, ignora ogni richiesta fatta dall'utente fino a quando non viene fornito il consenso e che, anche in seguito, quando vengono effettuate operazioni sui dati, controlla che l'utente abbia prima fornito il consenso. Dato che con un chatbot è difficile dimostrare che l'utente ha accettato le condizioni per dare il proprio consenso è necessario scrivere un messaggio esplicito al bot che poi salverà nel database insieme all'identificativo dell'utente e la data e ora dell'inserimento. Come nuovo codice identificativo degli utenti è stato scelto di usare un codice univoco di 12 caratteri dove possono comparire i caratteri "0123456789ABCDEFGHIJKLMNPQRSTUVWXYZ" e questo perché, alcune lettere come, ad esempio, o, 0, i, I, l non sono state inserite tra i possibili caratteri ammissibili in quanto possono essere difficilmente riconoscibili tra loro o con altri caratteri e al loro posto, riprendendo l'esempio appena fatto, sono stati scelti i numeri 0 e 1. Omettere alcuni dei caratteri fraintendibili dalla lista dei simboli accettati ha reso possibile implementare un sistema di controllo che, nel caso un utente inserisca proprio uno di tali caratteri perché magari interpreta male il codice inserito, sostituisca nel codice i caratteri non ammessi con quelli corretti rendendo comunque valido l'inserimento dell'utente. Spesso, quando viene visualizzato un codice identificativo nel chatbot o nelle altre piattaforme i dodici caratteri vengono separati a gruppi di tre per rendere più semplice la comunicazione nel caso questa debba avvenire verbalmente anche se, a livello effettivo, i codici vengono memorizzati senza alcun separatore.

**Creazione di una dashboard** Per valutare l'utilizzo di interfacce grafiche personalizzate allo scopo di rendere più semplice l'utilizzo del bot e per formare una base sul quale aggiungere gli strumenti per il personale medico è stato sviluppato un prototipo di applicazione che potesse funzionare in parallelo con il client di Telegram. In particolare l'applicazione utilizza, tramite librerie di interfacciamento, lo stesso cloud di Telegram usato nell'applicazione ufficiale, rendendo così i due sistemi sincronizzati in automatico tra loro. Anche se poi non è stata effettivamente sviluppata, un'altra ragione che ha portato alla creazione di un prototipo dell'applicazione è stata quella di poter sfruttare il modulo Bluetooth per la comunicazione con dispositivi di misura smart, in modo da rendere la procedura di inserimento dei dati ancora più semplice cosa che invece, con il client di Telegram, non sarebbe stato possibile fare. Per lo sviluppo è stato utilizzato il framework cross-platform Flutter e il linguaggio di programmazione Dart in modo da non dover programmare e mantenere due versioni dell'applicazione separate per Android e iOS. L'interfaccia si ispirava principalmente a quella delle principali applicazioni messaggistiche e permetteva di visualizzare i messaggi mandati al chatbot in precedenza e di scriverne

nuovi in linguaggio naturale così come ci si aspetterebbe da un'applicazione di questa categoria, tuttavia, come esempio, è stato anche implementato un menu dedicato all'inserimento di nuovi valori di pressione. Una volta scaricata l'applicazione è necessario fornire il proprio numero di telefono e inserire la one time password inviata tramite SMS al numero inserito in modo pressoché identico a quanto accade durante il primo avvio del client di Telegram. Di seguito sono mostrate alcune schermate dell'applicazione.

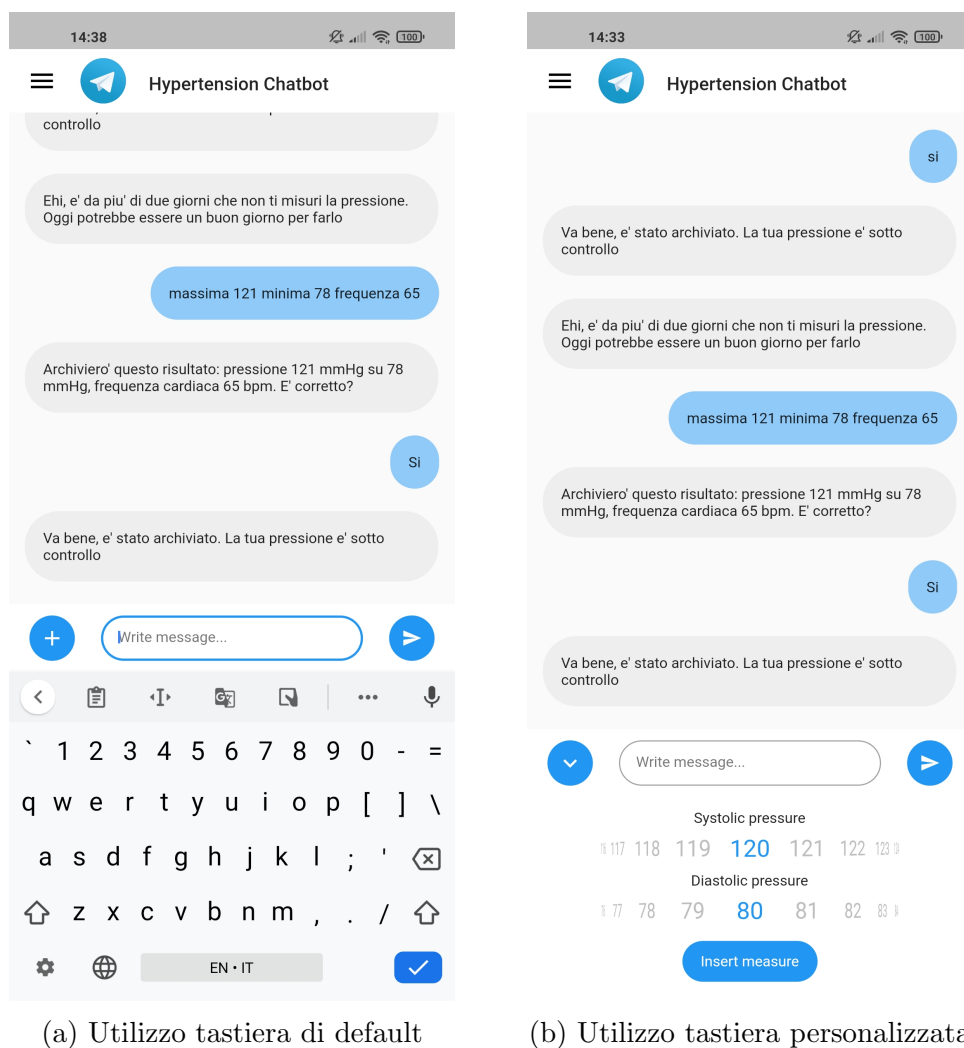


Figura 3.5: Immagini dell'interfaccia grafica di Hypertension App

Anche se poi è stato deciso che avere un'app dedicata fosse meno pratico, il fatto di aver dimostrato che, grazie al cloud di Telegram, è possibile interagire in modo semplice con il sistema e sviluppare un'applicazione alternativa è stato sicuramente interessante. Riguardo agli strumenti pensati per essere

utilizzati dal medico si è deciso di non implementarli in una applicazione come questa ma di renderli disponibili attraverso una piattaforma web. I dettagli della piattaforma web verranno analizzati nel capitolo successivo dato che la realizzazione ha richiesto un periodo di tempo notevole e la descrizione di esso sarebbe troppo ristretta se limitata a questo paragrafo.

# Capitolo 4

## Hypertension Website

A seguito dell'incontro con il dott. Martino Pengo è emersa come una delle priorità maggiori quella di incrementare gli strumenti a disposizione del personale medico rispetto al progetto AI4Hypertension iniziale. La parte di miglioramento effettuata sul chatbot descritta in precedenza ha permesso di estrarre i valori e le medie su di essi con maggiore flessibilità ma non era comunque uno strumento sufficiente da solo dato che comunque il medico avrebbe dovuto controllare i dati del paziente attraverso il dispositivo dello stesso o facendosi comunicare attraverso messaggi o immagini. La loro necessità era quella di avere una dashboard personalizzata per le loro esigenze che fornisse le funzionalità di cui avevano bisogno e, anche se era stata sviluppata Hypertension App che poteva essere eventualmente espansa a questo scopo, è stato deciso che un portale web sarebbe stata l'alternativa più pratica per l'utilizzo. Il codice sorgente della dashboard web è disponibile al seguente indirizzo: [https://bitbucket.org/Riccardo\\_Omiccioli/hypertensionwebsite/src/master/](https://bitbucket.org/Riccardo_Omiccioli/hypertensionwebsite/src/master/)

### 4.1 Requisiti e tecnologie scelte

Durante il confronto è emerso che il sito avrebbe dovuto mostrare per ogni paziente i suoi valori, alcuni indicatori statistici per visualizzare al volo gli andamenti, un grafico delle misurazioni e la possibilità di estrarre tutti i dati del paziente sotto forma di file CSV. Oltre a questi requisiti incentrati più sull'aspetto medico il sito avrebbe dovuto implementare un sistema di login, di gestione del proprio account e di gestione degli utenti per l'amministratore. Grazie alla nuova organizzazione dei dati nel chatbot, creata appunto considerando anche la possibilità di implementare una dashboard, si è pensato di rendere possibile l'uso del sistema, attraverso il sito, indipendente dalla piattaforma Telegram. Non solo un medico può usare il portale web senza dover inserire i propri dati personali telefonici o di Telegram ma anche il paziente

può inserire misure e visualizzare i propri dati tramite il sito se, per esempio, non ha intenzione di installare o utilizzare Telegram. Ovviamente utilizzando il sito si rinuncia a tutta la parte di interazione con il chatbot ma si è comunque cercato di rendere l'uso il più semplice possibile attraverso una interfaccia grafica semplice e ispirata, come “look and feel”, ad applicazioni mobile di uso comune. Dato che per utilizzare il sito è necessario creare un account tramite email e password si è voluta comunque dare la possibilità di unire l'account del sito a quello su Telegram in modo da poter utilizzare il sistema, per inserire e visualizzare le proprie misure, in maniera trasparente tra le due piattaforme. Dati i requisiti per questa dashboard è stato deciso che sarebbe stato sufficiente realizzare una pagina web utilizzando tecnologie già conosciute come HTML, CSS, Javascript e PHP ma, in futuro, si potrebbe pensare di trasformare la dashboard in una webapp che possa fornire anche una chat di interazione con il chatbot in maniera simile a quanto avviene su Telegram. In particolare HTML e CSS sono stati utilizzati, ovviamente, per strutturare il sito e per controllare il suo aspetto grafico, Javascript per l'interazione di alcuni elementi del sito e PHP per realizzare la parte di interazione con il database e le API utilizzate da Javascript per effettuare richieste AJAX.

## 4.2 Sviluppo e dettagli implementativi

Sulla stessa macchina virtuale fornita da AWS è stato installato e configurato il web server NGINX per poter ospitare il sito e tramite un registratore di domain names è stato associato il nome “hypertension.website” all'indirizzo IP pubblico della macchina virtuale. Tramite questa configurazione è quindi possibile utilizzare la stessa macchina virtuale sia per ospitare il sito che il chatbot riducendo in futuro una possibile fonte di costi dato che per far funzionare il sistema sarà sufficiente disporre di un dominio e di una sola macchina virtuale adeguata. Durante la fase di test anche la macchina virtuale gratuita fornita da AWS, che dispone di prestazioni ridotte, è stata in grado di far funzionare il sistema nella sua completezza e con tempi di esecuzione soddisfacenti. Il sito, per favorire il riuso di codice comune tra le pagine, utilizza un template di base che contiene la barra superiore (header) con le sue funzionalità e la barra inferiore (footer) mentre, nella parte centrale (body), viene caricato il contenuto che varierà da pagina a pagina. Viene utilizzato un foglio di stile generale per definire le regole comuni che definiscono l'aspetto del sito e, tramite fogli specifici per le singole pagine, vengono invece personalizzati alcuni elementi che dipendono dalla specifica pagina caricata. Il sito è stato creato utilizzando un approccio “mobile first” ovvero sviluppando prima una versione adatta alla visualizzazione per smartphone e poi estendendone una versione adatta alla



visualizzazione su desktop. Utilizzare l’approccio mobile first è una pratica consigliata per generare siti con un layout migliore essendo la prima versione sviluppata per uno schermo ridotto e che pone maggiori problematiche nel renderne la navigazione agevole e solo successivamente il sito viene adattato per la visualizzazione desktop, dove la maggiore grandezza dello schermo pone meno vincoli alle scelte di design del sito. Per presentare all’utente la versione migliore del sito rispetto al dispositivo che sta utilizzando sono state usate delle “media queries”, ovvero delle dichiarazioni CSS che cambiano le regole grafiche del sito a seconda di certi parametri che, nel caso di questo sito, fanno riferimento alla larghezza dello schermo del dispositivo dove sotto una soglia di 1200px di larghezza viene utilizzata la versione mobile e sopra la versione desktop. Nella seguente figura è possibile osservare una delle pagine del sito sia nella versione mobile che nella versione desktop.

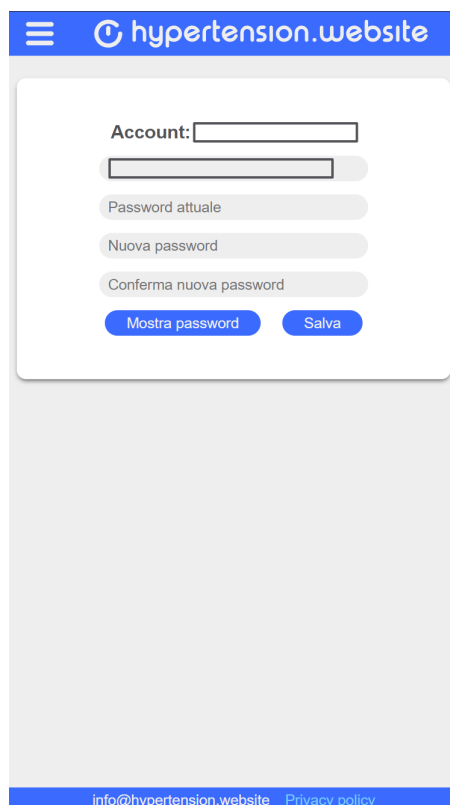
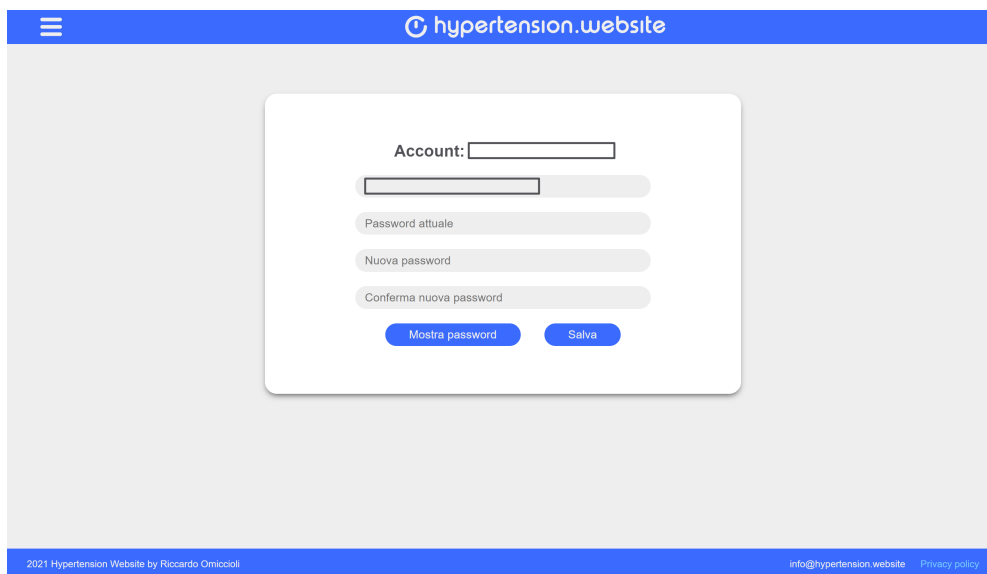


Figura 4.1: Visualizzazione pagina modifica informazioni account versione mobile



The image shows a desktop view of a web application interface for account management. At the top, there is a blue header with a hamburger menu icon on the left and the text "hypertension.website" on the right. The main content area is a light gray background with a white rounded rectangle in the center. Inside this rectangle, the word "Account:" is followed by a text input field. Below it are four password input fields: "Password attuale", "Nuova password", and "Conferma nuova password". At the bottom of the form are two blue buttons: "Mostra password" and "Salva". At the very bottom of the page, there is a blue footer with the text "2021 Hypertension Website by Riccardo Omiccioli" on the left, and "info@hypertension.website" and "Privacy policy" on the right.

Figura 4.2: Visualizzazione pagina modifica informazioni account versione desktop

Anche se non richiesto dalle specifiche è stata utilizzata una media query per rilevare la visualizzazione come anteprima di stampa e, per la pagina di visualizzazione dei dati di pressione, sono state scritte delle regole specifiche per rendere ottimizzata la visualizzazione dei dati nel caso si voglia stamparli in formato cartaceo o salvarli come PDF. Nell'immagine sottostante è rappresentata la pagina di visualizzazione dei valori di pressione così come verrebbe visualizzata durante la stampa.

Media utente:

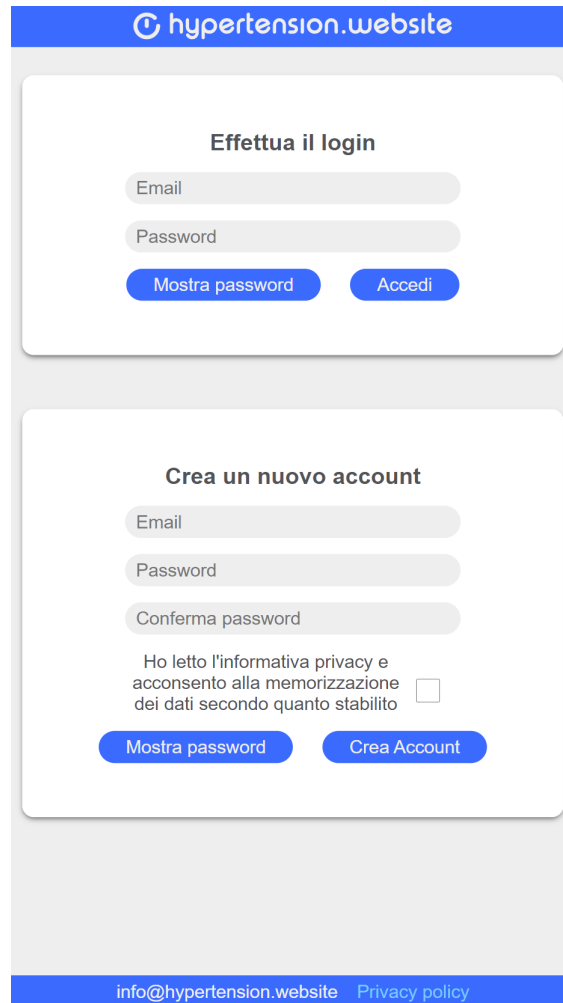
Media ultima settimana	Media ultimo mese	Media ultimi 3 mesi	Media ultimi 6 mesi
138/88 (3)	121/80 (40)	121/80 (40)	121/80 (40)

Dati utente:

Data (GMT)	Sistolica	Diastolica	Frequenza
2021-11-05 21:37:51	145	93	
2021-11-05 21:35:27	150	95	
2021-11-05 21:17:01	120	75	56
2021-11-01 13:06:30	115	75	67
2021-11-01 00:35:04	120	80	
2021-10-28 11:29:57	124	84	
2021-10-28 09:52:58	120	76	67
2021-10-28 09:38:28	130	83	
2021-10-28 06:55:07	125	83	
2021-10-25 17:11:54	120	80	
2021-10-25 16:05:35	125	79	66
2021-10-25 14:28:15	132	89	76
2021-10-25 14:27:59	130	87	
2021-10-25 14:27:51	125	85	
2021-10-25 13:41:43	125	80	
2021-10-25 13:29:52	115	70	
2021-10-25 10:12:26	120	80	
2021-10-25 10:02:55	119	75	
2021-10-24 16:23:08	120	80	67
2021-10-24 14:49:44	120	80	60
2021-10-24 13:29:44	124	87	
2021-10-24 13:29:31	113	76	
2021-10-24 13:29:21	123	82	
2021-10-24 13:29:05	120	80	76
2021-10-23 17:03:26	100	65	
2021-10-23 16:58:05	130	80	
2021-10-23 14:06:44	110	65	
2021-10-23 14:05:52	135	93	
2021-10-23 14:05:24	130	90	
2021-10-22 14:07:54	109	75	64
2021-10-22 14:05:13	110	76	65
2021-10-22 11:59:12	120	80	
2021-10-21 19:15:50	110	70	
2021-10-21 19:15:43	120	80	
2021-10-21 19:15:37	120	80	
2021-10-13 18:23:29	116	76	
2021-10-13 11:21:02	120	80	
2021-10-13 10:17:09	110	70	
2021-10-13 08:18:03	110	70	
2021-10-13 08:15:09	120	80	

Figura 4.3: Visualizzazione dell'anteprima di stampa della pagina di visualizzazione delle misure

Come già detto in precedenza per poter utilizzare il sito è necessario registrarsi su di esso inserendo email e password ed è possibile farlo attraverso la pagina mostrata in figura 4.4 che, oltre al form per completare la registrazione, presenta anche gli elementi necessari per effettuare il login. Tale pagina, viene visualizzata in automatico ogni volta che si prova ad accedere al sito senza aver effettuato il login nella sessione corrente impedendo la visualizzazione di altre pagine che non avrebbe senso vedere se non quando si è effettuato l'accesso al proprio account.



The image shows a mobile application interface for 'hypertension.website'. It features two main sections: a login form and a registration form. The login form, titled 'Effettua il login', includes input fields for 'Email' and 'Password', a 'Mostra password' button, and an 'Accedi' button. The registration form, titled 'Crea un nuovo account', includes input fields for 'Email', 'Password', and 'Conferma password', a 'Mostra password' button, and a 'Crea Account' button. Below the registration form, there is a checkbox for 'Ho letto l'informativa privacy e acconsento alla memorizzazione dei dati secondo quanto stabilito'. The footer contains the email 'info@hypertension.website' and a link to 'Privacy policy'.

Figura 4.4: Pagina login e registrazione versione mobile

Dato che dal sito è comunque possibile inserire nuove misure di pressione si è reso necessario, almeno in maniera semplice, gestire i consensi degli utenti che vengono forniti cliccando sulla casella obbligatoria durante la fase di registrazione. Per essere sicuri che l'utente abbia fornito il proprio consenso vengono effettuati controlli non solo lato Javascript andando a vedere se la casella sia spuntata o meno ma anche a livello di API controllando che sia presente il parametro relativo al consenso e inserendo in automatico, nel database, un nuovo record nella tabella dei consensi contenente il codice identificativo dell'utente, la data-ora e una frase preimpostata uguale a quella che viene visualizzata a lato della casella. In modo analogo a quanto avviene quando si utilizza il sistema tramite Telegram viene associato ad ogni utente registrato un codice univoco di dodici cifre che gode delle stesse proprietà discusse in

precedenza correggendo alcuni errori di digitazione o interpretazione dei caratteri che l'utente potrebbe commettere. Inoltre, per migliorare la sicurezza degli utenti, tutte le password degli account relativi al sito sono memorizzate a seguito di funzioni di hashing e salatura in modo che nessuno, neanche chi ha accesso in lettura al database, possa leggere o ricavare la password degli utenti. Una volta registrati o effettuato il login si viene rediretti alla pagina principale del sito ovvero quella per visualizzare le proprie misure ed aggiungerne di nuove del quale ne è presentato un esempio nell'immagine 4.5. Nella figura 4.6 è invece mostrato il form per l'inserimento di nuove misurazioni di pressione per il proprio account accessibile tramite il pulsante aggiungi che si trova nella parte destra della barra superiore. Il primo elemento di tale pagina è il grafico che riporta i valori di pressione sistolica e diastolica dell'utente insieme ai corrispondenti valori limite, esso, è stato realizzato utilizzando la libreria open source chart.js e riporta le date degli inserimenti nell'asse delle ascisse e i relativi valori di pressione nell'asse delle ordinate (espressi in mm-Hg). Di tutte le misure inserite solamente un numero ridotto di esse viene graficato contemporaneamente sia per ridurre il costo computazionale sia, soprattutto, per rendere più leggibile il grafico e, tramite i pulsanti presenti sotto il grafico, è possibile scorrere il grafico per visualizzare le misure effettuate in altre date. Subito sotto è presente un riepilogo delle pressioni medie nell'ultima settimana, mese, 3 mesi e 6 mesi e, per ogni media, è anche indicato tra parentesi il numero di misure che contribuisce al calcolo di tale media. Tale tabella è stata sviluppata su suggerimento del dott. Martino Pengo in modo da permettere al medico, durante il controllo dei valori dei propri pazienti, di vedere rapidamente le medie in periodi diversi e di considerare in relazione al numero di misure che contribuiscono al calcolo quanto possa essere variabile la media ottenuta. Successivamente viene mostrata una tabella riepilogativa delle misure inserite dall'utente riportando data, pressione sistolica, pressione diastolica ed opzionalmente la frequenza inserita in modo simile a quanto viene visualizzato tramite il chatbot con il comando per visualizzare tutte le misure dell'utente. Come ultimo elemento è presente un pulsante per effettuare il download delle misure in formato CSV il quale, attraverso una semplice funzione scritta in Javascript, avvia in automatico il download di un file con nome corrispondente al codice utente e come contenuto i valori di data-ora, pressione sistolica, pressione diastolica, frequenza cardiaca, media settimanale sistolica, media settimanale diastolica, deviazione standard sistolica, deviazione standard diastolica. I valori di media settimanale riportati sono calcolati considerando ogni settimana come i giorni che vanno tra lunedì a domenica piuttosto che effettuare le medie a partire dal giorno che si effettua il calcolo e andando indietro di sette giorni per volta.



Figura 4.5: Pagina di visualizzazione delle misure (versione desktop)



Figura 4.6: Form per l'inserimento di nuove misure (versione desktop)

Oltre a riportare le misurazioni dell'utente che ha effettuato il login questa è la pagina che viene anche usata dal personale medico per visualizzare i valori dei propri pazienti e, infatti, se l'utente è registrato nel sistema come medico esso potrà utilizzare la barra di ricerca presente in cima alla pagina, così come si può notare dall'immagine 4.5, mentre se l'utente che visualizza la pagina non possiede i permessi necessari la barra di ricerca non verrà caricata e visualizzata. Per visualizzare i dati di un paziente il medico dovrà inserire il codice utente del paziente ed effettuare la ricerca che, in caso di successo, caricherà la stessa pagina ma relativa questa volta all'utente cercato. Attraverso il menu accessibile con l'icona a sinistra della barra superiore il medico ha anche accesso ad una pagina contenente la lista degli utenti del sistema nella quale, come mostrato in figura 4.7, insieme ad una ulteriore barra di ricerca degli utenti è presente una tabella degli utenti che ne riassume i codici identificativi e le piattaforme utilizzate da esso. Ovviamente, nell'immagine inserita, i codici degli utenti sono stati nascosti.

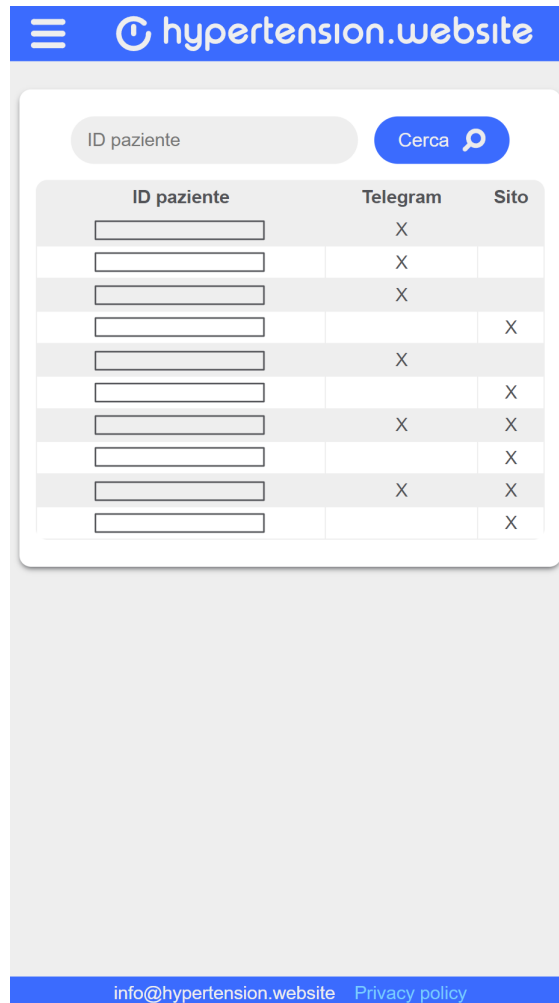


Figura 4.7: Pagina di visualizzazione della lista utenti versione mobile

Cliccando una delle righe della tabella verrà caricata in automatico la pagina di visualizzazione dei dati corrispondente all'utente selezionato. E' possibile notare nella tabella degli utenti di come alcuni di loro utilizzino esclusivamente il chatbot Telegram oppure il sito mentre alcuni utilizzano entrambe le piattaforme e questo perché, come accennato in precedenza, è stata implementata la possibilità di associare l'account creato per il sito a quello utilizzato con il chatbot. La procedura per unire i due account inizia chiedendo al chatbot di generare un nuovo codice per il sito il quale risponderà assegnando all'utente un nuovo codice di 18 cifre generato casualmente molto simile a quello utilizzato per gli utenti e che gode delle stesse proprietà già discusse in precedenza per i codici utilizzati. Dopo aver effettuato il login, andando nella propria pagina di impostazioni account è possibile inserire nell'apposita casella il codice



fornito dal chatbot e, dopo aver inserito la password corrente, è possibile confermare le nuove impostazioni. Appena salvate le impostazioni verrà effettuato il logout in automatico e sarà necessario effettuare nuovamente l'accesso utilizzando sempre la stessa email e password che veniva utilizzata prima, inoltre, a seguito della prima volta e in caso di successo della procedura, la casella per inserire il codice non verrà più mostrata. Durante l'unione degli account a tutte le misure inserite tramite il sito viene cambiato l'utente con quello relativo a Telegram e sempre in tale account vengono aggiunte le informazioni per effettuare il login tramite il sito. La scelta di portare tutte le informazioni dall'account del sito a quello di Telegram è stata solamente presa in quanto in questo modo vengono modificate meno tabelle nel database e nulla vieta di poter programmare la procedura nel modo opposto. Inoltre, in futuro, si potrebbe pensare di prevedere l'inserimento dei valori tramite file CSV in modo che l'utente possa fare il backup dei propri dati per, ad esempio, trasferirli in un nuovo account. Sempre attraverso la pagina delle impostazioni account è possibile anche modificare la propria email e password inserendo i nuovi dati, la vecchia password e cliccando sul pulsante per salvare le impostazioni. Per interagire con il database sono state create due classi PHP, DatabaseReader e DatabaseWriter delle quali si può osservare il diagramma in figura 4.8

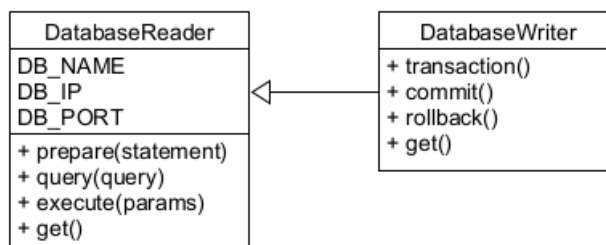


Figura 4.8: Diagramma delle classi DatabaseReader e DatabaseWriter

DatabaseWriter estende la classe DatabaseReader ed entrambe possiedono e definiscono il metodo get il quale istanzia la classe dopo aver recuperato dalla classe Credentials lo username e password rispettivamente dell'utente con permessi di lettura e di scrittura, migliorando in questo modo la separazione dei dati sensibili delle credenziali dal codice delle classi del database in modo da ridurre il rischio di fornire tali dati a terzi. DatabaseReader inizializza nel proprio costruttore la connessione con il database Postgres e fornisce inoltre i metodi prepare, query ed execute che a loro volta utilizzano i metodi adeguati di Postgres per effettuare tali operazioni. DatabaseWriter invece, oltre ai metodi ereditati da DatabaseReader, fornisce in modo simile le funzioni di transaction, commit e rollback utilizzate durante inserimenti o modifiche al

database. Esse sono molto importanti in quanto, dopo aver iniziato una transaction, nel caso una query sul database produca un errore nessuna operazione viene salvata e solo in caso di successo di tutte le query le modifiche vengono effettivamente confermate, tramite commit, evitando in caso di operazioni multiple il successo di solo una porzione di esse causando inconsistenza nei dati del database. Le due classi, User e Measurement, sono state create per rappresentare come classi PHP le informazioni presenti nelle relative tabelle del database ed infatti mettono a disposizione gli stessi campi che vengono utilizzati per la memorizzazione dei dati. Oltre a diversi metodi getter e setter per interagire sulle variabili delle classi esse presentano i metodi utilizzati per eseguire le query che riguardano appunto le funzionalità eseguite sugli utenti e sulle misurazioni. Utilizzando DatabaseReader e DatabaseWriter appena descritti nei metodi forniti da User e Measurement e creando query apposite per le funzioni necessarie al sito è possibile aggiungere nuovi dati o ottenere informazioni su quelli già presenti nel database. Le richieste fatte al server eseguite in varie parti del sito utilizzano AJAX (Asynchronous JavaScript and XML) per eseguire operazioni tramite le API PHP che il server mette a disposizione. Uno degli esempi più significativi è riportato nel listato 4.1 nel quale vengono richiesti al server i dati delle misurazioni da inserire poi nel grafico gestito da Chart.js.

```
let params = (new URL(document.location)).searchParams;
let idUser = params.get("idUser");
var oReq = new XMLHttpRequest();
var sendParams = new URLSearchParams();
if(idUser !== null) {
    sendParams.append('idUser', idUser);
}
oReq.open("GET", "/api/pressureGet.php?" + sendParams.toString());
oReq.send();
oReq.onreadystatechange = function() {
    if(this.readyState != 4) {
        return;
    }
    if (this.status == 200) {
        measurements = JSON.parse(this.response);
        if(measurements) {
            measurements.forEach(data => {
                var date = new Date(data["dateTime"]);
                labels.push(date.getDate() + "/" + (date.getMonth() +
                    1));
                systolic.push(data["systolic"]);
                diastolic.push(data["diastolic"]);
            });
        }
    }
}
```

```
        });  
        dataChart.update();  
    }  
}  
}
```

Listato 4.1: Richiesta AJAX per prelevare tutte le misurazioni di un utente e inserirle nel grafico

Come si può notare dal codice nel caso un medico stia cercando di visualizzare i dati di un paziente il codice utente del paziente viene prelevato dai parametri presenti nell'URL e aggiunto tra quelli da mandare nella richiesta che, in questo caso specifico, è di tipo GET. Come si può intuire la richiesta viene soddisfatta in modo asincrono e, quando lo stato della richiesta cambia, in caso di successo (segnato con stato uguale a 200) tutte le misure ottenute vengono aggiunte alle variabili del grafico labels, systolic e diastolic per poi essere aggiornato. Nel listato 4.2 è invece possibile osservare il codice della API relativa sempre all'operazione di estrazione delle misure di un utente. Innanzitutto viene effettuato un controllo per verificare l'esistenza di un utente nella sessione corrente e se esso sia presente tra quelli del database, in caso contrario la richiesta termina con codice 400. Nel caso la richiesta abbia come parametro l'idUser di un paziente e l'utente corrente sia un dottore allora le misurazioni restituite saranno quelle relative al paziente mentre, se non sono presenti parametri o se non si hanno i permessi di medico necessari, verranno restituite quelle dell'utente corrente. In caso di successo viene restituito il codice 200, come visto nella parte JavaScript, e la risposta contenente tutti i dati mentre in caso di errore nel prelevamento dei dati o di eccezione generale vengono restituiti rispettivamente i codici 401 e 500 per segnalare la condizione di errore.

```
<?php  
require_once(dirname($_SERVER['DOCUMENT_ROOT']) .  
    "/private/path.php");  
use Database\User as User;  
use Database\Measurement as Measurement;  
  
if (!isset($_SESSION)) {  
    session_start();  
}  
if (!isset($_SESSION["user"]) || User::get($_SESSION["user"]) ===  
    null) {  
    http_response_code(400);  
    exit();  
}
```

```
}  
  
$result = null;  
try {  
    if(isset($_GET["idUser"]) &&  
        User::get($_SESSION["user"]->getIdDoctor() !== 0) {  
        // If requesting data of idUser and current user is a doctor  
        $result = Measurement::getAll($_GET["idUser"]);  
    } else {  
        // Getting data of current user  
        $result = Measurement::getAll($_SESSION["user"]);  
    }  
} catch (Exception $e) {  
    http_response_code(500);  
    exit();  
}  
  
if ($result !== null) {  
    http_response_code(200);  
    $response = json_encode($result);  
    echo $response;  
    exit();  
} else {  
    http_response_code(401);  
    exit();  
}  
http_response_code(500);  
die();  
?>
```

Listato 4.2: Codice della API per ottenere le misurazioni di un utente

# Capitolo 5

## Discussioni finali sul sistema

Verranno fatte ora delle considerazioni finali sullo stato del sistema in relazione a quanto sviluppato e ai feedback forniti durante la fase di testing del sistema. Inoltre, durante il progetto, è stato portato in evidenza TensioBot un bot di Telegram molto simile a quello sviluppato e del quale si possono analizzare e confrontare i risultati ottenuti.

### 5.1 Considerazioni sul sistema

Lo scopo principale di questo progetto era quello di migliorare il chatbot sviluppato durante il progetto AI4Hypertension e di fornire nuovi strumenti al personale medico per aiutarli nello svolgimento del loro lavoro. Sotto questo punto di vista si può dire che il progetto ha raggiunto gli obiettivi prefissati, disponendo ora di un chatbot sufficientemente robusto e una dashboard sviluppata seguendo i suggerimenti forniti e, anche durante la fase di testing interno, i risultati sono stati soddisfacenti. Tuttavia ci sono ancora alcune aggiunte che potrebbero rendere l'uso più piacevole soprattutto per quanto riguarda l'uso effettuato da parte dei pazienti. Si è parlato con il Dott. Martino Pengo di come rendere il chatbot meno meccanico e alcune delle proposte venute fuori riguardano soprattutto il fatto di rendere i dialoghi più informali e di fornire risposte variabili all'utente migliorando l'illusione che l'interlocutore con cui sta parlando sia una persona e non un programma. Per quanto riguarda la dashboard la parte sviluppata è piuttosto completa e funzionale ma potrebbero essere comunque aggiunte alcune funzioni come l'import delle misurazioni e una gestione degli utenti migliore creando delle liste di pazienti seguiti in modo da rendere più agevole la ricerca dei propri pazienti. Si potrebbe inoltre dare la possibilità all'utente medico di visualizzare informazioni sulle misure dei propri pazienti direttamente tramite Telegram facendo richieste specifiche fornendo il codice del paziente. Riprendendo sempre l'esempio esposto prece-

dentemente, dove una linea telefonica veniva utilizzata per fornire assistenza da remoto, sono riportate nell'articolo alcune stime sui costi di tale soluzione che, durante il servizio, sono stati pari a 166 euro per ogni paziente con una media giornaliera di 2,1 euro per ogni paziente. Questi costi comprendono anche il rimborso effettuato alle infermiere e l'intervento di consulenza eseguito dai medici specializzati, tuttavia, una delle voci più costose è quella relativa all'affitto di strumentazione, software e connessione dati che ammonta a circa 76 euro per paziente [2]. In un possibile deployment di questo progetto il costo sarebbe quasi interamente dato dall'affitto della macchina virtuale fornita da AWS che rimane comunque intorno ai 100 euro mensili per una macchina molto più performante di quella utilizzata durante il progetto e che dispone di 8 vCore, 16GiB di RAM e 64GiB di archiviazione. Per fare un confronto, quella utilizzata durante il progetto, disponeva solamente di 1vCore, 1GiB di RAM e 8GiB di archiviazione. Anche se i due sistemi non sono direttamente paragonabili in quanto forniscono servizi diversi tra loro questa considerazione può essere utilizzata per rendersi conto di come una soluzione come quella sviluppata sia sicuramente accettabile rispetto ai costi degli altri sistemi di telemedicina.

## 5.2 TensioBot

Proprio durante lo svolgimento del progetto è stato pubblicato un interessante articolo nel quale si analizza l'uso di un bot di Telegram sviluppato dall'università dei paesi Baschi a supporto dei pazienti affetti da ipertensione arteriosa. I problemi che il team di TensioBot si è trovato ad affrontare sono gli stessi che sono stati visti durante questo progetto e, in particolare, anche loro affermano che spesso è richiesto ai pazienti di misurare i propri valori di pressione a casa compilando poi una tabella cartacea con data, ora e i dati della misurazione da consegnare poi durante il successivo appuntamento con il medico. Alcuni pazienti dimenticano di effettuare le misurazioni o le fanno agli orari sbagliati e capita che si dimenticano di portare la tabella con le misure durante la visita impedendo al medico di visualizzare i dati per un periodo di tempo ancora più lungo. Inoltre, anche in caso il paziente non effettua nessuna dimenticanza, il dottore deve trascrivere tutti i dati dalla tabella cartacea in un software informatico cosa che, secondo il parere del team di TensioBot, è una procedura lenta e soggetta ad errori. Per ovviare a questo il bot sviluppato invia ai pazienti notifiche per ricordargli di misurare la pressione, informa gli utenti riguardo alle buone pratiche per effettuare correttamente le misurazioni, presenta il grafico di tutti i dati inseriti e permette al personale medico di tenere sotto controllo i propri pazienti. Tuttavia, a differenza del chatbot

utilizzato in questo progetto, il bot creato dall'università dei paesi Baschi non utilizza alcuna intelligenza artificiale per l'interpretazione dei messaggi inviati dagli utenti e fa ricorso all'uso delle tastiere personalizzate che Telegram mette a disposizione. A meno che non venga scritto esattamente lo stesso testo presente nei comandi riconosciuti qualsiasi messaggio inviato dall'utente viene ignorato e questo forza l'utente ad accedere alle funzioni offerte dal bot solamente attraverso percorsi prestabiliti in modo piuttosto rigido e meccanico. D'altronde, essendo l'unico metodo pratico per interagire con il bot, le tastiere personalizzate di Telegram sono state ben curate e utilizzate in tutte le funzioni messe a disposizione al contrario del chatbot realizzato nel progetto che utilizza tali tastiere solamente in alcuni casi.

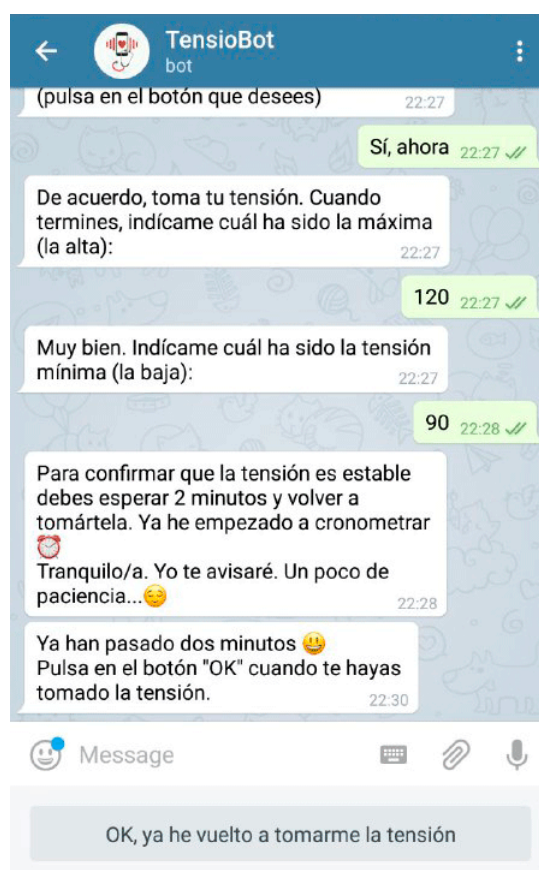


Figura 5.1: Schermata di TensioBot

Molto interessanti sono i risultati dei test che hanno eseguito, nel corso di due anni (tra il 2018 e il 2020), su 112 pazienti dei quali 55 utilizzavano il bot e i restati 57 appartenevano al gruppo di controllo utilizzando i metodi tradizionali. Dei 112 pazienti 45 soffrivano di ipertensione primaria, 54 di ipertensione

secondaria e 13 avevano un sospetto di ipertensione mentre l'età media di tutti i soggetti era di circa 52 anni con il paziente più giovane che aveva 21 anni e il più anziano 87. Durante la prima visita, i pazienti appartenenti al gruppo di controllo, ricevevano documenti con informazioni generali su come effettuare le misurazioni correttamente e venivano informati sul prossimo appuntamento, al contrario, i pazienti assegnati al gruppo del bot non hanno ricevuto informazioni sulla procedura e il personale medico li ha invece aiutati ad installare l'applicazione di Telegram e avviare il bot. Durante la prima visita tutti i pazienti sono stati sottoposti ad un test per verificare le loro conoscenze e, gli 88 pazienti che hanno effettuato una seconda visita, sono stati sottoposti nuovamente ad un test per valutare il miglioramento delle loro conoscenze. Entrambi i gruppi hanno mostrato un notevole miglioramento nelle loro conoscenze anche se, secondo i risultati ottenuti, il gruppo che ha utilizzato il bot ha ottenuto miglioramenti maggiori. Dopo la seconda visita tutti i pazienti hanno utilizzato un monitor Holter per valutare i loro valori medi di pressione nelle successive 24 ore e, rispetto al valore ottenuto in questo modo, le misure effettuate in modo autonomo dei pazienti che hanno utilizzato il bot si discostavano meno da tale valore anche se, è stato poi ritenuto che tale differenza non è significativa se confrontata con quella del gruppo che effettuava misure in modo autonomo utilizzando i metodi tradizionali. A 40 dei pazienti che hanno utilizzato il bot sono state poi poste alcune domande per valutare il grado di soddisfazione nell'uso di TensioBot. 92,5% dei pazienti ha ritenuto il bot molto semplice o comunque abbastanza semplice da utilizzare. Tutti hanno valutato il bot come abbastanza utile e tra questi il 72,5% lo hanno addirittura considerato molto utile. Sempre il 92,5% ha affermato che preferisce utilizzare TensioBot per effettuare le misurazioni in modo autonomo rispetto alla registrazione cartacea. Solo 6 persone, che corrispondono al 15%, hanno smesso di utilizzare il bot dopo l'esperimento e la ragione principale risiede nel fatto che trovano difficoltà in generale nell'uso di uno smartphone. Secondo i dati da loro raccolti non è stata notata una significativa differenza nel numero di misurazioni effettuate dai pazienti e quindi, nonostante il chatbot agisca in modo attivo notificando gli utenti, entrambi i metodi hanno avuto la stessa efficacia in questo ambito al contrario di quanto ci si sarebbe potuto aspettare. Uno tra i vantaggi principali che è notato dal team usando il bot è stato quello di semplificare sia il lavoro del personale medico che non deve più trascrivere le misurazioni nel sistema informatico sia la procedura di consegna dei dati da parte del paziente che ora non deve più ricordarsi della tabella cartacea e ha meno necessità di dover trascorrere del tempo nelle strutture sanitarie. Inoltre, anche secondo il loro parere, avere i dati già inseriti nel sistema informatico permette di accorgersi più rapidamente di valori fuori dalla norma non dovendo appunto aspettare la visita con il paziente per visionare le misurazioni che,



come già detto, può essere ulteriormente posticipato per la dimenticanza della tabella cartacea [6]. Nonostante i due progetti siano stati sviluppati in modo indipendente essi presentano alcuni elementi in comune che possono essere considerati come una conferma della giusta direzione presa in diversi punti di questo progetto.



# Conclusioni

Solo negli ultimi anni stanno nascendo chatbot per assistere il paziente iperteso e questo progetto si colloca quindi in un campo di applicazione piuttosto nuovo. Gli obiettivi prefissati con l'aiuto degli esperti nel settore sono stati raggiunti ed è sicuramente soddisfacente vedere di come il sistema possa essere considerato abbastanza maturo da avviare le procedure per un trial clinico. Anche se le prove effettuate sul sistema sono state svolte internamente, vedendo le similarità rispetto ad altri progetti simili ci si può aspettare che le decisioni prese durante questo progetto siano state adeguate. Alla fine del percorso svolto si dispone infatti di un chatbot Telegram sufficientemente robusto per interpretare correttamente le richieste degli utenti e che dispone di funzionalità pensate non solo per assistere il paziente ma anche per rendere il lavoro del personale medico più semplice. Sicuramente il contributo maggiore fornito da questo progetto è stato però quello di realizzare una nuova piattaforma web che possa permettere al medico di supervisionare i propri pazienti da remoto e in qualsiasi momento. Verranno ora analizzate le parti che possono essere il punto di partenza per ulteriori miglioramenti da sviluppare in futuro. A livello di chatbot purtroppo la parte di intelligenza artificiale è forse quella che al momento limita maggiormente le possibilità del progetto e un'eventuale cambio di IA potrebbe rendere il sistema più gradevole durante l'uso rispondendo in maniera più naturale rispetto alle risposte ripetitive che sono al momento programmate. Inoltre come già detto precedentemente nonostante le nuove possibilità di estrazione dei dati e delle relative medie non è ancora stato aggiunto un modo per il personale medico di richiedere le misure dei pazienti direttamente da Telegram e la lettura dei valori deve ancora essere eseguita sul dispositivo del paziente. Tuttavia queste nuove aggiunte potrebbero rivelarsi utili nel caso il paziente dimentichi di portare con sé i documenti cartacei all'appuntamento cosa che, come verificato nei chatbot già sviluppati, rappresenta uno dei problemi che questo sistema cerca di risolvere. Uno spunto interessante è sicuramente quello portato da TensioBot che, nonostante non faccia uso di intelligenze artificiali, utilizza in modo migliore le tastiere personalizzate di Telegram per cercare di rendere semplice l'uso del chatbot agli utenti. Per quanto riguarda il sito i risultati ottenuti sono soddisfacenti

dato che, a parte alcuni miglioramenti, potrebbe essere considerato un sistema utilizzabile. Sicuramente la notevole collaborazione con il Dott. Martino Pengo attraverso diversi confronti ha reso possibile realizzare una piattaforma web che possa soddisfare le necessità di chi deve effettivamente utilizzare tale strumento durante il proprio lavoro. Un altro aspetto da considerare, soprattutto in vista di uno studio pilota, è quello delle capacità computazionali della macchina che ospita il chatbot e il web server per la dashboard infatti, durante il progetto è stato sufficiente utilizzare la versione di prova gratuita fornita da AWS, mentre per un utilizzo più intensivo sarebbe consigliabile valutare una configurazione più performante.

# Ringraziamenti

Ringrazio il professore Alessandro Ricci e la professoressa Sara Montagna i quali hanno reso possibile la mia partecipazione a questo progetto e che mi hanno sempre seguito e consigliato durante lo svolgimento sia del percorso di tirocino che di tesi progettuale. Ringrazio anche il dottor Martino Pengo il quale lavora in prima persona per risolvere questo importante problema e che durante lo svolgimento del progetto ha contribuito fornendo suggerimenti e pareri su come migliorare e rendere utile il sistema.



# Bibliografia

- [1] Communication from the commission to the european parliament, the council, the european economic and social committee and the committee of the regions on telemedicine for the benefit of patients, healthcare systems and society, 04 11 2008.
- [2] Bertacchini Fabio Rivadossi Francesca Muiesan Maria Lorenza Bernocchi Palmira, Scalvini Simonetta. Home based telemedicine intervention for patients with uncontrolled hypertension—a real life non-randomized study. *PubMed*, 14:52, 2014.
- [3] Matteo Castellucci. Applicazione di tecniche ai per la progettazione di un sistema a supporto del paziente iperteso, 2019-2020.
- [4] Ministero della Salute. Alleanza italiana per le malattie cardio-cerebrovascolari. documento di strategia, 18 maggio 2017.
- [5] SIIA Società Italiana dell’Ipertensione Arteriosa. Ipertensione: i numeri in italia.
- [6] Ramón Saracho Leyre Echeazarra, Juanan Pereira. Tensiobot: a chatbot assistant for self-managed in-house blood pressure checking. 45:54, 2021.
- [7] Eri Maeda, Kyoko Nomura Yukiyo Kumazawa Hiromitsu Shirasawa Hidekazu Saito Akane Miyata, Jacky Boivin, and Yukihiro Terada. Promoting fertility awareness and preconception health using a chatbot: a randomized controlled trial. *Reproductive BioMedicine Online*, 41(6):1133–1143, 2020.
- [8] Sayali Randive Nivedita Bhirud, Subhash Tataale and Shubham Nahar. A literature review on chatbots in healthcare domain, Luglio 2019.
- [9] Ivana Ostoic. Megi app - initiative summary, 2021-01-20.
- [10] Titti Santamato. Con cambio privacy whatsapp boom telegram e signal, 14 gennaio 2021.

- [11] Kyaw Bhone Myint Kowatsch Tobias Joty Shafiq Theng Yin-Leng Atun Rifat Tudor Car Lorainne, Dhinagaran Dhakshenya Ardhithy. Conversational agents in health care: Scoping review and conceptual analysis. 22:e17158, 2020.