

**ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA
CAMPUS DI CESENA**

DIPARTIMENTO DI INFORMATICA - SCIENZA E INGEGNERIA

**CORSO DI LAUREA MAGISTRALE IN
INGEGNERIA E SCIENZE INFORMATICHE**

**PROGETTAZIONE E PROTOTIPAZIONE DI
UNA PIATTAFORMA CLOUD IBRIDA PER
IL MONITORAGGIO DELLA SALUTE**

**Tesi in
PERVASIVE COMPUTING**

**Relatore:
Prof. ALESSANDRO RICCI**

**Presentata da:
DANIELE SCHIAVI**

**Correlatori:
Dott. VINCENZO SATURNINO
Dott. MAURO GATTI**

Anno Accademico 2019-2020

Indice

Introduzione	vii
1 Contesto di riferimento	1
1.1 Telemedicina	3
1.2 Telemonitoraggio	6
1.3 Obiettivo del progetto	10
2 Stato dell'arte	13
2.1 Telemedicina	13
2.2 Telemonitoraggio	14
2.3 Tecnologie innovative	16
3 Definizione ed analisi dei requisiti	17
3.1 Requisiti	17
3.1.1 Architettura di riferimento	17
3.1.2 Piattaforma caso d'uso Covid-19	19
3.2 Analisi dei requisiti	21
3.2.1 Raccolta dati	22
3.2.2 Sicurezza e privacy dei dati	23
3.2.3 Caso d'uso Covid-19	24
3.3 Scenari caso d'uso Covid-19	29
4 Disegno architetturale	35
4.1 Raccolta dei dati	36
4.1.1 Smartphone	38

4.1.2	Piattaforme cloud	40
4.1.3	Componente edge	41
4.2	Trasformazione dei dati	44
4.2.1	Dispatcher	46
4.2.2	Mapper	47
4.3	Memorizzazione ed applicazioni	49
4.4	Sicurezza e privacy	50
4.5	Architettura	53
4.6	Mockup caso d'uso Covid-19	57
4.6.1	Applicazione del paziente	57
4.6.2	Applicazione del medico	59
5	Sviluppo prototipale e validazione	61
5.1	Framework e tecnologie	63
5.1.1	Apache Kafka	63
5.1.2	RabbitMQ	64
5.1.3	Spring	66
5.1.4	Apache Camel	67
5.2	Deploy	68
6	Conclusioni	71
6.1	Sviluppi futuri	72
	Bibliography	75

Elenco delle figure

1.1	Panoramica generale della piattaforma	11
2.1	Architettura a tre strati	15
3.1	Scenari caso d'uso Covid-19	29
4.1	Panoramica dei componenti della piattaforma	35
4.2	Panoramica metodologie di comunicazione dei dispositivi	38
4.3	Processo integrazione cloud-to-cloud tramite OAuth 2.0	40
4.4	Componente edge: modello dei componenti	42
4.5	Componente edge: architettura	44
4.6	Modello dei dati	45
4.7	Esempio di configurazione del Dispatcher	47
4.8	Esempio di configurazione del Mapper	49
4.9	Architettura di riferimento della piattaforma	54
4.10	Mockup applicazione del paziente: schermata dispositivi connessi	57
4.11	Mockup applicazione del paziente: schermata questionari e test effettuati	58
4.12	Mockup applicazione del medico: schermata panoramica dei pazienti . . .	59
4.13	Mockup applicazione del medico: schermata dettaglio del paziente sele- zionato	60
5.1	Panoramica dell'architettura utilizzata per lo sviluppo prototipale	62
5.2	Utilizzo di RabbitMQ per la comunicazione tra Dispatcher, Mapper e Worker	64

Elenco delle tabelle

3.1	Parametri caso d'uso Covid-19	26
3.2	Informazioni sulle attività fisiche caso d'uso Covid-19	27
3.3	Punteggi per il calcolo dell'indicatore sintetico	28

Introduzione

Internet of Things è un concetto che diventa ogni giorno più pervasivo, portando un contributo in molti settori come nella domotica per quanto riguarda la gestione della casa, nell'industria 4.0 per quanto riguarda l'automazione delle linee di produzione delle fabbriche e molti altri. Tra questi negli ultimi anni si afferma sempre più il suo utilizzo nel settore sanitario adottando diversi nomi, come Medicina 4.0, Internet of Medical Things (IoMT) o Healthcare Internet of Things (H-IoT); questa applicazione in ambito medico può essere pensata come un'infrastruttura connessa di dispositivi medici, applicazioni software, sistemi e servizi sanitari la cui adozione potrebbe portare diversi vantaggi, soprattutto per malattie tempo-dipendenti e patologie croniche, oltre a permettere l'individuazione ed identificazione delle malattie con un anticipo considerevole.

Lo sviluppo tecnologico nel campo informatico e delle telecomunicazioni ha portato alla scoperta di nuove tecnologie abilitanti per la trasformazione della medicina tradizionale in una nuova disciplina chiamata telemedicina; questa consiste nell'utilizzo di reti di telecomunicazione, software, dispositivi digitali ed internet per erogare servizi sanitari a distanza, senza che sia necessaria quindi la presenza di medici e pazienti nello stesso luogo fisico. Alcune delle tecnologie abilitanti che hanno permesso questa evoluzione sono: l'intelligenza artificiale per l'analisi di pattern di dati, il paradigma fog/edge che permette di avvicinare la potenza computazionale ai dispositivi che si occupano della raccolta dati, i big data per la gestione di enormi moli di dati, l'IoT per la realizzazione di reti di dispositivi intelligenti interconnessi e molte altre.

Il bisogno di sviluppo tecnologico nel campo della medicina per consentire la fruizione di servizi sanitari a distanza viene accentuato notevolmente e viene messo in evidenza in un periodo come quello attuale, cioè quello che segue la diffusione del virus Covid-19,

attualmente considerato una pandemia a livello mondiale, nel quale il distanziamento sociale gioca un ruolo fondamentale per limitarne la propagazione.

Una delle applicazioni della telemedicina è il telemonitoraggio, cioè il monitoraggio a distanza dei parametri vitali relativi ad un paziente attraverso dispositivi medici specifici; in questo caso vengono utilizzati dispositivi intelligenti che permettono la raccolta, registrazione, elaborazione e trasmissione di questi parametri, come ad esempio la pressione sanguigna, la temperatura, la saturazione di ossigeno nel sangue, la glicemia e molti altri valori essenziali per una valutazione ottimale dello stato di salute di un individuo.

Ci sono già molti dispositivi commerciali, come ad esempio termometri, sfigmomanometri, pulsiossimetri, smartband, che consentono di misurare alcuni parametri ed in caso di valori considerati allarmanti notificano il paziente con un alert sullo smartphone. Questi dispositivi sono spesso gestiti tramite l'applicazione del produttore del device che viene installata sullo smartphone dell'utente e permette la raccolta e la gestione dei dati relativi solo a quel determinato vendor.

Questa tesi è ambientata in questo particolare contesto e tratta la progettazione di un'architettura di riferimento e lo sviluppo prototipale della piattaforma per la raccolta di dati sanitari, relativi a diversi pazienti, attraverso l'utilizzo di dispositivi commerciali eterogenei, per poi renderli disponibili attraverso un modello standard per diverse tipologie di utilizzo. L'obiettivo è quello di riunire i dati raccolti da tali dispositivi, che solitamente sono disponibili solamente tramite l'applicazione del venditore del device o tramite l'utilizzo di aggregatori, in un'unica piattaforma, e di trasformarli per renderli conformi ad un modello comune in modo tale che vi si possa accedere in modo uniforme, indipendentemente dai dispositivi dai quali sono stati raccolti, dai vendor o dagli aggregatori dal quale potrebbero provenire. Sarà così possibile sfruttare questi dati per diversi tipi di applicazioni, come ad esempio per farne analitica, data processing, presentarli attraverso una interfaccia grafica o per la generazione di allarmi nel caso vengano verificate determinate condizioni. Il progetto è stato realizzato in collaborazione con il team Active Intelligence dell'azienda IBM durante un tirocinio curriculare; la mia esperienza in questa azienda è durata 6 mesi, da metà gennaio 2020 a metà luglio 2020, in concomitanza con il periodo iniziale di Covid-19 e le conseguenti restrizioni negli spostamenti. Tale situazione ci ha portati a lavorare inizialmente in presenza presso la sede di Bologna

dell'azienda per poi continuare da casa in smart working; proprio questa esperienza ci ha guidati verso un caso d'uso reale nel quale calare la piattaforma progettata.

Capitolo 1

Contesto di riferimento

In questo capitolo si cerca di fornire il contesto nel quale viene pensato e sviluppato questo progetto.

In molti paesi, la telemedicina, risulta una realtà molto diffusa per diverse tipologie di applicazioni, permettendo di operare a distanza per fornire prestazioni mediche di diverso tipo; in Italia vi sono molteplici esperienze distribuite in tutto il territorio nazionale ma la situazione risulta ancora limitata se paragonata agli altri paesi. La diffusione della Telemedicina è notevolmente rallentata a causa della burocrazia Italiana, inoltre vi è un avanzamento tecnologico nel settore medico altamente eterogeneo e differenziato tra le regioni Italiane, per questo motivo il Ministero ha istituito un Osservatorio Nazionale con il compito di uniformare le reti di Telemedicina di tutto il Paese e fornire dei modelli di riferimento da seguire per migliorare l'accessibilità e l'efficacia dei servizi forniti.

L'emergenza Covid-19 che ha colpito il mondo intero da inizio anno 2020 ha messo in evidenza ritardi e lacune nel sistema sanitario Italiano, ma allo stesso tempo le sue caratteristiche di diffusione hanno dato un'enorme spinta all'adozione della Telemedicina per la gestione dei pazienti a distanza, con l'obiettivo di ridurre al minimo il contatto tra le persone e di conseguenza limitare la propagazione del virus. Utilizzare tecnologie informatiche e di telecomunicazione innovative per fornire servizi a distanza però non è l'unico fattore da prendere in considerazione ed utilizzare per ottenere un buon livello di servizio sanitario, ma risulta necessario che tutti i sistemi utilizzati siano uniformati e connessi tra loro in modo tale che possano scambiarsi informazioni in tutto il territorio

e garantire quindi la continuità delle cure. Alcune federazioni e associazioni come Federsanità hanno proposto delle buone pratiche sulle soluzioni innovative come possiamo vedere nel documento [3], l'idea è di effettuare un confronto nazionale tra le aziende associate con l'obiettivo di individuare proposte per ridisegnare i modelli di cura al fine di risolvere tali problematiche e migliorare i servizi in ambito sanitario. Un'altra problematica molto rilevante nell'adozione di tecniche di Telemedicina in questo periodo riguarda gli utenti che ne dovrebbero trarre più profitto, cioè le persone anziane, quindi quelle che risultano essere più a rischio; i problemi di questi individui nell'utilizzo di servizi sanitari a distanza sono molteplici, come si può notare nello studio riportato in [8], ma è possibile individuare quelli più importanti e frequenti nella mancanza di preparazione a livello tecnologico e in problematiche di vista, udito o difficoltà comunicative.

Tra le nuove applicazioni della Telemedicina in diverse discipline adottate a seguito della pandemia sono stati effettuati studi per verificare quanti pazienti hanno partecipato a questa nuova tipologia di erogazione dei servizi sanitari ed il loro livello di soddisfazione; tra questi sondaggi possiamo trovare un esempio italiano di utilizzo in un centro bariatrico [11], uno indiano presso un centro di cure palliative per malati di cancro [2] ed uno americano nel campo ortopedico [10].

Il campo della Telemedicina di interesse per questa tesi è quello del telemonitoraggio, ossia del monitoraggio remoto dei parametri vitali dei pazienti attraverso dispositivi intelligenti connessi che ne consentono la misurazione e la registrazione presso dispositivi locali, quali smartphone o tablet, o presso piattaforme di memorizzazione cloud del venditore del device. Sono presenti sul mercato un'infinità di dispositivi commerciali di questa tipologia, appartenenti a molti produttori differenti; i dati raccolti solitamente vengono utilizzati attraverso le loro applicazioni o attraverso cloud proprietari, questo porta ad un problema di aggregazione dei dati forniti da dispositivi di venditori differenti perchè richiedono l'utilizzo di diversi software e non si ha una visione generale dello stato salutare dell'individuo.

1.1 Telemedicina

Non vi è una definizione univoca di Telemedicina ma ne esistono di diverse; il termine vede i suoi primi utilizzi negli anni 60-70 ed ha subito notevoli cambiamenti nel tempo, in seguito sono riportate le due definizioni considerate più rilevanti.

Definizione adottata dall'Organizzazione Mondiale della Sanità (OMS)[6]:

Telemedicine is delivery of health care services, where distance is a critical factor, by all health care professionals using information and communication technologies for the exchange of valid information for diagnosis, treatment and prevention of disease and injuries, research and evaluation, and for the continuing education of health care providers, all in the interests of advancing the health of individuals and their communities.

Definizione fornita dal Ministero della Salute nel documento *Telemedicina - Linee di indirizzo nazionali* del 2012[13]:

Per Telemedicina si intende una modalità di erogazione di servizi di assistenza sanitaria, tramite il ricorso a tecnologie innovative, in particolare alle Information and Communication Technologies (ICT), in situazioni in cui il professionista della salute e il paziente (o due professionisti) non si trovano nella stessa località. La Telemedicina comporta la trasmissione sicura di informazioni e dati di carattere medico nella forma di testi, suoni, immagini o altre forme necessarie per la prevenzione, la diagnosi, il trattamento e il successivo controllo dei pazienti. [...]

In entrambe le definizioni possiamo individuare gli stessi punti chiave e concetti che rappresentano la Telemedicina, cioè un insieme di attività, operazioni e servizi in ambito medico che necessitano di essere eseguiti a **distanza**, senza che sia richiesta la presenza fisica delle parti, tramite l'utilizzo di tecnologie informatiche o di telecomunicazione (**ICT**); questa definizione non è limitata solamente all'interazione remota tra personale medico e pazienti ma può essere applicata anche nei casi di consultazione tra professionisti, senza che il paziente sia presente.

Come è possibile immaginare, l'applicazione della Telemedicina comporta molti vantaggi sotto diversi aspetti:

- consente il monitoraggio delle malattie, in particolare quelle croniche, e dell'aderenza alla cura prescritta, per avere un quadro della situazione in tempo reale;
- consente l'erogazione di servizi sanitari a distanza, senza che sia richiesta la presenza fisica dei partecipanti nello stesso luogo;
- aumenta l'efficacia nella gestione e cura dei pazienti, la qualità dell'assistenza e garantisce la continuità delle cure;
- permette di avere cure personalizzate senza il bisogno di recarsi spesso dal medico;
- permette di avere equità di accesso all'assistenza sanitaria;
- riduce il numero delle visite presso il proprio medico curante, i Pronto Soccorsi e gli ambulatori;
- permette di effettuare analisi sui dati raccolti per scoprire correlazioni tra cause e malattie;
- riduce i tempi per identificare una malattia dai primi sintomi;
- riduce i costi per la cura del paziente ed in generale per tutto il settore della sanità.

La Telemedicina entra a far parte del Sistema Sanitario Nazionale (SSN) grazie all'approvazione delle *Indicazioni nazionali per l'erogazione di prestazioni in telemedicina* da parte del Ministero della Salute alla Conferenza Stato-Regioni tenuta giovedì 17 Dicembre 2020. In questo documento vogliono essere fornite indicazioni da adottare per l'erogazione di prestazioni di telemedicina, in particolare le applicazioni trattate sono la televisita, la teleconsulenza medico-sanitaria, il teleconsulto medico, la teleassistenza e la teleriferazione; vengono invece tralasciate le altre applicazioni che, come si può leggere nel documento in questione, verranno trattate in documenti successivi.

La Telemedicina può essere utilizzata in diversi campi medici specifici assumendo declinazioni come telecardiologia, teleradiologia, telepatologia, teledermatologia, teleoftalmologia e molti altri; inoltre le applicazioni possibili sono molteplici, in seguito ne vengono individuate e descritte alcune tra quelle considerate più rilevanti:

- televisita, si tratta di una visita del paziente da parte del medico curante o di uno specialista, effettuata a distanza ed in tempo reale;
- teleconsulto medico, consiste nella discussione a distanza sulla situazione clinica di un paziente da parte di due o più professionisti, può essere svolta sia in modo sincrono che asincrono e vi può partecipare anche il paziente;
- teleconsulenza medico-sanitaria, consiste in una videochiamata tra due o più individui appartenenti al settore sanitario e/o medico a seguito di una richiesta di supporto da uno di essi per ottenere indicazioni su una decisione da prendere o sulle azioni assistenziali da eseguire, anche in questa attività non è obbligatoria la presenza del paziente;
- teleassistenza da parte di professioni sanitarie, si tratta di assistenza a distanza fornita dal personale sanitario al paziente o ad un suo caregiver, per guidarlo nel corretto svolgimento di attività assistenziali attraverso l'utilizzo di una videochiamata e/o di apposite applicazioni;
- telerefertazione, è una relazione digitale trasmessa tramite sistemi di telecomunicazione che viene rilasciata dal medico a seguito di un esame clinico o strumentale;
- teleriabilitazione, si tratta dell'erogazione di servizi di riabilitazione a distanza per un paziente o un caregiver da parte di un professionista, può essere svolta in modo sincrono tramite una comunicazione audio-video o asincrono tramite l'utilizzo di applicazioni o la condivisione di materiale;
- telemonitoraggio, consiste nel monitoraggio dei parametri vitali di un paziente da parte di un professionista attraverso dispositivi medici specifici, utile in molte situazioni come per mantenere controllato lo stato di salute in caso di malattie croniche, patologie o per verificare l'aderenza di una terapia o di una riabilitazione.

1.2 Telemonitoraggio

Il telemonitoraggio è una tipologia di servizio sanitario che viene erogato a distanza grazie all'utilizzo di strumenti tecnologici e consiste nel monitoraggio remoto dello stato di salute di un paziente attraverso la raccolta di informazioni, come ad esempio alcuni parametri vitali. I dati possono essere raccolti in modo autonomo direttamente dai pazienti che si trovano al proprio domicilio, tramite l'utilizzo di dispositivi che hanno la possibilità di condividere tali dati in rete per essere poi memorizzati, analizzati, elaborati e condivisi con i medici curanti; possono inoltre essere generati allarmi per avvisare i medici in modo da intervenire tempestivamente in caso di anomalie nei dati rilevati. Questo tipo di servizio può risultare molto utile in diverse situazioni, come ad esempio per il controllo di malati cronici o anziani, o la verifica dell'andamento della terapia prescritta al paziente per verificarne l'appropriatezza e la corretta assunzione dei farmaci. Il telemonitoraggio è la branca della telemedicina nel quale è situato lo sviluppo di questa tesi.

Nel documento [12] vengono forniti alcuni esempi di applicazioni del telemonitoraggio come il controllo del glucosio nel sangue per pazienti diabetici, la temperatura corporea, la pressione sanguigna e l'elettrocardiogramma; inoltre è stata condotta un'indagine attraverso un questionario per individuare il punto di vista degli intervistati sul tema del sistema sanitario intelligente e la risposta è risultata positiva.

Un importante aiuto e miglioramento nel campo del monitoraggio è fornito dalle tecniche di machine learning, che hanno un grande potenziale se utilizzate in sistemi sanitari basati su IoT; questi algoritmi sono in grado di supportare i medici e gli operatori sanitari nel proprio lavoro permettendogli di monitorare, diagnosticare, individuare il problema e proporre una soluzione accurata nel minor tempo possibile. L'utilizzo di tali tecniche ha permesso l'automatizzazione dell'analisi dei dati, che prima veniva effettuata manualmente da un medico o da personale sanitario, abilitando il monitoraggio continuo dei dati, riducendone drasticamente i tempi di ispezione ed aumentando il numero di pazienti che è possibile controllare; sfruttando questi algoritmi è possibile analizzare i dati e, nel caso di anomalie, allertare i medici, il personale sanitario o i caregiver attraverso un sistema di allarme. Altri esempi di utilizzo di tali tecniche nel campo della telemedicina li possiamo trovare nella procedura di diagnosi utilizzata per prevedere

la gravità di una malattia, come si può vedere in [14] che effettua una esplorazione sulla tecnica di classificazione nell'approccio di monitoraggio e diagnosi della salute, nei sistemi di supporto alle decisioni per aiutare nel processo decisionale o nell'analisi di dati provenienti da sensori indossabili per l'activity recognition. Alcuni degli algoritmi di machine learning applicabili nel campo della telemedicina possiamo trovarli in [4].

Le informazioni possono essere acquisite in diversi modi a seconda dello strumento utilizzato per la raccolta dati, in particolare è possibile suddividere logicamente tali metodi nelle seguenti casistiche:

- Utilizzando reti di sensori indossabili. Risultano la tipologia di dispositivi di raccolta dati più invasivi perchè devono essere indossati per lunghi periodi per svolgere il loro compito. Tali sensori solitamente non hanno una grande potenza computazionale ed il loro unico compito è quello di misurare parametri fisiologici o ambientali, ed in alcuni casi effettuare semplici operazioni di pre-processamento prima di trasmetterli ad altri dispositivi connessi nella stessa rete locale; questi altri device sono connessi sia alla rete locale utilizzata dei sensori che alla rete esterna ed il loro scopo è quello di aggregare tutti i dati forniti dai sensori, eventualmente effettuare analisi o elaborazioni, e trasferirli verso servizi esterni, come ad esempio data center o piattaforme cloud, per poi essere elaborati e memorizzati. Le reti locali dei sensori solitamente sono Wireless Sensor Network (WSN) o Wireless Body Area Network (WBAN), cioè reti formate da dispositivi con una connessione a corto raggio che sono in grado di comunicare con i vicini per condividere le proprie informazioni o propagare quelle ricevute da altri partecipanti della rete. È possibile trovare un esempio di questa tipologia di raccolta dati nel paper [5], dove viene preso come caso di studio il monitoraggio delle patologie Parkinson e Huntington, che sono caratterizzate da disturbi del movimento; per effettuare il monitoraggio viene utilizzata una rete composta da 5 sensori applicati in posizione specifiche del paziente in modo tale da raccogliere dati riguardo la loro posizione.
- Attraverso test e sondaggi remoti. Essi hanno lo scopo di verificare lo stato di un paziente rispetto ad una problematica o patologia e di valutarne il livello di gravità. Alcuni esempi di test sono il "walking test", che consiste nel far camminare il paziente per un certo periodo di tempo ed effettuare periodicamente misure di

alcuni parametri vitali di interesse come la frequenza cardiaca e la saturazione di ossigeno nel sangue, e lo "speech test", che consiste nel far leggere un testo al paziente o farlo contare fino a quando non termina l'ossigeno per valutare quante parole è riuscito a leggere o fino a che numero è riuscito a contare. Un possibile esempio di sondaggio invece è il test di valutazione della broncopneumopatia cronica ostruttiva (BPCO), che è un questionario che valuta il livello dei sintomi e lo stato del paziente per questa particolare patologia.

- Utilizzando dispositivi situati nell'ambiente. Questa tipologia solitamente viene utilizzata per il monitoraggio delle attività e per l'analisi del comportamento di un individuo. I comportamenti e le attività quotidiane svolte sono buoni indicatori dello stato di salute di un individuo, inoltre permettono di rilevare situazioni anomale o di pericolo, come ad esempio il caso di caduta di una persona anziana; per ottenere tali informazioni il metodo più utilizzato è quello di applicare dispositivi visivi all'interno della casa del paziente o in strutture adibite a tale scopo. I dispositivi in questione solitamente sono sistemi di visione in grado di processare ed elaborare i dati in tempo reale a livello edge, in modo tale da condividere verso l'esterno solamente le informazioni inerenti all'attività identificata e non le immagini, per questioni legate alla privacy dell'utente.
- Utilizzando dispositivi intelligenti medici o commerciali. I dispositivi di questa categoria sono in grado di misurare i parametri vitali di un paziente e di salvarli in un dispositivo abbinato, ad esempio uno smartphone attraverso l'utilizzo di un'apposita applicazione, o in locale qualora la loro comunicazione non risulti momentaneamente disponibile. Attualmente è possibile trovare molti di questi device in commercio che permettono di ottenere misurazioni, con gradi diversi di precisione, di svariati parametri vitali; alcuni esempi di dispositivi sono i termometri, i pulsossimetri, gli spirometri, le bilance e molti altri. I dispositivi appartenenti a questa tipologia solitamente vengono utilizzati per effettuare misurazioni puntuali, consentendo un tipo di monitoraggio periodico su richiesta, ma ce ne sono alcuni che permettono il monitoraggio continuo come ad esempio gli smartwatch; i dati raccolti da tali device vengono resi disponibili appena possibile nell'applicazione

del venditore e, qualora fosse supportato, sulla piattaforma cloud dello stesso.

Una volta raccolti, i dati possono essere analizzati, elaborati e memorizzati; queste operazioni possono essere eseguite seguendo due approcci possibili:

- in locale, attraverso i dispositivi dell'utente come smartphone o tablet, o tramite componenti dedicati che possiamo chiamare componenti edge, in questo caso la computazione avviene vicino alla raccolta del dato e si parla di edge computing;
- in remoto, attraverso servizi distribuiti in piattaforme cloud e viene denominata cloud computing.

L'approccio più comunemente utilizzato è quello di effettuare le operazioni di analisi, elaborazione e memorizzazione in remoto ma vi sono casi, come quello precedentemente citato per il monitoraggio delle attività, nei quali è preferibile o necessario effettuare l'elaborazione e l'analisi dei dati localmente; i motivi per adottare questo secondo approccio possono riguardare la riservatezza dei dati o essere legati a limitazioni tecnologiche, come ad esempio nel caso di sistemi tempo-dipendenti, per ridurre i tempi di risposta eliminando la latenza della rete, oppure per effettuare una prima elaborazione dei dati per ridurre la dimensione prima di trasmetterli in rete e di conseguenza abbassare il livello di banda richiesta. Per quanto riguarda la memorizzazione in locale viene effettuata molto raramente, l'unico vantaggio che porta risiede negli aspetti relativi alla sicurezza e riservatezza dei dati ma è un approccio limitante e non consente di ottenere le caratteristiche, funzionalità e vantaggi ottenibili utilizzando le piattaforme cloud come la condivisione automatica, la possibilità di effettuare analitica o di processare i dati per diverse tipologie di applicazioni.

1.3 Obiettivo del progetto

L'obiettivo di questa tesi è la progettazione di un'architettura di riferimento per la realizzazione di una piattaforma ibrida in grado di raccogliere dati sanitari da dispositivi commerciali eterogenei provenienti da diversi venditori, di aggregarli sotto un modello standard comune e di renderli disponibili per molteplici applicazioni, come ad esempio per processarli, farne analitica per ottenere inferenza o per presentarli attraverso un'interfaccia web in modo da consentirne la visualizzazione ed il monitoraggio. Con il termine ibrida si vuole denotare il fatto che la piattaforma sarà composta da componenti che soddisfano paradigmi differenti; la parte che si occupa della raccolta dei dati, è pensata come applicazione nativa sullo smartphone dei pazienti ed ha lo scopo di collezionare localmente i dati dai dispositivi compatibili e propagarli verso la piattaforma cloud. La piattaforma cloud si occupa della trasformazione del dato per renderlo conforme ad un modello standard, della sua memorizzazione e della sua esposizione verso l'esterno. Infine vi è la parte di rappresentazione ed esposizione dei dati, che si è pensato fosse utile fornirla attraverso un'interfaccia web in modo da renderla disponibile sia da dispositivi mobili, smartphone o tablet, ma anche da notebook o workstation. Successivamente si vuole realizzare un prototipo sulla base dell'architettura prodotta, come Proof of Concept, per verificare che ciò che si è pensato sia veramente attuabile e funzionale; dato il periodo si è pensato di calare il prototipo nel caso d'uso Covid-19 ed il suo scopo è il telemonitoraggio dei parametri vitali dei pazienti, in modo da individuare nel minor tempo possibile il verificarsi dei primi sintomi della malattia, per pazienti sani, e per controllare lo stato di pazienti che hanno già contratto il virus.

Con questa piattaforma si vogliono supportare principalmente 3 tipologie di telemonitoraggio:

- su richiesta, nel quale il medico potrà guardare i dati in qualsiasi momento quando lo riterrà necessario, come ad esempio a seguito di una chiamata del paziente. In questa categoria non sono previste notifiche o allarmi, è il caso più semplice da realizzare ed è quello in cui sarà situato il caso d'uso Covid-19;
- continuo, dove si ipotizza che il paziente sia ospedalizzato a casa e debba essere costantemente monitorato, in questo caso i dati raccolti verranno analizzati nel

minor tempo possibile attraverso degli algoritmi che consentono la generazione di allarmi, qualora i valori fossero ritenuti anomali o critici;

- predittivo, nel quale i dati raccolti vengono analizzati ed utilizzati per sviluppare degli algoritmi in grado di prevedere l'evoluzione della patologia e generare quindi allarmi nel caso siano previsti dei peggioramenti.

In Figura 1.1 viene mostrata una visione generale della piattaforma ad un livello di astrazione molto elevato; partendo da sinistra è possibile trovare il paziente ed alcuni dei suoi dispositivi personali dal quale raccogliere dati relativi alla sua salute, nella parte centrale si vede la piattaforma cloud che si occuperà dell'aggregazione e trasformazione dei dati rendendoli conformi ad un modello standard, per poi esporli per diverse tipologie di applicazioni, alcune delle quali sono riportate nella parte destra della figura.

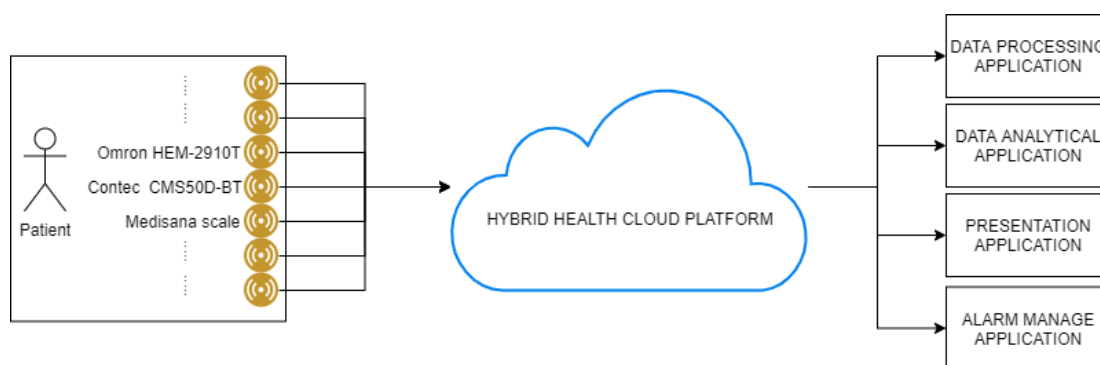


Figura 1.1: Panoramica generale della piattaforma

Capitolo 2

Stato dell'arte

2.1 Telemedicina

In [1] viene fornita una panoramica dello stato dell'arte e della diffusione della telemedicina nel mondo, facendo un confronto tra il periodo precedente alla diffusione di Covid-19 e quello attuale; tra i paesi all'avanguardia sulle tecniche e sulle invenzioni nel campo della telemedicina possiamo trovare gli Stati Uniti, anche se la diffusione ha riscontrato alcuni problemi legati alla differenza di adozione e dei quadri politici nei diversi stati, e Singapore. A seguito della diffusione del Covid-19 in quasi tutti i paesi si è notato un drastico aumento di utilizzo delle piattaforme che permettono la televisita e la teleconsulenza; in Cina l'implementazione di tecniche di telemedicina variano molto all'interno del territorio, a partire da applicazioni di basso livello, fino ad arrivare a tecnologie innovative come ad esempio nel campo della gestione della cura del paziente contactless, dove questa viene fornita tramite robot e dispositivi digitali.

Il Kazakistan ha implementato una rete nazionale di telemedicina già nel 2004, il cui scopo era quello di eliminare il divario nell'erogazione di cure mediche tra le popolazioni delle aree urbane e rurali; durante il periodo del Covid-19 è considerato un esempio dell'uso ottimale delle piattaforme di telemedicina ed un modello da seguire per paesi con una vasta area geografica ma bassa densità di popolazione.

In Italia il livello di adozione della telemedicina varia molto su tutto il territorio a causa dell'eterogeneità delle infrastrutture come ad esempio la copertura o la capacità

di connessione ad internet insufficienti o non appropriate; è possibile trovare alcune applicazioni come ad esempio il fascicolo sanitario elettronico (FSE), che consente la condivisione di referti e prescrizioni con il personale autorizzato come medici e farmacie, il sistema Telemed-CN2 [3] per la diagnosi, l'assistenza e la cura a distanza o la piattaforma TEL.TE.CoViD19 [15] per il telemonitoraggio di pazienti Covid.

2.2 Telemonitoraggio

Per quanto riguarda il caso di studio di questa tesi o più in generale il telemonitoraggio, è possibile trovare molteplici modelli di architetture e piattaforme che si pongono come obiettivo il monitoraggio a distanza dello stato di salute dei pazienti; tra questi modelli possiamo trovare anche quello fornito dalla IEEE Standard Association in [7]. Con lo standard proposto ci cerca di definire un'architettura per dispositivi medici PoC (Point-of-Care) distribuiti orientati ai servizi ed individuare i parametri di qualità del servizio che devono essere rispettati. Una tra le architetture più utilizzate e riconosciute nel campo del telemonitoraggio, che è possibile vedere in Figura 2.1, è rappresentata da una struttura a tre strati:

- strato delle cose, è quello situato più vicino al paziente ed in esso sono contenuti tutti i sensori, attuatori e dispositivi con i quali interagisce. I sensori sono componenti utilizzati per la raccolta dei dati come la misurazione dei parametri vitali ed in alcuni casi per una loro prima pre-elaborazione, gli attuatori invece hanno lo scopo di agire sull'ambiente o con il paziente, ad esempio per trasmettere un output o un segnale di allarme, infine per dispositivi si intendono tutti quelli in grado di comunicare localmente con i primi due ma che sono connessi anche con la rete esterna per consentire la condivisione dei dati con server o servizi cloud. Questi ultimi componenti si occupano di aggregare i dati raccolti dai sensori e trasmetterli verso piattaforme, server o servizi cloud remoti che si occuperanno della loro elaborazione, analisi e memorizzazione; tali operazioni in alcuni casi possono essere eseguite, in parte o interamente, già sull'hardware locale prima di trasferire i dati, riducendo significativamente il ritardo causato dal trasporto delle informazioni in rete. La comunicazione locale tra i dispositivi, i sensori e gli attuatori solitamente

avviene tramite l'utilizzo di tecnologie wireless a basso consumo energetico come Bluetooth Low Energy (BLE), ZigBee, Wi-Fi o RFID;

- strato di rete, si occupa di garantire che i dati rilevati vengano trasmessi al livello di elaborazione remoto. In questo strato si possono trovare i componenti hardware, le tecnologie ed i protocolli di comunicazione ad ampio raggio per la trasmissione dei dati dai dispositivi ai servizi remoti, in particolare si possono trovare tecnologie come 3G, 4G, GSM, LoRa, NBIoT o Sigfox;
- strato di processamento, è quello che comprende i server ed i servizi cloud remoti che si occupano dell'elaborazione, dell'analisi e della memorizzazione dei dati; in esso sono comprese anche le applicazioni ed i servizi che utilizzano tali informazioni.

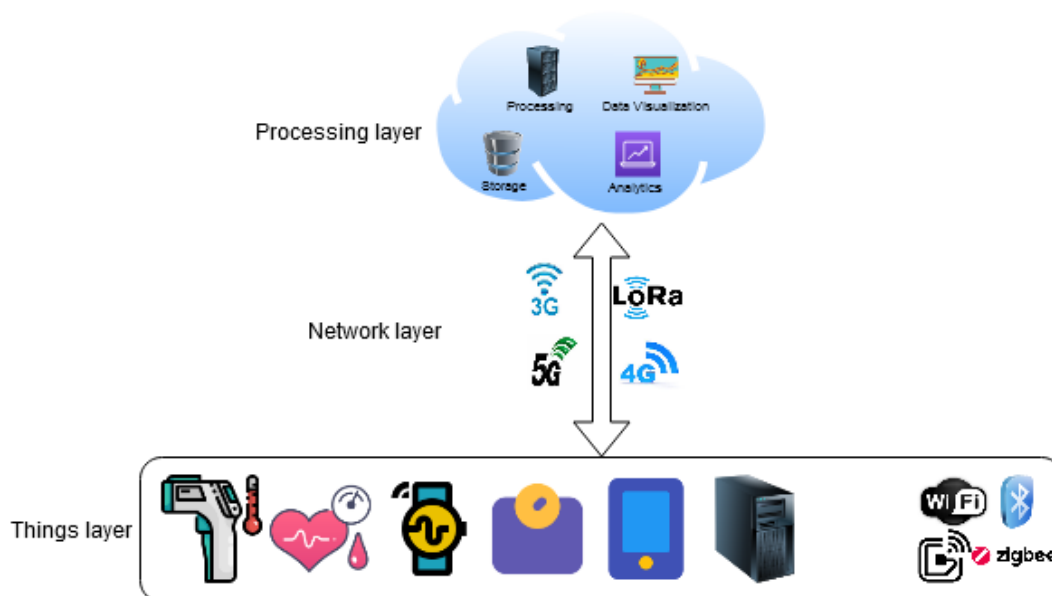


Figura 2.1: Architettura a tre strati

Ci sono architetture alternative che a questi strati ne aggiungono altri due: quello edge e quello dei servizi/applicazioni. Il primo viene inserito tra lo strato delle cose e quello di rete; viene utilizzato per separare i dispositivi edge situati nella rete locale, che si occupano del pre-processamento ed elaborazione locale e che nella rappresentazione precedente erano raggruppati nello strato dei sensori. Lo strato dei servizi/applicazioni

viene inserito dopo il livello di processamento, viene utilizzato per esplicitare la divisione tra i server ed i servizi cloud utilizzati per l'elaborazione e l'analisi dei dati e tutte quelle applicazioni che vengono alimentate da essi.

2.3 Tecnologie innovative

Parlando dello stato dell'arte bisogna considerare anche lo stato delle tecnologie attuali e future che possono impattare sulle piattaforme H-IoT, in particolare in [9] vengono esplorate alcune delle tecnologie innovative, l'impatto che possono avere sulle piattaforme H-IoT e l'ambito nel quale potrebbero essere utilizzate. Il documento tratta di tecniche e tecnologie come machine learning, edge computing, big data, blockchain e software-defined network; per quanto riguarda machine learning fa una panoramica sull'utilizzo attuale, sia a livello di applicazione per la correzione degli errori, l'elaborazione del segnale e la sicurezza, che a livello di rete per l'allocazione delle risorse, la valutazione della qualità della connessione e per la previsione del traffico. Un aspetto fondamentale per i sistemi H-IoT sono i big data e le tecniche specializzate per l'analisi e l'elaborazione di grandi moli di informazioni, dato il flusso continuo di dati generati dai sensori; questi permettono la creazione di modelli matematici sempre più accurati per guadagnare rapidità ed efficienza nella diagnosi. Il modello edge computing ha l'obiettivo di diminuire la latenza, riducendo la distanza tra i sensori ed i nodi di elaborazione e velocizzando la trasmissione, inoltre permette l'interoperabilità tra dispositivi eterogenei creando uno strato aggiuntivo tra questi ed i servizi cloud; le blockchain possono essere utilizzate per eliminare terze parti per il controllo degli accessi e per migliorare l'interoperabilità e la gestione dei dati, in particolare per quanto riguarda la loro privacy e sicurezza, portando vantaggi in termini di velocità, efficienza e soprattutto sicurezza. Infine le software-defined network (SDN) possono essere utilizzate per la gestione della rete, in particolare queste permettono di disaccoppiare il piano di controllo da quello dei dati consentendo quindi la configurazione e la gestione dinamica delle regole di inoltro nella rete; l'utilizzo di SDN rende quindi il sistema più programmabile portando vantaggi in termini di privacy, sicurezza e migliorando la gestione e l'allocazione delle risorse.

Capitolo 3

Definizione ed analisi dei requisiti

Lo scopo è lo studio e la progettazione di un'architettura per la raccolta dati in campo medico/sanitario attraverso questionari e dispositivi eterogenei, aggregandoli sotto un modello comune, al fine di poterli riutilizzare per diverse tipologie di applicazioni, come l'elaborazione, la presentazione grafica o l'analisi per ottenere inferenza dai dati. Inoltre è richiesta la progettazione e lo sviluppo prototipale di una piattaforma che si basi sull'architettura realizzata, per verificarne la fattibilità, calandola nel caso d'uso Covid-19.

3.1 Requisiti

In questa sezione sono presenti i requisiti individuati inizialmente, in particolare possono essere divisi in due parti, nella prima sono presenti quelli riguardanti l'architettura da progettare, e nella seconda quelli del caso d'uso Covid-19 raccolti nell'incontro con un medico.

3.1.1 Architettura di riferimento

La piattaforma deve essere progettata per essere in grado di:

- raccogliere dati relativi allo stato di salute di un paziente, tra i quali devono essere considerati i seguenti parametri:

- saturazione di ossigeno nel sangue
 - frequenza cardiaca
 - temperatura corporea
 - peso corporeo
 - pressione sanguigna
 - funzione respiratoria
 - frequenza respiratoria
- raccogliere dati relativi all'attività fisica, in particolare devono essere supportati i seguenti parametri:
 - numero di step
 - durata attività
 - frequenza cardiaca massima, minima e media
 - calorie bruciate
 - distanza percorsa
- raccogliere dati ambientali relativi alla qualità dell'aria, in particolare devono essere supportati i seguenti parametri:
 - temperatura
 - umidità
 - pressione
 - livello di composti organici volatili
 - livello di anidride carbonica
 - livello di monossido di carbonio
 - livello di particolato PM10
 - livello di particolato PM2,5
 - livello di diossido di azoto

- livello di ozono
- supportare dispositivi commerciali eterogenei prodotti da differenti venditori
- supportare la raccolta di informazioni tramite l'utilizzo di questionari
- supportare diverse tipologie di applicazioni sui dati raccolti, alcuni esempi sono i seguenti: elaborazione, presentazione, analitica e generazione di allarmi

3.1.2 Piattaforma caso d'uso Covid-19

Nella seconda parte è richiesta la progettazione di una piattaforma per il monitoraggio di pazienti Covid-19 in modo da validare l'architettura calandola in un caso reale di esempio applicativo nel quale può essere sfruttata. La piattaforma dovrà essere progettata per supportare il monitoraggio su richiesta dei parametri vitali di pazienti che hanno o potrebbero aver contratto il Covid-19; dovrà quindi avere due interfacce, una utilizzata dal paziente per la connessione coi propri dispositivi personali e per essere sottoposto a questionari e test di valutazione del proprio stato di salute, e l'altra utilizzata dal medico per visualizzare i dati forniti dal paziente.

L'interfaccia utilizzata dal paziente dovrà essere pensata come applicazione che possa funzionare sui dispositivi mobili con sistema operativo Android, tramite questa il paziente potrà:

- accedere al proprio account
- modificare i propri dati anagrafici o le proprie credenziali
- associare nuovi dispositivi o rimuovere quelli attualmente connessi
- compilare o consultare questionari
- effettuare o consultare test

L'interfaccia utilizzata dal medico dovrà essere progettata per funzionare su dispositivi eterogenei, come smartphone, computer, tablet, e tramite questa il medico potrà:

- accedere al proprio account

- visualizzare una panoramica dei pazienti mostrando un sommario dei dati più di rilievo per Covid-19, tra questi:
 - saturazione dell'ossigeno nel sangue
 - frequenza cardiaca
 - temperatura corporea
 - pressioni sistolica e diastolica
 - saturazione minima dell'ultima attività fisica
 - un indicatore sintetico che varia in base ai parametri dei pazienti per segnalare il livello di rischio dell'individuo
- selezionare un paziente per vedere un sommario dettagliato di tutti i suoi dati raccolti, quindi parametri vitali, storico delle attività fisiche e questionari e test effettuati

I parametri che dovranno essere supportati sono i seguenti:

- saturazione dell'ossigeno nel sangue
- frequenza cardiaca
- temperatura corporea
- pressioni sistolica e diastolica
- peso corporeo
- frequenza respiratoria
- passi giornalieri
- funzione respiratoria

Le informazioni riguardanti le attività fisiche effettuate dai pazienti che dovranno essere supportate sono:

- tempo di inizio allenamento

- passi
- calorie bruciate
- distanza percorsa
- altitudine
- frequenza cardiaca media, massima e minima
- durata allenamento
- saturazione iniziale e minima
- velocità di desaturazione

Il questionario che verrà sottoposto ai pazienti sarà utilizzato dai medici per ottenere una valutazione da parte del paziente per quanto riguarda le difficoltà respiratorie e l'intensità della tosse. I test che dovranno essere supportati sono il walking test e lo speech test; il primo consiste nel determinare la variazione di saturazione di ossigeno nel sangue prima e dopo aver effettuato una camminata di 5/6 minuti, mentre il secondo test viene utilizzato per valutare la funzione polmonare di un individuo e consiste nel leggere più parole possibili o contare fino a terminare il fiato dopo aver effettuato un respiro profondo.

3.2 Analisi dei requisiti

In questa sezione verranno analizzati i requisiti in modo tale da individuare in modo chiaro cosa dovrà essere progettato. Da una prima analisi dei requisiti è possibile individuare alcune caratteristiche del sistema, innanzitutto dovrà essere distribuito in quanto è richiesta la raccolta dati da dispositivi che sono già di per sé distribuiti, inoltre queste informazioni potranno essere accumulate tramite smartphone o altri componenti che difficilmente saranno in grado di fornire le applicazioni proposte, in quanto limitati in potere computazionale e per la natura distribuita di alcune tipologie di applicazioni. In base a questo è possibile dividere il sistema in due parti: la prima situata in prossimità del paziente, che si dovrà occupare della connessione ai dispositivi locali e della

loro raccolta dati, e la seconda che si occuperà della memorizzazione dei dati e della loro trasformazione, in modo tale che possano essere utilizzati dalle eventuali applicazioni.

3.2.1 Raccolta dati

Per quanto riguarda la raccolta dati, il sistema dovrà essere altamente configurabile nella gestione dei dispositivi rendendo possibile l'aggiunta e la rimozione di device con il minimo sforzo; inoltre, uno dei requisiti consiste nel supportare il maggior numero possibile di dispositivi commerciali e vendor, ovviamente non è pensabile integrare tutti i dispositivi presenti sul mercato considerandoli uno per volta, questo perchè il loro numero è molto elevato ed in continua espansione. Per questo motivo è importante identificare le possibili metodologie e tecnologie con le quali tali dispositivi sono in grado di condividere i dati raccolti, quelle individuate sono le seguenti:

- utilizzando protocolli a corto raggio come ad esempio Bluetooth, Zigbee, Z-wave e Wifi. In questo caso possiamo raccogliere i dati tramite uno smartphone, sfruttando gli SDK forniti dai produttori dei dispositivi, o tramite un componente hardware che possiamo chiamare componente edge;
- sfruttando le API dei servizi cloud offerti dai produttori. In questo caso i dati saranno inviati dai dispositivi, in modo diretto se connessi ad una rete wifi o passando attraverso l'applicazione del vendor installata sullo smartphone del paziente, ad una piattaforma cloud del fornitore e saranno resi disponibili effettuando delle richieste alle API dei servizi offerti. Alcuni esempi di aziende che utilizzano questa tipologia sono Withings e Garmin;
- sfruttando aggregatori come Google Fit e Validic che già di per sé integrano una grande fetta di dispositivi delle marche più conosciute sul mercato.

Inoltre devono essere supportati anche i questionari che potrebbero essere compilati dai pazienti attraverso i propri dispositivi da casa; sono possibili due strade: la realizzazione di un'applicazione per smartphone o di un servizio nel quale i pazienti potranno compilare i questionari ed inserirli nel sistema. Dato che non sono presenti vincoli e che dovrà comunque essere prevista un'applicazione per smartphone per la connessione con

i dispositivi locali utilizzati per la raccolta dei parametri, la soluzione più consona è la prima, quindi l'inserimento di una funzionalità apposita nell'applicazione mobile.

Per quanto riguarda le applicazioni da supportare dovrà essere realizzato un modello standard e tutti i dati raccolti dovranno essere trasformati in modo tale da essere conformi con esso; in questo modo sarà possibile fornire i dati alle applicazioni indipendentemente dai dispositivi con i quali sono stati raccolti. Una delle applicazioni presenti nei requisiti è la generazione di allarmi, questa operazione potrebbe essere più complessa delle altre in quanto si presume sia tempo-dipendente; per questo motivo nella progettazione dell'architettura dovrà essere preso in considerazione questo aspetto, avvicinando il più possibile tale processo alla fase di raccolta dei dati.

3.2.2 Sicurezza e privacy dei dati

Un aspetto di particolare rilievo, data la delicatezza dei dati trattati, è il tema della sicurezza e della privacy, che non può essere quindi tralasciato in fase di design; per prima cosa la piattaforma dovrà essere pensata per essere GDPR compliant, questo significa che dovrà rispettare le caratteristiche individuate in esso, come ad esempio il diritto di rettifica ed eliminazione dei dati, il diritto di accesso ad essi e di portabilità ed il diritto di notifica in caso di compromissione. Gli aspetti da prendere in considerazione quando si tratta di sicurezza sono diversi, tra questi vi sono alcune proprietà che il sistema dovrà garantire come la confidenzialità, l'integrità e la disponibilità del dato; per ottenere disponibilità potrebbe essere utile permettere la replicazione sia dei database che di tutti i servizi facenti parte della piattaforma, mentre per garantire confidenzialità ed integrità la questione è più complessa. Un altro degli aspetti da considerare è il meccanismo di autenticazione al sistema, una delle tecniche per aumentare la sicurezza di questo processo è la multi-factor authentication che consiste nell'utilizzo di più fattori per verificare con certezza l'identità di un utente, ad esempio potrebbe essere richiesta una password ed un codice di verifica che verrà inviato ad un suo dispositivo personale.

I dati relativi allo stato di salute di un individuo devono essere de-identificati, cioè non deve essere possibile ricondurre tali dati presenti all'interno della piattaforma al paziente dal quale sono stati raccolti se non si è un soggetto autorizzato, come ad esempio il paziente stesso o il suo medico curante; per far ciò i dati che possono ricondurre al-

l'identità del paziente, come ad esempio l'anagrafica, dovranno essere separati dal resto dei dati e potranno essere disponibili solamente da chi ha i permessi per farlo e non da tutta l'organizzazione. Questi dati saranno quindi criptati e la chiave sarà fornita solo agli individui con i permessi necessari; per collegare i record relativi allo stato di salute al particolare paziente dal quale sono stati raccolti, verrà utilizzato un identificativo che non fornisca alcuna indicazione sull'identità dell'individuo ma che permetta di individuare la sua anagrafica, che essendo criptata non potrà essere visionata da personale non autorizzato. Nessun dato dovrà mai essere trasmesso o memorizzato in chiaro, questo non vale solo per i dati relativi all'anagrafica dei pazienti ma per tutti i dati presenti nella piattaforma; questi dovranno essere crittografati e la chiave dovrà essere condivisa solamente tra gli individui che hanno un ruolo nel sistema, garantendo così la confidenzialità del dato in quanto nessuno esterno ad essa potrà accedere alle informazioni presenti.

3.2.3 Caso d'uso Covid-19

Per quanto riguarda la progettazione della piattaforma per il caso d'uso Covid-19 è possibile identificare da subito tre parti principali: le prime due sono le stesse identificate nella sezione precedente, che si occupano quindi della raccolta dati locale e della trasformazione per renderli conformi al modello standard adottato dalla piattaforma, mentre la terza parte è quella utilizzata dai medici e che gli permette di visualizzare e monitorare i dati dei pazienti. Nell'analisi delle prime due parti possiamo considerare valide le assunzioni viste precedentemente; per quanto riguarda il componente dedicato alla raccolta dati locale dai dispositivi, i requisiti richiesti sono più specifici, viene infatti richiesta la progettazione di un'interfaccia compatibile con Android e, dato che non sono presenti altri vincoli, è possibile realizzare questa parte come applicazione nativa per tale sistema operativo. Tramite questa applicazione il paziente dovrà essere in grado di collegare nuovi dispositivi o rimuovere quelli attualmente connessi, effettuare e consultare walking test e speech test e compilare e consultare il questionario per l'autovalutazione delle difficoltà respiratorie; il modello di questionario che sarà utilizzato per valutare le difficoltà respiratorie dei pazienti è il test di valutazione della broncopneumopatia cronica ostruttiva (BPCO), che consiste in un insieme di domande che prevedono in risposta un valore nell'intervallo tra 0 e 5, inoltre tale questionario potrebbe essere esteso quindi

deve essere previsto il supporto a domande che accettano risposte del tipo si e no.

Il walking test richiede la misurazione della variazione di saturazione di ossigeno nel sangue a seguito di una camminata, potrebbe quindi essere effettuato richiedendo la misurazione di tale parametro prima e dopo aver effettuato una camminata di tempo prefissato, inoltre può essere utile permettere al paziente di terminare il test prima dello scadere del tempo qualora non sia in grado di portarlo a termine; in questo caso dovrà essere richiesta comunque una nuova misurazione della saturazione e sarà rilevante anche l'informazione del tempo totale di camminata che il paziente ha sostenuto.

Lo speech test potrebbe essere effettuato in due modi: fornendo un testo standard e determinando quante parole il paziente è in grado di leggere, oppure facendolo contare fino a terminare il fiato; non essendoci vincoli su quale soluzione attuare conviene scegliere la seconda dato che risulta quella più semplice. Per effettuare questo test sarà quindi necessario che il paziente faccia un respiro profondo ed inizi a contare fino a quando non terminerà il fiato, a questo punto dovrà fornire il numero che ha raggiunto come risultato del test.

L'interfaccia del medico dovrà essere progettata per funzionare su dispositivi eterogenei, per questo motivo potrebbe essere utile pensare alla realizzazione di una applicazione web, così potrà essere utilizzata tramite il computer personale, una workstation o tramite dispositivi mobili quali smartphone e tablet. Per quanto riguarda i dati che dovranno essere raccolti e visualizzati in tale interfaccia, nelle tabelle 3.1 e 3.2 sono mostrati rispettivamente i parametri di salute e le informazioni sugli allenamenti; per ognuno viene indicata l'unità di misura ed il livello di priorità del dato per il caso d'uso Covid-19, ottenuti da un secondo incontro con un medico.

Parametro	Unità di misura	Priorità
Saturazione ossigeno nel sangue (SpO2)	percentuale (%)	alta
Frequenza cardiaca	battiti per minuto (bpm)	alta
Temperatura corporea	gradi centigradi (°C)	alta
Pressione sanguigna sistolica	millimetro di mercurio (mmHg)	alta
Pressione sanguigna diastolica	millimetro di mercurio (mmHg)	alta
Frequenza respiratoria	atti respiratori al minuto	alta
Passi giornalieri	passi	alta
Funzione respiratoria (FEV1)	litri (L)	media
Funzione respiratoria (PEF)	litri al secondo (L/s)	media
Altezza	centimetri (cm)	bassa
Peso corporeo	chilogrammi (Kg)	bassa
Percentuale di grasso corporeo	percentuale (%)	bassa
Indice di massa corporea (BMI)	-	bassa

Tabella 3.1: parametri caso d'uso Covid-19

Parametro	Unità di misura	Priorità
Passi	-	alta
Distanza percorsa	metri (m)	alta
Durata attività	secondi (s)	alta
Saturazione iniziale	percentuale (%)	alta
Saturazione minima	percentuale (%)	alta
Velocità di desaturazione	secondi (s)	alta
Calorie bruciate	Equivalente Metabolico dell'Attività (MET)	media
Altitudine	piani	media
Battito cardiaco medio	battiti per minuto (bpm)	media
Battito cardiaco massimo	battiti per minuto (bpm)	media
Battito cardiaco minimo	battiti per minuto (bpm)	media
Tempo di inizio	-	media

Tabella 3.2: informazioni sulle attività fisiche caso d'uso Covid-19

Inoltre dovrà essere presente un indicatore sintetico che permetterà al medico di individuare immediatamente il grado di pericolo delle condizioni dei pazienti, questo viene rappresentato attraverso l'uso di colori che ne rappresentano il livello di rischio. Tale indicatore può assumere i seguenti valori:

- verde, che rappresenta una situazione senza rischio
- giallo, che rappresenta una situazione di basso rischio
- arancione, che rappresenta una situazione di medio rischio
- rosso, che rappresenta una situazione di alto rischio

Il colore di tale indicatore è definito, seguendo il National early warning score (NEWS), in relazione ai punteggi assegnati ai seguenti parametri: saturazione di ossigeno nel sangue, temperatura corporea, pressione sistolica e frequenza cardiaca; ad ogni parametro viene assegnato un punteggio da 0 a 3 come indicato in tabella 3.3 e questi vengono sommati

per ottenere lo score totale che determina il colore dell'indicatore nel seguente modo: verde con 0 punti, giallo fino a 4 punti, arancione fino a 6 punti e rosso da 7 punti in poi.

	3	2	1	0	1	2	3
Saturazione ossigeno	≤ 91	92-93	94-95	≥ 96			
Temperatura corporea	$\leq 35,0$		35,1-36,0	36,1-38,0	38,1-39,0	$\geq 39,1$	
Pressione sistolica	≤ 90	91-100	101-110	111-219			≥ 220
Frequenza cardiaca	≤ 40		41-50	51-90	91-110	111-130	≥ 131

Tabella 3.3: punteggi per il calcolo dell'indicatore sintetico

3.3 Scenari caso d'uso Covid-19

In questa sezione verranno individuati ed esplorati gli scenari di utilizzo della piattaforma in modo tale da rappresentare come gli utenti utilizzeranno il sistema proposto, quindi quali sono le azioni e le operazioni che essi potranno compiere. Tali scenari sono rappresentati graficamente nella figura 3.1 e saranno esplorati, fornendone una breve descrizione ed eventuali precondizioni o postcondizioni, nelle tabelle che seguono.

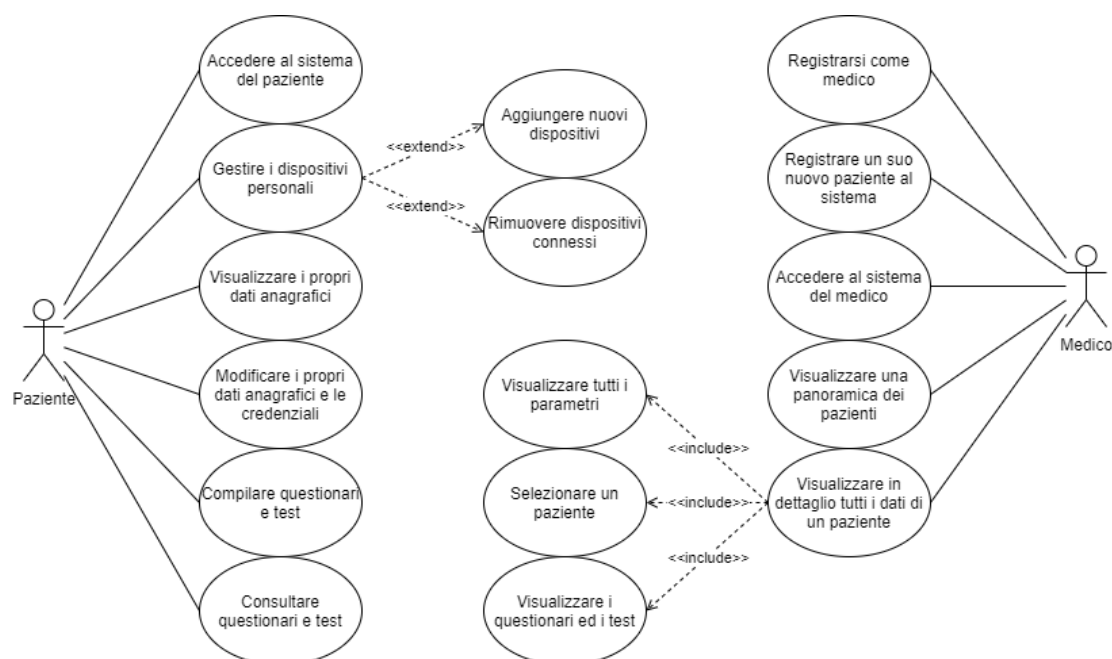


Figura 3.1: Scenari caso d'uso Covid-19

Accedere al sistema del paziente

Descrizione	L'utente inserisce le proprie credenziali nell'interfaccia dedicata al paziente.
Precondizione	L'utente deve essere registrato nel sistema come paziente.
Postcondizione	<ul style="list-style-type: none"> ➤ Credenziali corrette: il sistema caricherà tutte le informazioni del paziente e passerà nello stato operativo. ➤ Credenziali errate: il sistema comunica all'utente l'immissione errata delle credenziali e lo riporta all'interfaccia di accesso.

Gestire i dispositivi personali

Descrizione	<ul style="list-style-type: none"> ➤ Il paziente decide di aggiungere un nuovo dispositivo. ➤ Il paziente decide di rimuovere un dispositivo.
Precondizione	<ul style="list-style-type: none"> ➤ L'utente deve aver effettuato l'accesso al sistema come paziente. ➤ (caso rimozione dispositivo) Il dispositivo deve essere stato precedentemente connesso.
Postcondizione	<ul style="list-style-type: none"> ➤ (caso aggiunta dispositivo) Il dispositivo verrà aggiunto dalla lista dei dispositivi personali del paziente. ➤ (caso rimozione dispositivo) Il dispositivo verrà rimosso dalla lista dei dispositivi personali del paziente.

Visualizzare i propri dati anagrafici

Descrizione	Il paziente apre il proprio profilo per consultare i suoi dati anagrafici.
Precondizione	L'utente deve aver effettuato l'accesso al sistema come paziente.

Modificare i propri dati anagrafici e le credenziali

Descrizione	<ol style="list-style-type: none"> 1. Il paziente modifica uno o più campi tra i suoi dati anagrafici e le credenziali. 2. Conferma la modifica apportata.
Precondizione	<ul style="list-style-type: none"> ➤ L'utente deve aver effettuato l'accesso al sistema come paziente. ➤ Il paziente deve aver aperto il proprio profilo dove saranno presenti i dati anagrafici e le credenziali.
Postcondizione	<ul style="list-style-type: none"> ➤ Vengono aggiornati i dati all'interno della piattaforma. ➤ L'interfaccia grafica viene aggiornata coi dati modificati.

Compilare questionari e test

Descrizione	<ul style="list-style-type: none"> ➤ Il paziente decide di aggiungere un questionario. <ol style="list-style-type: none"> 1. Il paziente compila il questionario rispondendo alle domande presenti. 2. Il paziente conferma la terminazione del questionario inviandolo al sistema. ➤ Il paziente decide di effettuare un walking test. <ol style="list-style-type: none"> 1. Il paziente si misura la saturazione dell'ossigeno nel sangue. 2. Il paziente effettua una camminata. 3. Il paziente si misura nuovamente la saturazione dell'ossigeno nel sangue. 4. Il paziente conferma la terminazione del walking test inviandolo al sistema. ➤ Il paziente decide di effettuare uno speech test. <ol style="list-style-type: none"> 1. Il paziente effettua un respiro profondo e conta fino a terminare l'aria. 2. Il paziente fornisce il valore al quale è arrivato a contare. 3. Il paziente conferma la terminazione dello speech test inviandolo al sistema.
Precondizione	<ul style="list-style-type: none"> ➤ L'utente deve aver effettuato l'accesso al sistema come paziente. ➤ (per il walking test) Il paziente deve essere in grado di misurare il proprio livello di saturazione di ossigeno nel sangue.
Postcondizione	<ul style="list-style-type: none"> ➤ Viene aggiunto il questionario o il test all'interno del sistema.

Consultare questionari e test

Descrizione	Il paziente può scegliere di consultare un questionario, un walking test o uno speech test.
Precondizione	<ul style="list-style-type: none"> ➤ L'utente deve aver effettuato l'accesso al sistema come paziente. ➤ Il paziente deve aver effettuato almeno un questionario, un walking test o uno speech test.

Registrarsi come medico

Descrizione	L'utente vuole registrarsi come medico nel sistema.
Precondizione	Deve essere verificata l'identità del medico.
Postcondizione	Se l'azione è consentita, viene aggiunto un nuovo account di medico all'interno del sistema.

Registrare un suo nuovo paziente al sistema

Descrizione	<ol style="list-style-type: none"> 1. Il medico può aggiungere un nuovo paziente al sistema fornendo l'email del paziente. 2. Il sistema genera una password temporanea per il paziente che dovrà essere cambiata al primo accesso.
Precondizione	<ul style="list-style-type: none"> ➤ L'utente deve aver effettuato l'accesso al sistema come medico. ➤ Il paziente deve richiedere di poter essere monitorato al proprio medico curante.
Postcondizione	L'account del paziente viene aggiunto al sistema.

Accedere al sistema del medico

Descrizione	L'utente inserisce le proprie credenziali nell'interfaccia dedicata al medico.
Precondizione	L'utente deve essere registrato nel sistema come medico.
Postcondizione	<ul style="list-style-type: none"> ➤ Credenziali corrette: il sistema caricherà tutte le informazioni del medico e dei pazienti registrati presso lui, e passerà nello stato operativo. ➤ Credenziali errate: il sistema comunica all'utente l'immissione errata delle credenziali e lo riporta all'interfaccia di accesso.

Visualizzare una panoramica dei pazienti

Descrizione	Il medico accede alla schermata panoramica dei suoi pazienti registrati visualizzando un sommario parziale dei parametri raccolti.
Precondizione	L'utente deve aver effettuato l'accesso al sistema come medico.

Visualizzare in dettaglio tutti i dati di un paziente

Descrizione	<ol style="list-style-type: none">1. Il medico accede alla panoramica generale dei suoi pazienti registrati.2. Il medico seleziona il paziente al quale è interessato.3. Il medico seleziona i dati del paziente che vuole visualizzare nel dettaglio, potrà quindi visualizzare i questionari, i test e tutti i parametri raccolti.
Precondizione	L'utente deve aver effettuato l'accesso al sistema come medico.

Capitolo 4

Disegno architettonico

In questo capitolo verrà definita l'architettura della piattaforma spiegando come si è pensato di progettare, mostrandone il modello dei componenti e motivando le scelte architettoniche che sono state effettuate. In figura 4.1 è mostrata una panoramica dei componenti ad un alto livello di astrazione, evidenziando le parti che dovranno essere prese in considerazione e che verranno discusse in questo capitolo; in essa possiamo individuare la fase di raccolta dei dati, la loro trasformazione per renderli conformi al modello comune della piattaforma, la loro memorizzazione e le varie tipologie di applicazioni. Tra queste si può notare la generazione degli allarmi, che è collocata in una posizione differente rispetto alle altre perché di natura tempo-dipendente, per questo motivo è stata posizionata prima della fase di memorizzazione dei dati, per ridurre i tempi di latenza, e dopo la trasformazione dei dati, in quanto si vuole utilizzare i dati nel formato conforme al modello della piattaforma e non in quello dei dispositivi con i quali sono stati raccolti.

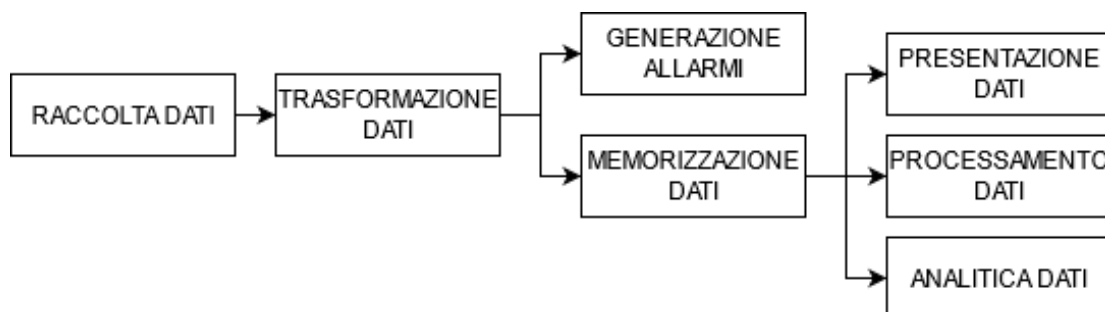


Figura 4.1: Panoramica dei componenti della piattaforma

4.1 Raccolta dei dati

In questa sezione approfondiamo le possibili metodologie utilizzabili per la raccolta dei dati relativi allo stato del paziente, in particolare dovranno essere considerati sia i parametri raccolti attraverso i dispositivi, sia i questionari ed i test che potranno essere effettuati.

Partendo dalla raccolta dati dai dispositivi e tralasciando momentaneamente i questionari ed i test, prima di tutto deve essere fatta una valutazione delle metodologie con le quali questi possono comunicare; in figura 4.2 viene mostrata una panoramica generale di tali tipologie di comunicazione. Può essere utile iniziare facendo chiarezza sui concetti contenuti nella legenda:

- per connessione a corta distanza si intendono quelle tecnologie e protocolli utilizzati per connettere localmente i dispositivi, tra questi troviamo Bluetooth, Zigbee, Z-wave e BLE;
- Wi-fi ed ethernet sono connessioni che consentono la comunicazione tra i componenti ed il router;
- per comunicazione a lunga distanza si intendono tecnologie e protocolli che permettono il collegamento dei dispositivi ad Internet, tra questi troviamo GSM, fibra, 3G, 4G, 5G.

Nell'immagine è possibile vedere 4 differenti dispositivi che identificano ognuno una differente tipologia. Il dispositivo 1 identifica la tipologia di device in grado di connettersi direttamente ad Internet per condividere i dati raccolti con la piattaforma cloud del proprio vendor; questi dispositivi sono connessi localmente con il router tramite una connessione wi-fi o ethernet. I dispositivi 2 e 4 utilizzano una connessione a corto raggio per comunicare localmente, in questo caso è necessario un componente che avrà il ruolo di mediatore di protocollo, cioè riceverà i dati dai device tramite i protocolli Zigbee, Z-wave o BLE, e li inoltrerà alla piattaforma cloud sfruttando la connessione con il router, o se disponibile una connessione a lungo raggio. La differenza tra il flusso dati dei dispositivi 2 e 4 sta nel componente utilizzato: nel caso 2 viene utilizzato un gateway che ha una potenza computazionale limitata ed ha il solo ruolo di mediatore di protocollo, mentre

nel 4 viene utilizzato un componente edge, questo è un dispositivo pensato per avere più potenza computazionale e differenti possibilità di connessione. Il componente edge, oltre a ricoprire il ruolo di mediatore di protocollo, potrebbe quindi effettuare operazioni di pre-processamento, memorizzazione locale o trasformazione dei dati, inoltre potrebbe essere provvisto di tecnologie di comunicazione a lungo raggio che gli consentirebbero di lavorare anche in assenza del router. Infine rimane il dispositivo 3, in questo caso la connessione a corto raggio è limitata alle tecnologie supportate dagli smartphone, quindi il Bluetooth; tramite un'applicazione sullo smartphone sarà possibile raccogliere i dati e trasmetterli alla piattaforma cloud del vendor del dispositivo.

Una volta viste le diverse modalità di comunicazione dei dispositivi, e considerando che dovranno essere raccolte informazioni anche tramite questionari e test, è possibile individuare le metodologie con le quali la piattaforma da realizzare potrà raccogliere le informazioni di interesse, queste sono: tramite un'applicazione presente sullo smartphone del paziente che gli permetterà di connettere dispositivi ed effettuare questionari e test, interrogando le piattaforme cloud dei produttori o di aggregatori come Google Fit o Validic, oppure attraverso un componente edge che dovrà essere installato presso l'abitazione del paziente.

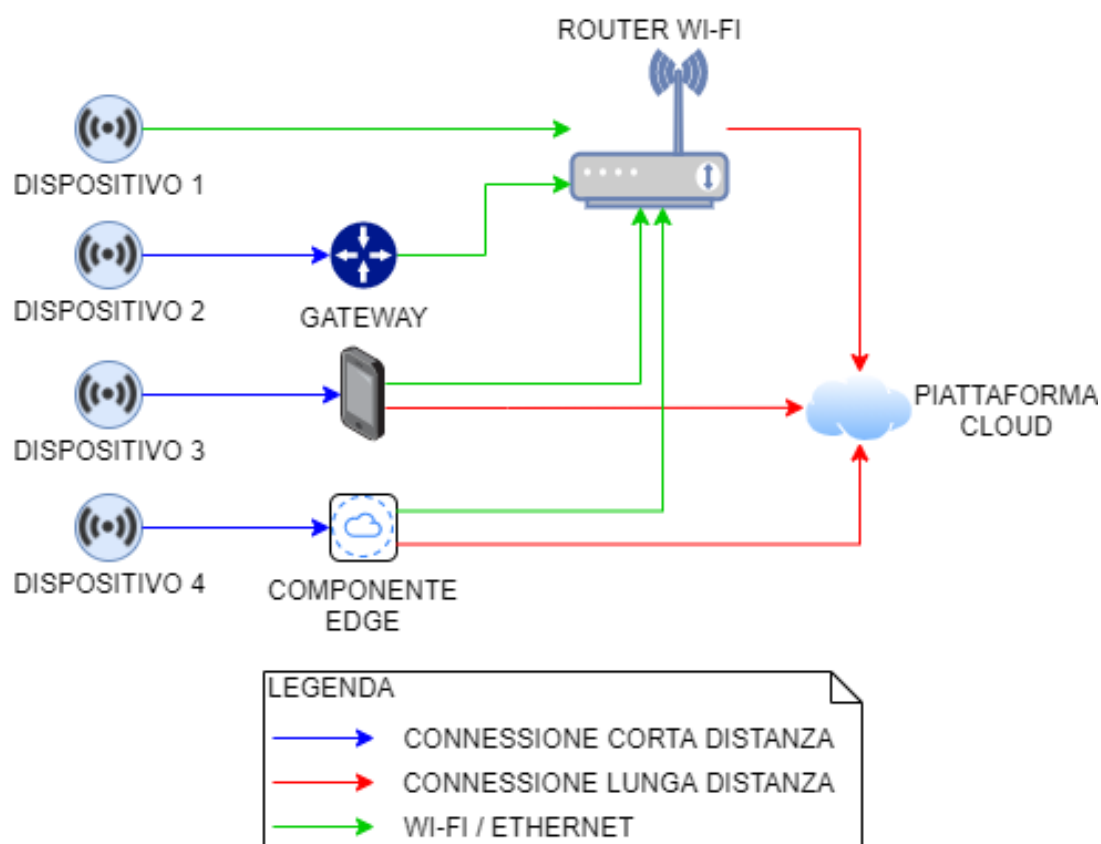


Figura 4.2: Panoramica metodologie di comunicazione dei dispositivi

4.1.1 Smartphone

Per consentire la raccolta di informazioni tramite lo smartphone sarà necessario progettare e realizzare un'applicazione nativa Android; queste caratteristiche sono state scelte in quanto nei requisiti viene esplicitamente richiesta la compatibilità con il sistema operativo Android, inoltre dovrà essere nativa per poter sfruttare al massimo tutte le funzionalità offerte dal sistema operativo e per utilizzare gli SDK forniti da alcuni produttori di dispositivi, che permettono la connessione del device e la condivisione delle misurazioni. L'utilizzo dell'SDK non è l'unica soluzione che può essere fornita dai vendor dei dispositivi per ottenere i dati raccolti; come visto precedentemente alcuni produttori hanno delle piattaforme che utilizzano per collezionare i dati dai diversi modelli di dispositivi prodotti e tramite queste forniscono servizi che, attraverso richieste alle loro

API, permettono di ottenere le informazioni raccolte. Per poter usufruire di questi servizi offerti dalla piattaforma sarà necessario che il paziente fornisca l'autorizzazione per l'accesso ai propri dati; solitamente tale operazione viene effettuata attraverso l'utilizzo del protocollo OAuth 2.0, che consiste in un processo di autenticazione dell'utente al fine di fornire i permessi necessari per l'accesso ai suoi dati. Attraverso l'applicazione sarà quindi possibile effettuare tale operazione per ottenere i dati direttamente dalle piattaforme cloud, connettere i propri dispositivi personali attraverso l'utilizzo delle librerie messe a disposizione dai produttori, compilare questionari ed effettuare walking e speech test.

Compilare i questionari attraverso l'applicazione è un'operazione alquanto semplice, basta realizzare una schermata contenente le domande da fare al paziente ed al momento della terminazione dovranno essere raccolte le domande effettuate e le risposte per essere inviate alla piattaforma; dato che si vuole supportare una possibile estensione delle domande e questionari con domande differenti, il risultato del questionario deve codificare sia le domande che le risposte in modo da poter ricondurre ogni risposta alla domanda alla quale era riferita.

Per la realizzazione del walking test è possibile individuare 3 possibili fasi: la prima consiste nella misurazione iniziale della saturazione dell'ossigeno nel sangue e termina con l'inizio della camminata, la seconda comprende la camminata fino al suo termine, e la terza consiste in una nuova misurazione del SpO₂. Il metodo più semplice per la transizione tra le fasi è quello di far comunicare all'utente gli eventi di inizio e fine camminata; inoltre devono essere previste due possibili modalità di inserimento della saturazione dell'ossigeno nel sangue, tramite un dispositivo associato o tramite l'inserimento manuale, qualora il paziente non disponesse di un dispositivo compatibile. Una volta terminato il test dovranno essere inviate alla piattaforma le misurazioni di saturazione dell'ossigeno nel sangue ed il tempo di camminata sostenuto.

Lo speech test può essere realizzato in modo automatizzato o manuale; per automatizzato si intende una modalità nella quale sarà l'applicazione, attraverso tecniche di Natural language processing (NLP), ad individuare il valore al quale l'utente è arrivato a contare, mentre per manuale si intende una metodologia nella quale sarà l'utente ad inserire tale informazione. Dato che non sono posti vincoli sulla tipologia da adottare

si sceglie quella più semplice, quindi l'utente, dopo aver effettuato un profondo respiro, conterà fino a terminare il fiato ed inserirà il numero al quale è arrivato in un campo, questa sarà l'informazione rilevante che dovrà essere inviata alla piattaforma.

4.1.2 Piattaforme cloud

Per raccogliere informazioni dalle piattaforme cloud dei produttori dei dispositivi, come ad esempio Garmin e Withings, o di aggregatori come Google Fit e Validic, è necessaria la progettazione di uno o più componenti in grado di effettuare l'integrazione di tipo cloud-to-cloud. Questi componenti dovranno essere in grado di raccogliere i dati dalle piattaforme cloud per integrarli in quella realizzata; ci sono diverse metodologie con le quali è possibile collezionare dati dalle piattaforme, la più utilizzata è quella che prevede di effettuare richieste tramite RESTful API messe a disposizione dai servizi o attraverso l'uso di streaming API, come ad esempio quelle che utilizzano la tecnologia Server-sent events (SSE).

Per questa tipologia di raccolta dati occorre autenticarsi ed ottenere i permessi dagli utenti, il protocollo più utilizzato a tale scopo è OAuth 2.0, che viene utilizzato da piattaforme cloud come quelle di Withings, Garmin e Google Fit; la prima fase per utilizzare tale protocollo è quella di autenticazione dell'utente per fornire i permessi di accesso ai propri dati, a questa richiesta viene restituito un token che dovrà essere memorizzato perché indispensabile per accedere alle informazioni del paziente. In figura 4.3 è possibile vedere tale processo illustrato attraverso un diagramma di sequenza.

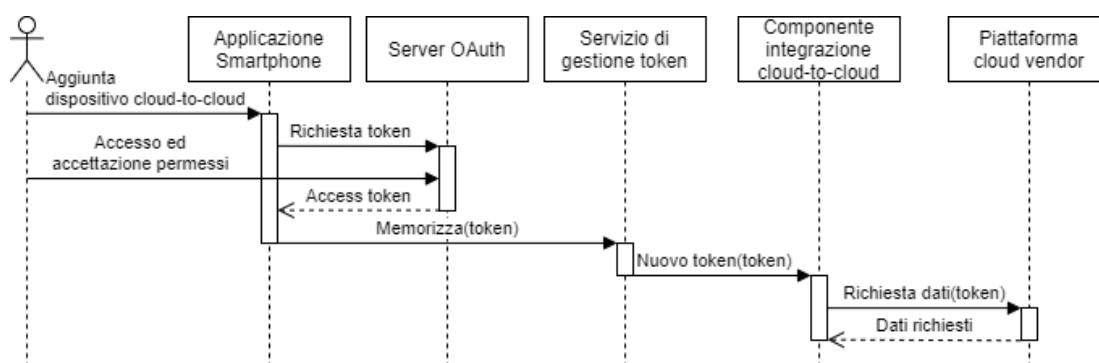


Figura 4.3: Processo integrazione cloud-to-cloud tramite OAuth 2.0

Una piattaforma che si è voluta prendere in considerazione è quella realizzata dall'azienda Americana Validic, questa funge da aggregatore di dati integrando una parte dei dispositivi disponibili in commercio e fornendoli a parti terze utilizzando un formato standard; in particolare supporta la raccolta dati tramite un'applicazione sullo smartphone, fornendo un loro SDK per connettere i dispositivi compatibili e comunicare con i loro servizi cloud, e tramite integrazioni cloud-to-cloud. Una funzionalità molto interessante della piattaforma è VitalSnap che permette di integrare dispositivi che non possiedono alcuna connettività, questa la possiamo trovare nell'SDK fornito per la realizzazione di applicazioni mobile; VitalSnap utilizza la fotocamera per leggere la misurazione effettuata dal display dei dispositivi supportati.

Sfruttare tale piattaforma ha aspetti positivi e negativi; tra quelli positivi è possibile intuire subito che molti dispositivi potrebbero essere integrati tramite questa, senza il bisogno di effettuare ulteriore lavoro, inoltre la funzionalità VitalSnap permetterebbe di integrare device senza connessione che altrimenti non sarebbero supportati da questa piattaforma. Gli aspetti negativi sono relativi alla perdita di controllo sui dati, sfruttando tale piattaforma infatti ci si deve adeguare alle loro tempistiche di raccolta ed al loro modello dati, che potrebbe essere una limitazione qualora servissero misurazioni più frequenti o dovessimo ottenere un parametro non supportato da tale piattaforma.

4.1.3 Componente edge

Tramite questa tipologia di raccolta dati si vogliono integrare tutti i dispositivi che comunicano tramite connessioni a corto raggio e protocolli come Zigbee o Z-wave che non sono compatibili con gli smartphone; inoltre potrebbero essere utilizzati anche come alternativa ad essi per raccogliere dati dai dispositivi che utilizzano la connettività Bluetooth, situati all'interno dell'abitazione e che non vengono mai utilizzati al di fuori di essa, come ad esempio le bilance. Quando si parla di questi componenti solitamente non ci si limita alla raccolta dati, ma vengono incluse un insieme di funzionalità come la memorizzazione, il preprocessamento e l'elaborazione locale, la generazione di allarmi, l'esecuzione di azioni o il controllo di attuatori; essendo collocati vicino alla raccolta dati sono ottimi candidati per le situazioni nelle quali è necessaria una risposta tempestiva ed autonoma sulla base delle informazioni ricevute. Inoltre, dato che verrebbero collocati

all'interno dell'abitazione dei pazienti, potrebbe essere fondamentale gestire tale dispositivo da remoto per modificarne il comportamento, le applicazioni che vengono eseguite, aggiungere o togliere funzionalità.

In figura 4.4 viene mostrata una possibile visione del modello del componente edge distinguendo alcune delle sue funzionalità principali, partendo da sinistra sono rappresentati due dispositivi che potrebbero essere integrati direttamente, o passando attraverso un gateway qualora venisse utilizzato un protocollo non supportato da tale componente; per i protocolli supportati ci sarà una parte che avrà lo scopo di mediazione di protocollo, cioè una funzionalità che dovrà permettere la raccolta dati attraverso questi per renderli disponibili per le fasi successive. In seguito vi è una fase di gestione dei dati che mostra alcune delle eventuali operazioni applicabili su di essi, come la memorizzazione locale, utile in caso di connettività ad Internet limitata o assente, l'analisi, l'applicazione di regole per la generazione degli allarmi ed infine l'inoltro o la condivisione dei dati verso la piattaforma cloud.

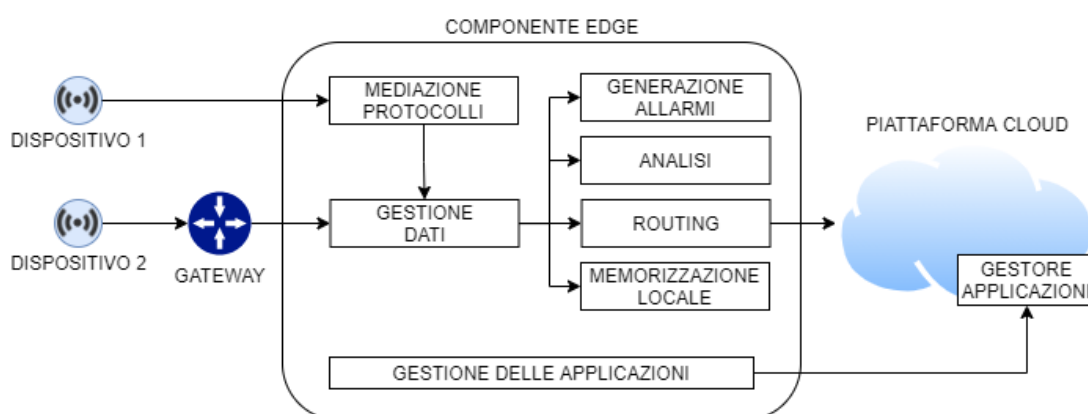


Figura 4.4: Componente edge: modello dei componenti

In quest'ottica devono essere effettuate alcune valutazioni ed assunzioni; prima di tutto questo componente non potrà avere grandi dimensioni in quanto dovrà essere collocato nell'abitazione del paziente, questa limitazione è in forte contrasto con la potenza computazionale, in quanto tali caratteristiche solitamente sono direttamente proporzionali tra loro. Inoltre, la potenza di calcolo di tale componente può variare notevolmente in base alla tipologia di applicazioni per la quale si vuole sfruttare, si pensi ad esempio

alla differenza tra i casi di utilizzo a scopo di elaborazione dei dati o per il solo inoltro degli stessi, per queste motivazioni potrebbe essere utile fornire a questo componente caratteristiche come modularità ed estendibilità. Queste caratteristiche potrebbero essere soddisfatte considerando tale componente come un cluster di dispositivi interconnessi tra loro per collaborare ed aumentare la potenza computazionale complessiva, questa modellazione gli permetterebbe inoltre di adattarsi alla tipologia di applicazione aggiungendo nuovi dispositivi su richiesta in base all'esigenza.

In figura 4.5 è mostrata una possibile architettura per soddisfare tali richieste, sono presenti 3 dispositivi interconnessi dove il primo rappresenta il componente “manager”, che avrà il compito di gestire il cluster e comunicare con l'esterno, e gli altri sono “Worker”, utilizzati per estendere la capacità computazionale. Dato che in questa soluzione vengono utilizzati più componenti sarà necessario la condivisione di risorse all'interno del cluster, a tale scopo è possibile utilizzare un sistema di condivisione del File System come ad esempio il Network File System (NFS), che possiamo trovare nell'immagine come ultimo livello di ogni dispositivo. Per la gestione delle applicazioni e delle funzionalità potrebbero tornare molto utili le tecnologie di containerizzazione come ad esempio Docker, in questo modo si facilita la distribuzione delle applicazioni inglobandole all'interno di container ed eseguendole quando e dove si preferisce; inoltre vi sono software che permettono l'orchestrazione dei container all'interno di un cluster, in questa soluzione queste tecniche possono quindi risultare molto utili. Al di sopra dello strato Docker possiamo trovare alcuni esempi di applicazioni containerizzate, tra le più importanti possiamo trovare l'agente, che permette la gestione dell'hub e delle applicazioni dall'esterno, che quindi funge da intermediario tra il componente e la piattaforma, il registro, per il salvataggio delle immagini dei software applicativi in locale in modo tale che sia possibile fare il deploy di applicazioni anche in assenza di connessione, ed un broker MQTT pensato per integrare dispositivi attraverso l'uso di tale protocollo.

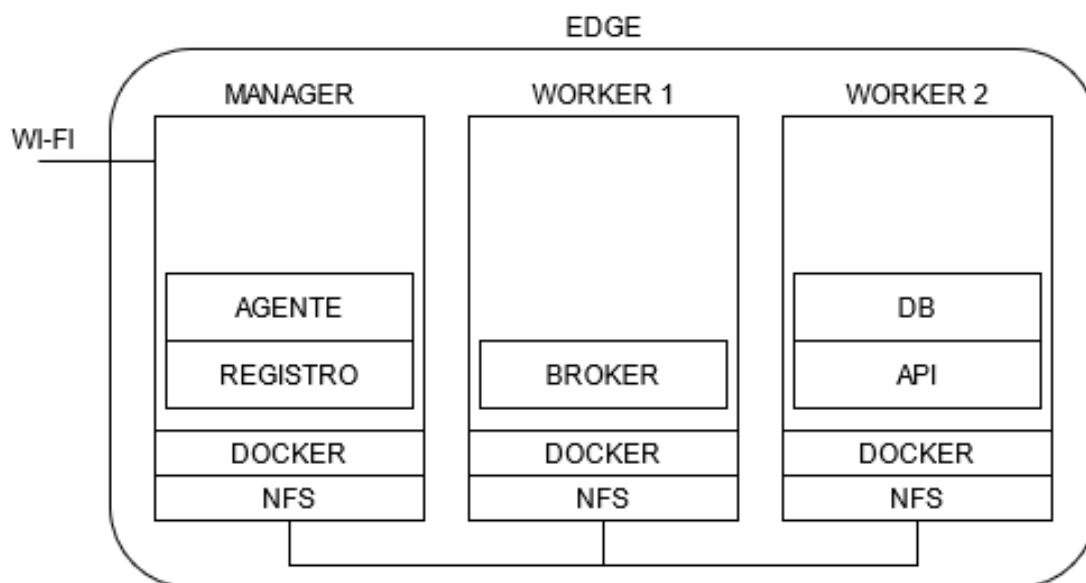


Figura 4.5: Componente edge: architettura

Dato che nei requisiti non sono richieste funzionalità avanzate in prossimità dei dati che giustificano la progettazione di tale componente, si è scelto di considerare la soluzione più semplice, quindi l'utilizzo di esso solo allo scopo di integrare i dispositivi che non rientrano nelle casistiche precedentemente presentate; con questo non si vuole però limitare una progettazione più estesa in sviluppi futuri, che potrebbe rimpiazzare il componente attuale.

4.2 Trasformazione dei dati

Dopo aver raccolto i dati vi è una fase di trasformazione per renderli conformi al modello utilizzato dalla piattaforma; prima di tutto è necessario definire il modello dei dati che si vogliono supportare, questo è mostrato in figura 4.6 e suddivide i dati in 3 strutture relative alle tipologie di dato che si vogliono raccogliere, quindi troviamo rispettivamente i parametri relativi alle misurazioni effettuate, quelli relativi alle attività fisiche ed infine quelli ambientali.

MEASUREMENTS		ACTIVITIES		ENVIRONMENT	
Parametro	Unità di misura	Parametro	Unità di misura	Parametro	Unità di misura
spo2	%	steps	-	temperature	°C
heart_rate	bpm	calories_burned	MET	humidity	%
weight	Kg	distance	m	pm10	ug/m3
body_temperature	°C	elevation	floors	pm2,5	ug/m3
systolic_pressure	mmHg	average_heart_rate	bpm	vocs	ppb
diastolic_pressure	mmHg	max_heart_rate	bpm	co2	ppm
height	m	min_heart_rate	bpm	pressure	hPa
respiratory_rate	bpm	duration	s	ozone	ppb
body_fat_percentage	%	start_time	-	carbon_monoxide	ppm
bmi	-	initial_spo2	%	nitrogen_dioxide	ppb
daily_steps	-	min_spo2	%		
fev1	L	desaturation_speed	s		
pef	L/s				

Figura 4.6: Modello dei dati

È possibile dividere il percorso di trasformazione dei dati, dal formalismo utilizzato dai dispositivi con i quali sono stati raccolti a quello della piattaforma, in diverse operazioni:

- indirizzamento alla tipologia corretta, dato un record deve essere collocato nella giusta struttura tra quelle viste sopra, quindi i dati ambientali dovranno essere indirizzati verso environment, le informazioni relative agli allenamenti in activities ed i parametri misurati in measurements;
- traduzione del formato, dato un record tradurre i campi dalla notazione utilizzata dai dispositivi a quella del modello della piattaforma, quindi se dovesse arrivare una misurazione di temperatura corporea con un campo “temp” dovrebbe essere trasformato in “body_temperature”;
- trasformazione del contenuto, dato un record trasformare il contenuto dei campi per renderli conformi al modello, un esempio di questa tipologia di operazioni è la conversione dell’unità di misura, se dovesse essere fornita una misurazione in gradi Fahrenheit dovrebbe essere trasformata in gradi Celsius.

Per effettuare queste operazioni si è pensato di dividerle in due componenti: uno chiamato Dispatcher, che si occuperà dell’instradamento dei dati verso la tipologia corretta,

e l'altro chiamato Mapper, che effettuerà le operazioni di traduzione del formato e trasformazione del contenuto. Questa scelta è stata effettuata per due motivi principali: la separazione dei concetti, delegando il primo componente al solo indirizzamento ed il secondo alla trasformazione del record, e per questioni di dimensioni e scalabilità dei componenti, disaccoppiando quindi i due compiti che hanno complessità differenti in modo da poter scalare solo quelli che ne hanno necessità.

Una volta definiti i due componenti si deve decidere dove dovranno essere collocati all'interno della piattaforma, quindi se conviene inserirli in prossimità del paziente o nella piattaforma cloud; nel primo caso si ha il vantaggio di ridurre notevolmente i tempi di latenza, che potrebbe essere molto significativo nella realizzazione di applicazioni tempo-dipendenti come la generazione di allarmi, ma significherebbe fare assunzioni molto forti sulla potenza computazionale dei dispositivi lato utente, che non vogliono essere imposte nel nostro caso. Collocare tali componenti lato utente comporterebbe due grandi svantaggi: uno dal punto di vista della perdita di controllo, in quanto non è possibile controllare in maniera fine tali dispositivi, e l'altro riguarda la loro progettazione e distribuzione in quanto sarebbe necessario realizzarne uno per lo smartphone ed uno per il componente edge. Per queste motivazioni si è scelto di collocare questi componenti nel lato cloud della piattaforma.

I componenti Dispatcher e Mapper devono essere progettati per essere estendibili e flessibili, con queste caratteristiche si vuole intendere che l'aggiunta, la modifica o la rimozione di dispositivi, piattaforme o componenti utilizzati per la raccolta dati, non debbano essere operazioni che rivoluzionino la piattaforma, ma devono essere rese il più semplice possibile. A tale scopo si è pensato di portare la configurazione dei componenti al loro esterno in modo da poterli configurare dinamicamente senza doverne modificare il codice; per questa soluzione è possibile utilizzare un database NoSQL nel quale saranno contenute le configurazioni di Dispatcher e Mapper.

4.2.1 Dispatcher

Il Dispatcher è un componente semplice che dato un record in ingresso lo assegna ad una o più tipologie in uscita; è da considerare anche il caso che un record possa appartenere a più categorie in quanto potrebbero esserci dispositivi che forniscono misurazioni

di più parametri, ad esempio uno smartwatch permette la raccolta di informazioni riguardanti le attività fisiche ma anche la misurazione di parametri vitali come il battito cardiaco. Una volta ricevuto un record, il Dispatcher dovrà ottenere l'informazione che identifica il dispositivo, che sarà fornita nella fase di raccolta del dato, leggere le tipologie di dato a cui deve essere assegnato dal database delle configurazioni ed inoltrarlo al Mapper. Per le configurazioni del Dispatcher si è pensato di utilizzare un documento JSON e rappresentare ogni dispositivo attraverso un campo che ha come chiave l'identificativo del device e come valore la lista delle tipologie di dato a cui appartiene, è possibile vedere un esempio di configurazione in figura 4.7; in questo esempio vengono mostrati l'identificativo del documento e le configurazioni di 3 dispositivi.

```
{
  "_id": "dispatcher",
  "bme280": [
    | "environment"
  ],
  "garmin vivoactive 3": [
    | "activities",
    | "measurements"
  ],
  "withings thermo": [
    | "measurements"
  ]
}
```

Figura 4.7: Esempio di configurazione del Dispatcher

4.2.2 Mapper

Il Mapper è un componente che dato un record in ingresso lo trasforma sia nel formato che nel contenuto per renderlo conforme al modello utilizzato dalla piattaforma; il suo

comportamento è più complesso di quello del componente precedente perché oltre ad analizzare il record dovrà effettuare modifiche di due tipologie differenti, di conseguenza risulterà più complesso anche il documento di configurazione.

Per capire il funzionamento del Mapper può essere utile innanzitutto illustrare come si è pensato di rappresentare le sue configurazioni, un esempio è mostrato in figura 4.8; prima di tutto è necessario fare una precisazione, si è scelto di dividere le configurazioni del Mapper per le tipologie dei dati, quindi si avrà un documento per i parametri, uno per le attività ed uno per i dispositivi ambientali. In ogni documento è possibile trovare un insieme di campi la cui chiave è l'identificativo del dispositivo, come per le configurazioni del Dispatcher, ed il contenuto è un oggetto JSON che definisce la logica di trasformazione del record per ogni parametro che si vuole considerare. Ogni campo di questo oggetto avrà come chiave il nome utilizzato dal dispositivo per identificare quel parametro e come contenuto un oggetto JSON che definisce il nuovo nome che dovrà assumere e le trasformazioni che dovranno essere eseguite per renderlo conforme al modello della piattaforma.

Nell'esempio riportato è possibile vedere la configurazione di un Mapper per i record ambientali, è presente il solo dispositivo bme280 che ci fornirà i parametri "temp" come stringa in gradi Fahrenheit ed "hum"; il primo dovrà essere rinominato in "temperature" ed al suo contenuto dovranno essere applicate due trasformazioni, la prima per pulire il dato, estraendo solamente il valore dalla stringa, e la seconda per convertirlo in gradi Celsius. Il secondo campo invece dovrà solamente essere rinominato in "humidity" in quanto la sua unità di misura è già corretta e non sono previste trasformazioni del contenuto.

```
{
  "_id": "environment-mapper",
  "bme280": {
    "temp": {
      "model-name": "temperature",
      "transformations": [
        "cleanData",
        "fahrenheitToCelsius"
      ]
    },
    "hum": {
      "model-name": "humidity",
      "transformations": []
    }
  }
}
```

Figura 4.8: Esempio di configurazione del Mapper

4.3 Memorizzazione ed applicazioni

Una volta che i dati sono stati trasformati per essere conformi al modello della piattaforma, sarà possibile utilizzarli principalmente per due scopi: per la loro memorizzazione o per essere utilizzati da applicazioni di natura tempo-dipendente, infatti questo è il punto più vicino alla raccolta dati nel quale essi possono essere utilizzati attraverso i formalismi della piattaforma. Un'applicazione di questa tipologia è la generazione di allarmi che non dovrebbe essere eseguita dopo la memorizzazione dei dati per ridurre il più possibile il tempo di intervento, questo componente quindi otterrà i dati direttamente dal Mapper per analizzarli ed effettuare azioni qualora essi risultino allarmanti.

Per la memorizzazione dei dati si è pensato di utilizzare un database NoSQL dove ogni

tipologia di dato viene rappresentata da una collezione di documenti, inoltre è necessario aggiungere un componente chiamato “Worker” che si occupi della memorizzazione dei dati ricevuti dal Mapper, il suo compito è abbastanza semplice e consiste nell’individuare a quale tipologia di dato appartiene il record, informazione ottenuta dal Dispatcher, e salvarlo nel database corrispondente. Dopo che i dati sono stati salvati potranno essere utilizzati per tutte le tipologie di applicazioni, per il loro processamento, per farne analitica ed ottenerne inferenza, o per la loro presentazione attraverso un’interfaccia grafica; non è di rilevante interesse per questa tesi la progettazione di questi servizi ma è sufficiente predisporre l’ambiente in modo tale che possano ottenere i dati per i loro scopi. Alcune tipologie di applicazioni potrebbero inoltre necessitare di dati più strutturati, a tale fine si è pensato di aggiungere un servizio con lo scopo di leggere i dati memorizzati nel database NoSQ, elaborarli e trasformarli per poi salvarli in un database SQL.

Per rendere i dati accessibili in tutta la piattaforma si è pensato di realizzare dei servizi interni che esponano delle RESTful API e che hanno lo scopo di raccogliere le informazioni dai database o da altri servizi, effettuare elaborazioni e restituirli ai richiedenti; per elaborazioni si intendono tutte quelle operazioni sui dati come il campionamento, l’estrazione di informazioni da un insieme di dati, l’aggregazione di dati di tipologie differenti o provenienti da servizi diversi. Per rendere i dati accessibili da applicazioni di terze parti invece, è sembrato opportuno aggiungere un componente che funga da intermediario tra i servizi interni e le richieste esterne alla piattaforma. Questo componente è chiamato API gateway e potrà essere utilizzato per gestire e proteggere le richieste alle API provenienti dall’esterno, intercettandole ed eseguendo politiche di sicurezza prima di inoltrarle al servizio interessato.

4.4 Sicurezza e privacy

Per quanto concerne la parte di sicurezza e privacy, la piattaforma dovrà prima di tutto essere GDPR compliant, questo significa che dovrà rispettare le caratteristiche presenti in tale regolamentazione; in questa sezione vengono approfondite tali caratteristiche e gli aspetti legati alla sicurezza e privacy della piattaforma e dei dati, dal punto di vista del design e della progettazione, tralasciando la gestione delle responsabilità ed

il consenso al trattamento dei dati, che dovranno essere approfonditi e realizzati quando avverrà l'implementazione a livello di produzione.

Per quanto riguarda la normativa GDPR, i principi sul quale è possibile lavorare a livello di design e progettazione sono quelli che riguardano la gestione dei dati, cioè come possiamo garantire alcuni diritti come quello di accesso, rettifica, eliminazione e migrazione; una soluzione potrebbe essere la progettazione di un servizio che si occupi di questo compito, ossia dovrà offrire delle API che, dato in ingresso l'identificativo del paziente, permetta di effettuare tutte le operazioni precedentemente citate. L'operazione di accesso ai propri dati personali sarà la più semplice in quanto dovrà solamente interrogare i database per raccogliere tutte le informazioni di un determinato paziente e fornirle in output, questa funzionalità può essere inoltre utilizzata per la migrazione dei dati verso un altro fornitore di servizi, esportando i dati e fornendoli al diretto interessato sotto richiesta dell'utente. Il diritto di rettifica può essere ottenuto realizzando una API che permetta di modificare i record contenuti nei database fornendo in ingresso l'identificativo del paziente, le informazioni per identificare il record e quelle che permettono di definire come dovrà essere modificato; infine per l'eliminazione dei dati del paziente dovrà essere realizzata una API che, dato l'identificativo del paziente, permetta di eliminare parzialmente o completamente le informazioni di quest'ultimo, avendo cura di innescare lo stesso meccanismo di cancellazione anche a tutte le applicazioni che utilizzano questi dati. Le altre normative comprese nel GDPR ma di cui non si discute nella progettazione sono la locazione in cui vengono memorizzati i dati ed il diritto di notifica in caso di compromissione, questi dovranno essere trattati nella fase di implementazione a livello di produzione, in quanto dipendono dalle scelte dei fornitori di servizi che l'azienda dovrà compiere nel momento di realizzazione della piattaforma; l'unico consiglio che può essere fornito è quello di automatizzare la verifica di vulnerabilità attraverso gli strumenti che solitamente vengono messi a disposizione dai fornitori di servizi cloud.

Un primo aspetto quando si tratta di sicurezza è l'autenticazione degli utenti per accedere alla piattaforma, per rendere sicura questa operazione vi sono alcune tecniche introdotte nella fase di analisi dei requisiti ma per la progettazione di questa parte si è scelto di risolvere il problema implementando una soluzione del tipo single sign-on basata sul Sistema pubblico di identità digitale (SPID); questa soluzione consiste nel delegare

l'operazione di autenticazione ai gestori di identità digitale, chiamati Identity provider, che si occuperanno della verifica delle credenziali dell'utente e di tutte le sue proprietà, compresa la sicurezza di tale procedura.

La piattaforma deve essere progettata in modo tale che i dati all'interno di essa non vengano mai trasmessi o memorizzati in chiaro, ma sempre dopo aver subito operazioni di crittografia per renderli illeggibili da entità esterne ad essa, questo è l'unico modo per gestire i dati sensibili con sicurezza, rispettando i requisiti normativi e le politiche sulla privacy. Devono essere quindi considerati i meccanismi e le tecniche da utilizzare a tale scopo, per quanto riguarda i dati in transizione è possibile utilizzare comunicazioni end-to-end sicure che sfruttino protocolli come il Transport layer security (TSL) per trasmettere le informazioni criptate, e garantire quindi il massimo livello di sicurezza ottenibile in tale processo. Per quanto riguarda la memorizzazione dei dati, è opportuno utilizzare un algoritmo di cifratura robusto, come ad esempio AES, per cifrarli prima del loro salvataggio; dato che la fase di memorizzazione è eseguita dal componente Worker, si ritiene che tale operazione potrebbe essere eseguita anch'essa da quest'ultimo. Di conseguenza, tutti i servizi che utilizzano i dati salvati all'interno dei database dovranno essere in grado di decifrarli prima di poterli sfruttare.

Un ulteriore livello di sicurezza può essere inoltre fornito dal componente API gateway, che funge da intermediario tra le richieste dall'esterno della piattaforma ed i servizi interni; questo servizio potrà analizzare e gestire le richieste attuando politiche di sicurezza, come ad esempio l'implementazione di funzionalità di controllo degli accessi basato sui ruoli (RBAC) per limitare l'accesso o la modifica dei dati in maniera più fine.

Come visto nella fase di analisi dei requisiti, è necessario che i dati dei pazienti legati allo stato di salute siano de-identificati, ossia che non contengano informazioni che possano ricondurli al paziente dal quale sono stati raccolti; per questo motivo verrà assegnato ad ogni paziente un codice identificativo casuale, a tale scopo potrebbe essere utilizzato un UUID, che lo rappresenta all'interno della piattaforma e che sarà l'unico modo per ricondurre le informazioni ai pazienti dai quali sono state raccolte. Dovranno essere realizzati due database separati, uno per i dati anagrafici ed uno per quelli relativi allo stato del paziente, questi avranno inoltre permessi di accesso differenti in quanto i primi sono considerati dati sensibili altamente confidenziali e potranno essere acceduti

solamente dai soggetti autorizzati.

4.5 Architettura

In figura 4.9 viene mostrata l'architettura di riferimento della piattaforma che è stata pensata come risultato dell'analisi effettuata. Lo stile architetturale utilizzato per la progettazione della parte cloud della piattaforma è quello a microservizi. Per l'interazione tra i vari componenti si è scelto di adottare due diverse metodologie: le RESTful API per l'interazione con componenti esterni alla piattaforma e la comunicazione asincrona basata sullo scambio di messaggi per quella tra i servizi interni ad essa; dato che i componenti smartphone ed edge sono parti integranti della piattaforma e dovranno essere realizzati insieme ad essa, rientrano anch'essi nella seconda metodologia da utilizzare. In particolare, per la comunicazione asincrona basata sullo scambio di messaggi si è pensato di sfruttare un Message-oriented middleware (MOM) in modo tale da facilitare la gestione di tale operazione e usufruire di tutti i vantaggi e delle proprietà forniti da esso, soprattutto per ciò che riguarda le caratteristiche relative alla qualità del servizio.

L'architettura della piattaforma è stata divisa in 5 layer:

- raccolta dati, che contiene le modalità con le quali è possibile raccogliere dati dai dispositivi;
- integrazione, che contiene i servizi che permettono di integrare componenti esterni, come quelli che consentono la trasformazione dei dati raccolti, e l'API gateway, per esporre i dati alle applicazioni di terze parti;
- microservizi, dove sono situati tutti i servizi che lavorano sui dati conformi al modello della piattaforma;
- database, rappresenta lo strato dove verranno memorizzati i dati, in esso sono compresi sia quelli SQL che quelli NOSQL;
- applicazioni di terze parti, che contiene le applicazioni di terze parti che utilizzano i dati forniti dalla piattaforma.

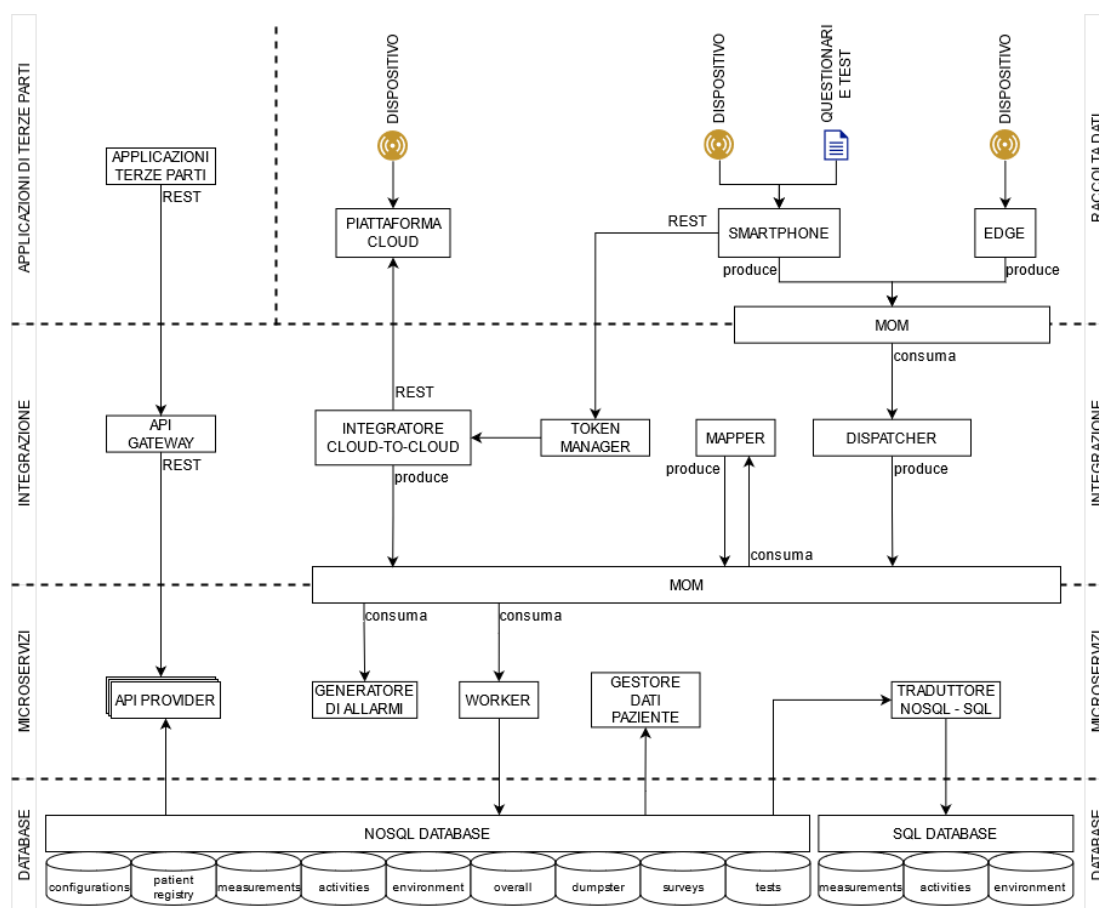


Figura 4.9: Architettura di riferimento della piattaforma

Nello strato di raccolta dati possiamo trovare tre dispositivi per differenziare le diverse metodologie che possono essere utilizzate, quindi attraverso le piattaforme cloud, sia dei produttori di dispositivi che degli aggregatori (ad esempio Google Fit e Validic), attraverso il componente edge o utilizzando un'applicazione installata sullo smartphone del paziente, che consentirà inoltre di compilare questionari ed effettuare test. Subito sotto possiamo trovare i servizi necessari per l'integrazione, quindi quelli che una volta ottenuti i dati li trasformano per renderli conformi al modello della piattaforma, tra questi è possibile vedere il Dispatcher ed il Mapper di cui si è già discusso precedentemente, l'Integratore cloud-to-cloud che è il componente per ottenere dati dalle piattaforme cloud, ed il Token manager che verrà utilizzato per gestire i token di accesso dell'autenticazione OAuth 2.0. I componenti di questi due strati possono interagire utilizzando RESTful

API o attraverso un MOM, la prima metodologia è forzata per integrare i dati delle piattaforme cloud esterne, in quanto è il modo più comunemente utilizzato da queste per esporre i dati raccolti; si è inoltre pensato di utilizzare due MOM separati, uno per la comunicazione tra i servizi interni alla piattaforma ed uno per interagire con smartphone e componente edge, in quanto si suppone che le due casistiche abbiano livelli di rischio differenti, e di conseguenza si ritiene utile la compartimentalizzazione di tali responsabilità. In particolare si pensa che il middleware tra il Dispatcher e la raccolta dati possa essere più vulnerabile in quanto è esposto verso dispositivi di cui non è possibile avere pieno controllo, per questo potrebbero essere quindi necessari maggiori controlli per quanto riguarda la sicurezza e le vulnerabilità.

Al di sotto del MOM situato tra lo strato di integrazione e dei microservizi, i dati saranno conformi al modello della piattaforma e potranno essere utilizzati per le applicazioni tempo-dipendenti, come esempio vi è il servizio per la generazione di allarmi, e per la loro memorizzazione, attraverso il componente Worker. Nello strato dei microservizi possiamo inoltre trovare servizi come il gestore dei dati dei pazienti, che verrà utilizzato per soddisfare le richieste degli utenti per rendere la piattaforma GDPR compliant come spiegato in sezione 4.4, il Traduttore NOSQL - SQL, che si occupa di leggere i dati dal database NOSQL, elaborarli e salvarli in quello SQL, ed i servizi API provider, che hanno lo scopo di leggere i dati dal database NOSQL, decifrarli, elaborarli e fornirli attraverso RESTful API. Nello strato inferiore è possibile trovare i database, sia quello NOSQL che SQL, ed al di sotto di essi sono rappresentate le collezioni che si vogliono supportare, in particolare possiamo ritrovare quelle di cui si è precedentemente discusso ed alcune aggiuntive; partendo da sinistra possiamo vedere:

- configurations, contenente tutte le configurazioni necessarie, come ad esempio quelle del dispatcher e del mapper;
- patient registry, contenente l'anagrafe dei pazienti;
- measurements, activities ed environment, che rappresentano le tipologie di dato attualmente supportate dalla piattaforma e conterranno rispettivamente le informazioni riguardo i parametri di salute, delle attività e quelli ambientali;

- overall, pensato per contenere tutti i dati raccolti prima delle operazioni di trasformazione, potrebbe essere utile per recuperare i dati originali qualora si verificassero problemi o si volessero effettuare controlli;
- dumpster, pensato come contenitore nel quale inserire tutti i dati non riconosciuti, come ad esempio dai dispositivi non supportati dalla piattaforma, potrebbe essere utile per identificare eventuali problemi o per la scelta di nuovi dispositivi da supportare;
- surveys e tests, che contengono i dati relativi ai questionari ed ai test effettuati.

Infine, risalendo verso le applicazioni di terze parti è possibile trovare il componente API gateway, che ricoprirà il ruolo di intermediario tra le applicazioni esterne alla piattaforma ed i servizi API provider, applicando funzionalità di controllo degli accessi e politiche di sicurezza.

4.6 Mockup caso d'uso Covid-19

In questa sezione viene illustrato come sono state progettate le interfacce principali delle applicazioni del paziente e del medico.

4.6.1 Applicazione del paziente

In figura 4.10 è possibile vedere la schermata di riepilogo dei dispositivi personali e delle piattaforme cloud connesse dal paziente, è la prima mostrata una volta effettuato l'accesso all'applicazione; da questa interfaccia sarà possibile vedere e modificare l'anagrafica del paziente cliccando sull'icona in alto a destra, connettere nuovi dispositivi o piattaforme cloud con il pulsante in basso a destra o rimuovere quelli attualmente connessi. Inoltre, con il menù in alto a sinistra sarà possibile spostarsi nella schermata di riepilogo dei questionari e test spiegata in seguito.

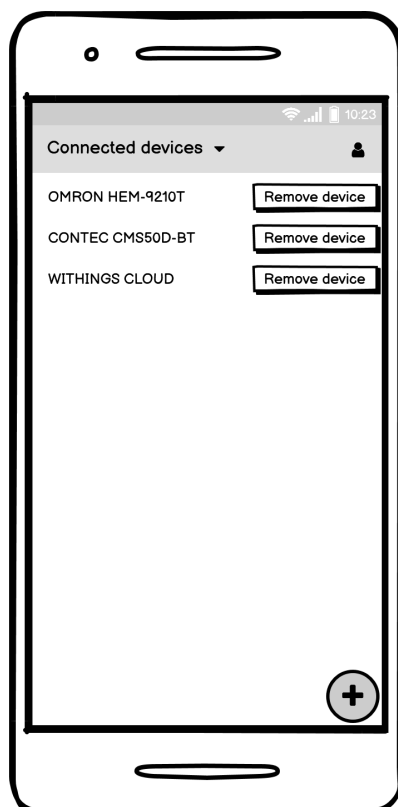


Figura 4.10: Mockup applicazione del paziente: schermata dispositivi connessi

In figura 4.11 è mostrata la schermata di riepilogo dei questionari e test effettuati; da questa sarà possibile visualizzare o eliminare i questionari ed i test svolti, aggiungerne di nuovi, visualizzare o modificare l'anagrafica e tornare alla schermata precedente. Nella parte superiore sono presenti il menù per passare alla schermata di riepilogo dei dispositivi ed il pulsante per l'anagrafica come visto precedentemente, nella parte centrale vi sono i test ed i questionari, e per ognuno di essi sarà possibile la visualizzazione o la cancellazione. Infine, in basso a destra è presente un pulsante che, alla sua pressione, aprirà una tendina nel quale sarà possibile scegliere quello che si vuole aggiungere, se un questionario, un walking test o uno speech test.

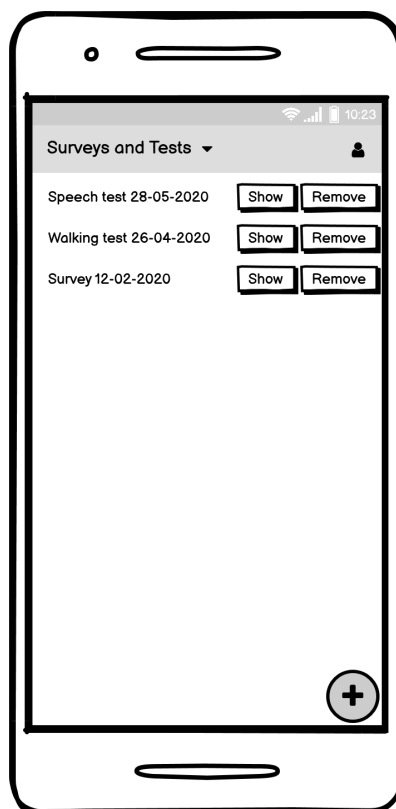
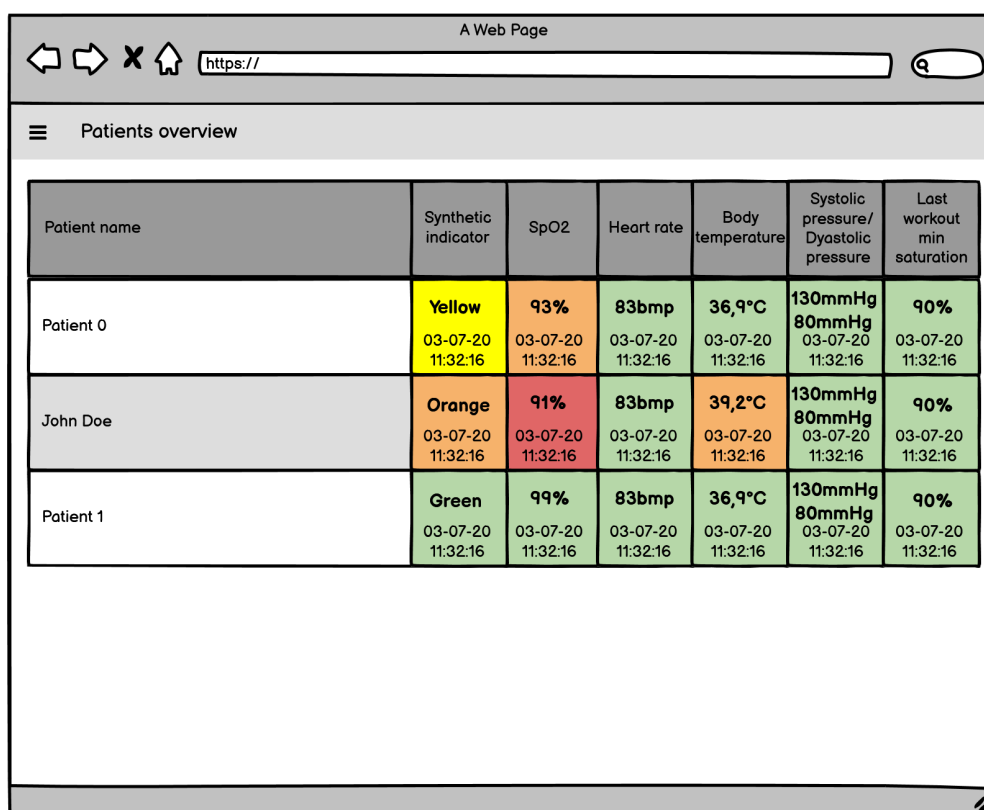


Figura 4.11: Mockup applicazione del paziente: schermata questionari e test effettuati

4.6.2 Applicazione del medico

Per quanto riguarda l'applicazione del medico si è pensato di mostrare le due schermate considerate più rilevanti: una contenente la panoramica generale di tutti i pazienti in cura presso quel professionista mentre l'altra mostra il dettaglio di un singolo paziente; la prima è visibile in figura 4.12 e mostra le informazioni più importanti determinate in fase di analisi. Nell'angolo in alto a sinistra dell'immagine è possibile trovare il menù che, una volta aperto, permetterà al medico di effettuare il logout. Nella parte centrale è visibile il riepilogo dei pazienti e le informazioni che consentono una prima valutazione dello stato di salute di ognuno, in particolare è possibile ritrovare l'indicatore sintetico che, in base al colore assunto, fornisce un primo riscontro veloce per individuare i casi che richiedono maggiore attenzione. Cliccando su uno dei parametri presenti nel riepilogo sarà possibile visualizzarne lo storico delle misurazioni, mentre cliccando sul nome di un paziente si potrà passare alla schermata che contiene il suo dettaglio.



Patient name	Synthetic indicator	SpO2	Heart rate	Body temperature	Systolic pressure/ Diastolic pressure	Last workout min saturation
Patient 0	Yellow 03-07-20 11:32:16	93% 03-07-20 11:32:16	83bmp 03-07-20 11:32:16	36,9°C 03-07-20 11:32:16	130mmHg 80mmHg 03-07-20 11:32:16	90% 03-07-20 11:32:16
John Doe	Orange 03-07-20 11:32:16	91% 03-07-20 11:32:16	83bmp 03-07-20 11:32:16	39,2°C 03-07-20 11:32:16	130mmHg 80mmHg 03-07-20 11:32:16	90% 03-07-20 11:32:16
Patient 1	Green 03-07-20 11:32:16	99% 03-07-20 11:32:16	83bmp 03-07-20 11:32:16	36,9°C 03-07-20 11:32:16	130mmHg 80mmHg 03-07-20 11:32:16	90% 03-07-20 11:32:16

Figura 4.12: Mockup applicazione del medico: schermata panoramica dei pazienti

Infine, in figura 4.11 è mostrato il dettaglio di tutte le informazioni raccolte dal paziente selezionato. Questa schermata è divisa in due parti, nella parte alta è visibile una rappresentazione grafica dello storico dei dati più importanti attraverso un grafico a linee, mentre nella parte inferiore è possibile vedere lo storico dei dati in formato tabellare. Per quanto riguarda il grafico, si è pensato di renderlo personalizzabile, consentendo al medico di cliccare sui parametri per aggiungerli o rimuoverli da esso, e di conseguenza permettergli di visualizzare solo quelli che ritiene rilevanti. Per la rappresentazione tabellare si è pensato di suddividere i dati in più schede: Summary che contiene il sommario delle misurazioni effettuate, Surveys che mostra i questionari effettuati, Walking tests e Speech tests per visualizzare i risultati ottenuti dai test eseguiti, ed infine Workouts che contiene tutte le informazioni relative alle attività fisiche svolte.

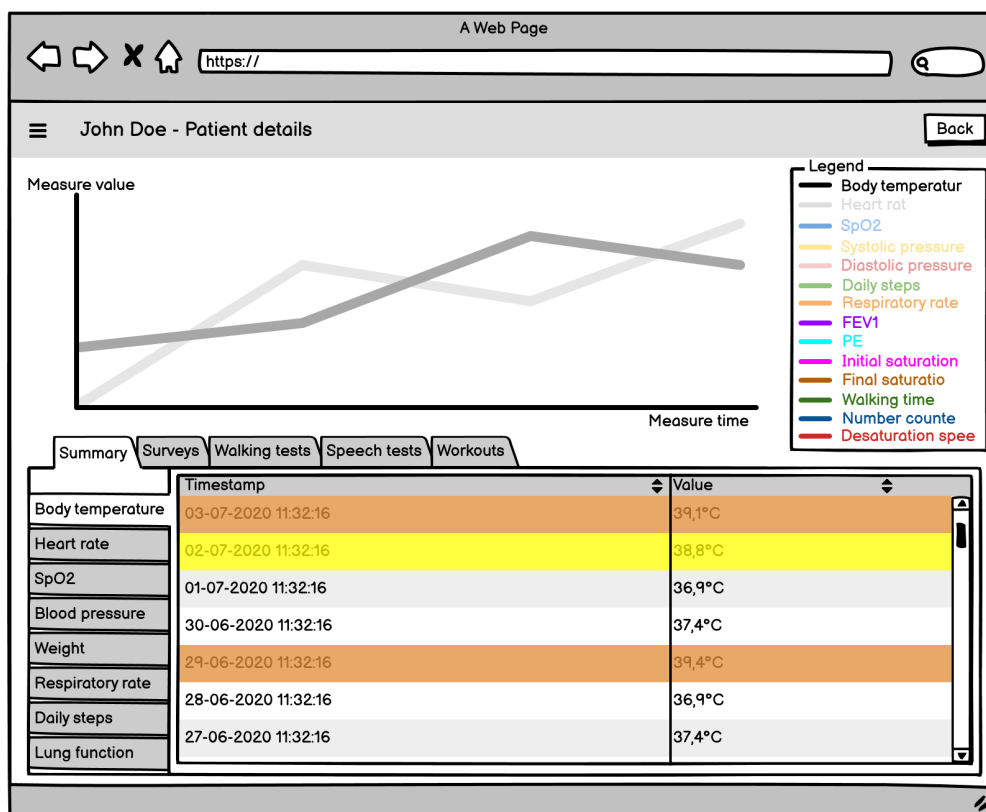


Figura 4.13: Mockup applicazione del medico: schermata dettaglio del paziente selezionato

Capitolo 5

Sviluppo prototipale e validazione

In questo capitolo verrà illustrato lo sviluppo prototipale della piattaforma che è stato effettuato a scopo di Proof of Concept per validare l'architettura che è stata proposta, verificando che sia veramente realizzabile. In particolare ci si è voluti concentrare sul flusso dati all'interno della piattaforma cloud tralasciando le parti che si occupano della raccolta e presentazione del dato; a tale scopo si è pensato di utilizzare un componente mock che simuli l'invio dei dati da parte dello smartphone, e per validare il funzionamento dell'intero processo è stato utilizzato lo strumento Postman per ottenere i dati raccolti, interrogando le API esposte dai servizi e dall'API gateway. Nella realizzazione del prototipo sono stati inoltre tralasciati gli aspetti di sicurezza, come le fasi di cifratura e decifratura, perché non ritenuti funzionali alle verifiche che ci si è proposti di eseguire.

In figura 5.1 è mostrata una panoramica generale della piattaforma sviluppata a livello prototipale, vista come estensione dell'architettura esposta nel capitolo precedente; in questa immagine si sono volute codificare alcune delle scelte tecnologiche che sono state effettuate ed alcuni dei servizi cloud utilizzati, inoltre sono stati evidenziati in verde i componenti che sono stati realizzati in questa fase. I componenti nell'immagine che non risultano evidenziati sono quelli sul quale non si è lavorato, perché ritenuti secondari rispetto le verifiche che ci si è posti di effettuare sull'architettura, ma che si è comunque scelto di rappresentare. Tra le parti non realizzate è possibile trovare: il componente edge e l'applicazione per lo smartphone, il servizio per la generazione di allarmi, la fase di trasformazione dei dati da NOSQL a SQL, il servizio per la gestione dei dati dei

pazienti e quello per la gestione dei token; inoltre non sono stati trattati i questionari ed i test, per questo motivo non sono stati predisposti nemmeno i database per contenere tali informazioni.

Continuando nella lettura del capitolo saranno mostrate le tecnologie, i framework, ed i servizi cloud utilizzati per la realizzazione e distribuzione dei vari componenti.

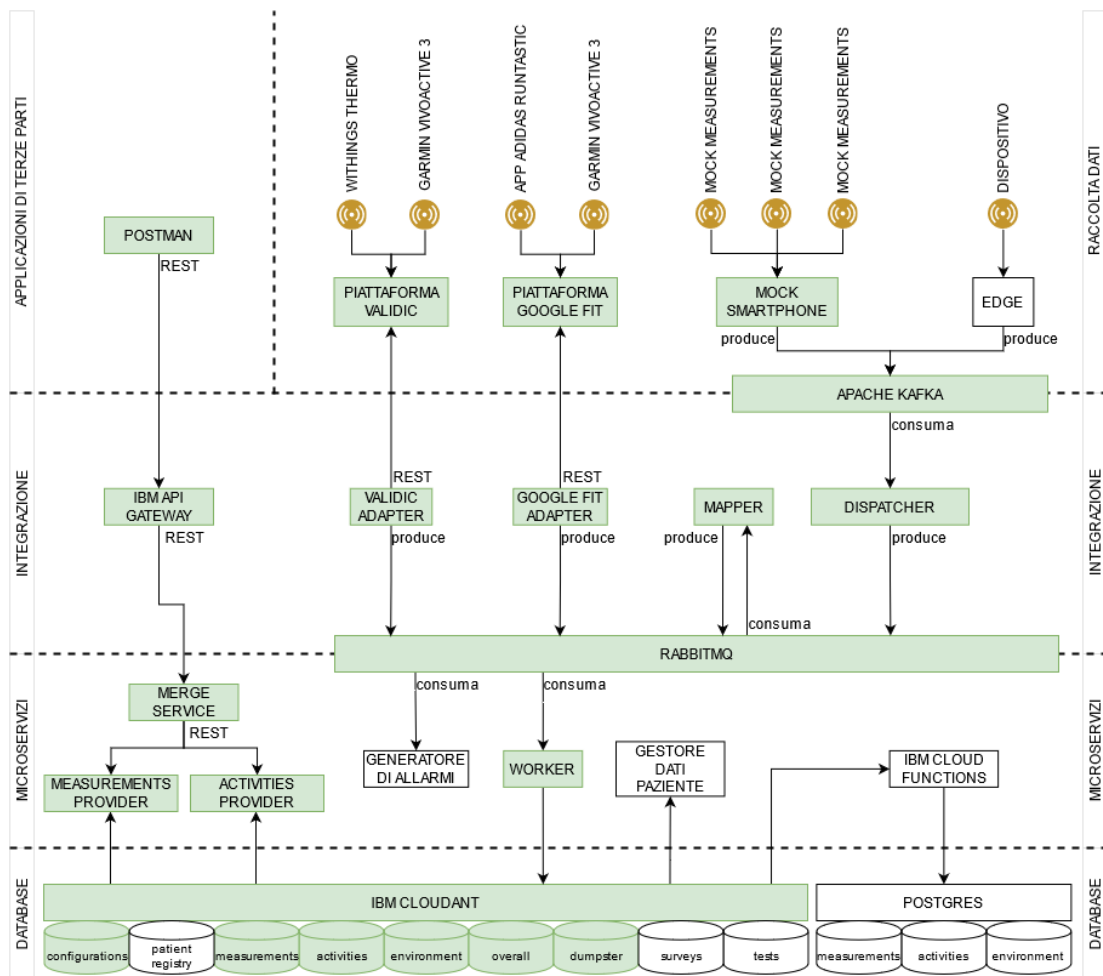


Figura 5.1: Panoramica dell'architettura utilizzata per lo sviluppo prototipale

5.1 Framework e tecnologie

In questa sezione verranno descritti i framework utilizzati e le scelte tecnologiche intraprese per la fase di sviluppo prototipale, facendo riferimento all'architettura di figura 5.1.

5.1.1 Apache Kafka

Apache Kafka è una piattaforma distribuita di data streaming che permette lo scambio di informazioni in modo asincrono tra più processi, in particolare supporta l'invio di messaggi da più sorgenti, o produttori, verso a più destinatari, o consumatori; tale piattaforma è nota per la sua capacità di gestire, trasferire, memorizzare e processare grandi volumi di dati in tempo reale. Per queste sue caratteristiche si è pensato di utilizzarla per la comunicazione tra lo strato di raccolta dati e quello di integrazione, dove smartphone ed edge component agiscono come produttori, che pubblicano i dati che hanno collezionato dai dispositivi locali, ed il Dispatcher come consumatore, che una volta ricevuti i record, li analizza e li indirizza verso la tipologia dati appropriata.

Nello sviluppo prototipale è stato possibile utilizzare questa soluzione in quanto si è scelto di simulare lo smartphone implementando un'applicazione in Java ma, da una successiva analisi più approfondita, è stato scoperto che non è presente una libreria Android per implementare tale funzionalità e quindi diventa più complesso utilizzare questa tecnologia per integrare la raccolta dati dai componenti esterni alla piattaforma cloud. Sono state studiate soluzioni alternative per integrare smartphone ed edge component, tra queste ne sono state individuate due più di rilievo: sostituire Kafka con RabbitMQ in quanto è presente un SDK per Android oppure mettere a disposizione delle RESTful API. La prima soluzione proposta risulta essere più performante e permette la gestione di una grande mole di dispositivi e dati ma ha lo svantaggio di vincolare la raccolta dati alla tecnologia, la seconda invece risulta meno vincolante ma anche meno performante, inoltre, per quest'ultima potrebbe essere richiesto lo sviluppo di un servizio per la gestione dell'interazione tra i componenti di raccolta ed il Dispatcher.

5.1.2 RabbitMQ

RabbitMQ è un middleware che permette la comunicazione asincrona tramite lo scambio di messaggi tra diversi processi, può essere pensato come un broker di messaggi che segue il paradigma publish-subscribe. Esso supporta diversi protocolli, come AMQP, MQTT e STOMP, ed è basato su un sistema a code, nelle quali i messaggi possono essere pubblicati da diversi produttori ed essere consumati da diversi consumatori; è una tecnologia che permette di ottenere alte performance nello scambio dei messaggi, scalabilità ed affidabilità, e per queste caratteristiche viene spesso utilizzata per implementare il meccanismo di comunicazione tra i servizi nelle architetture a microservizi. Proprio a tale scopo si è pensato di utilizzarlo nella fase di sviluppo prototipale, permettendo l'interazione tra tutti i servizi precedenti alla fase di memorizzazione del dato.

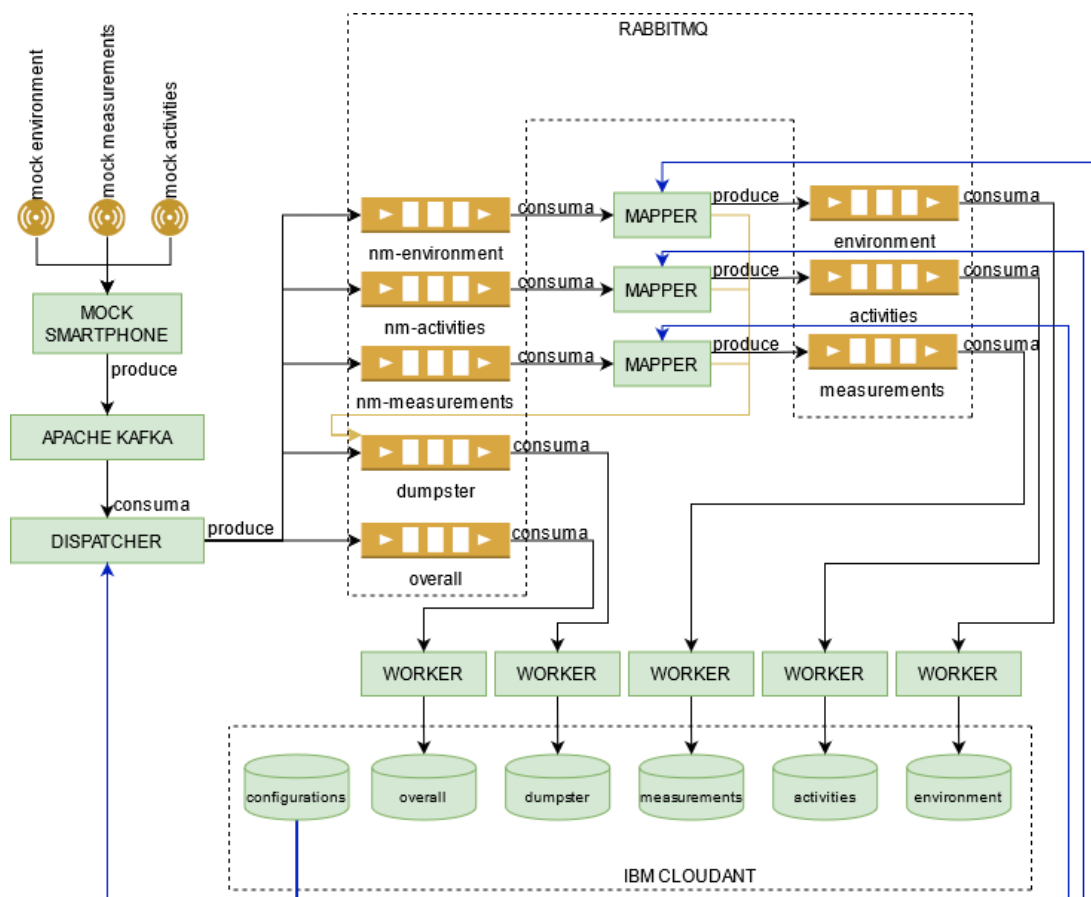


Figura 5.2: Utilizzo di RabbitMQ per la comunicazione tra Dispatcher, Mapper e Worker

In figura 5.2 è mostrata la soluzione proposta ed utilizzata per il processo di trasformazione del dato dal modello dei dispositivi a quello della piattaforma, sfruttando il broker RabbitMQ; in essa è rappresentato il caso che utilizza lo smartphone simulato come sorgente dati, a scopo illustrativo, ma potrà essere applicato sia allo smartphone fisico che al componente edge quando questi saranno realizzati. Nell'immagine è possibile vedere che sono presenti più Mapper e più Worker, questa scelta è stata effettuata per distribuire il carico di lavoro e perché si ritiene che le operazioni che verranno eseguite sulle diverse tipologie di dato potranno essere differenti, in questo modo si vogliono separare i servizi in modo tale che, qualora si volesse aggiungere una nuova operazione per una determinata tipologia, sarebbe necessario modificare ed eseguire il deploy solamente del componente corrispondente, lasciando gli altri invariati. Il processo di trasformazione del dato inizia nel momento in cui il Dispatcher riceve un record; a quel punto lo analizza e sulla base delle configurazioni ottenute dal database Configurations lo invia alle code corrispondenti. Le code sono state divise in base a due dimensioni, alla tipologia di dato ed al fatto che i dati al suo interno siano già stati mappati o meno, quindi sarà possibile trovare due code per ogni tipologia di dato, una per i dati non ancora mappati e l'altra per quelli mappati. Inoltre, sono state aggiunte due code per mantenere i dati originali e quelli non supportati, rispettivamente chiamate Overall e Dumpster. Una volta ricevuto un record, il Dispatcher dovrà identificare il dispositivo con il quale è stato raccolto, se è supportato lo inoltrerà alle code Overall ed a quella di dati non mappati della tipologia corrispondente, mentre in caso contrario lo invierà alla coda Dumpster. I Mapper invece hanno il compito di ottenere i record non mappati dalla coda corrispondente alla loro tipologia, effettuare tutte le trasformazioni necessarie lette dal database delle configurazioni ed inoltrare il record risultante alla coda corrispondente di dati mappati, o alla coda Dumpster qualora si verificassero problemi nella trasformazione. Infine i Worker dovranno consumare i record dalle code di dati mappati, da Dumpster ed Overall, e memorizzarli nei database corrispondenti.

5.1.3 Spring

Spring è un framework che facilita lo sviluppo di applicazioni enterprise in Java riducendone la complessità, mettendo a disposizione una serie di strumenti per semplificare l'implementazione o fornire soluzioni alle problematiche più ricorrenti nella fase di realizzazione di software.

Spring segue il principio Inversion of control (IoC) invertendo il flusso di controllo del sistema, questo significa che sarà il framework ad occuparsi della creazione, inizializzazione, gestione e manipolazione degli oggetti sulla base di comandi e configurazioni impartiti dallo sviluppatore; una modalità con la quale viene realizzato IoC è il Dependency Injection, nel quale sono gli oggetti che definiscono le loro dipendenze. Sfruttando questo principio l'utente può usufruire in modo semplice dei design pattern codificati all'interno del framework, configurandolo adeguatamente; inoltre, dato che sarà Spring ad occuparsi di gestire l'intero ciclo di vita degli oggetti, lo sviluppatore potrà concentrarsi maggiormente sulla business logic del problema. Ad esso sono stati affiancati molti altri progetti ideati per fornire funzionalità aggiuntive al framework, come ad esempio Spring data e Spring batch; tra questi possiamo trovare anche Spring boot, che è stato utilizzato molto all'interno dello sviluppo prototipale di questo progetto. Spring boot ha lo scopo di semplificare l'esecuzione di applicazioni Spring e facilitarne la configurazione nella fase di bootstrap di un nuovo progetto, fornisce infatti una configurazione automatica standard basata sulle dipendenze che sono state aggiunte, e ne permette la modifica in caso di necessità per personalizzare l'ambiente di sviluppo.

Spring, ed in particolare Spring boot, sono stati utilizzati per lo sviluppo di tutti i servizi realizzati, tranne che per l'Activity provider che è stato realizzato in Python sfruttando il framework Flask per la gestione delle RESTful API; le funzionalità utilizzate riguardano principalmente l'integrazione di Apache Camel, che vedremo in seguito, di Kafka e RabbitMQ, sia direttamente che tramite il Framework Camel, del database Cloudant, ed infine per la realizzazione e per effettuare richieste a RESTful API.

5.1.4 Apache Camel

Camel è un framework che si pone l'obiettivo di semplificare l'integrazione tra i sistemi, fornendo astrazioni semplici e gestibili che permettono allo sviluppatore di concentrarsi maggiormente su come interagire con le varie parti, senza preoccuparsi di come esse funzionino. Esso consente di definire le proprie regole di instradamento, sarà quindi possibile decidere da quali sorgenti accettare i messaggi e verso quali destinatari dovranno essere indirizzati, inoltre permette di effettuare elaborazioni su di essi prima di inoltrarli; tra le caratteristiche più interessanti di Camel possiamo trovare il fatto che non necessita di conoscere il formato dei dati o di convertirli in un formato canonico per elaborarli, e che offre un livello elevato di astrazione che permette di interagire coi sistemi utilizzando la stessa API indipendentemente dal protocollo o dal tipo di dati utilizzati. Camel supporta una moltitudine di protocolli e tipi di dato, e la sua architettura estendibile e modulare permette di aggiungere nuovi protocolli qualora fosse necessario.

Nella fase di sviluppo prototipale della piattaforma, questo framework risulta estremamente utile per l'implementazione della maggior parte dei componenti lato cloud, in quanto ognuno di essi dovrà essere integrato con i servizi o le tecnologie dal quale riceverà informazioni ed al quale dovrà inoltrarle una volta effettuate alcune operazioni. Utilizzare Camel consente di facilitare l'integrazione con tutti i protocolli utilizzati nella piattaforma per l'interazione tra i componenti, in particolare il framework è stato sfruttato nei seguenti casi:

- nel Dispatcher è stata creata una regola che prevede di consumare messaggi dal broker Kafka, analizzarli e inoltrarli verso la coda RabbitMQ corrispondente;
- nei Mapper vengono consumati i record dalle code RabbitMQ di dati non mappati, vengono trasformati ed indirizzati nelle code di dati mappati;
- negli adapter per Validic e Google Fit è definita una regola che permette di effettuare periodicamente richieste alle RESTful API delle piattaforme cloud, elaborare le informazioni ricevute ed inoltrarle alle code RabbitMQ;
- nel Merge service per definire le RESTful API che devono essere esposte da questo componente ed il comportamento che dovrà scaturire dalla richiesta ad una di esse;

dovrà essere effettuata una richiesta ad entrambi i servizi Measurements provider ed Activities provider e le informazioni ottenute dovranno essere aggregate ed inoltrate in risposta alla chiamata ricevuta.

Per i servizi Worker, Measurements provider ed Activities provider non è stato utilizzato Camel in quanto il database IBM Cloudant non è nativamente supportato, necessita infatti dell'utilizzo del driver CData JDBC per funzionare; si è quindi ritenuto fosse più semplice realizzare tali componenti sfruttando solamente il framework Spring.

5.2 Deploy

Per la fase di deploy si è scelto di effettuarne una parte in locale ed una in cloud, sfruttando alcuni dei servizi offerti dalla piattaforma cloud di IBM; facendo riferimento alla figura 5.1, i componenti eseguiti in locale sono il Mock smartphone, il Dispatcher, i Mapper, Validic adapter, Google Fit adapter ed i Worker. Questi componenti sono stati eseguiti in locale, e non distribuiti utilizzando la piattaforma IBM, per i limiti imposti da essa; infatti, il deploy di servizi con il piano gratuito è vincolata ad un quantitativo di risorse ristretto e di conseguenza si è dovuto decidere quali servizi distribuire sulla piattaforma IBM e quali eseguire in locale.

I servizi che sono stati distribuiti utilizzando gli strumenti offerti dalla piattaforma cloud IBM sono Measurements provider, Activities provider e Merge service, per i quali si è sfruttato il Platform as a service (PAAS) Cloud foundry, che consente di eseguire il deploy di applicazioni e di scalarle a piacere. Inoltre, per la realizzazione di queste applicazioni si è scelto di adottare un approccio DevOps sfruttando il servizio IBM Continuous delivery, che permette di automatizzare i processi di build e di deploy.

I servizi della piattaforma IBM cloud che sono stati utilizzati sono i seguenti:

- IBM Event Streams, è un servizio di data streaming basato su Apache Kafka e nello sviluppo prototipale è stato utilizzato per permettere la comunicazione tra il simulatore dello smartphone ed il Dispatcher;
- CloudAMQP, è un servizio che permette di gestire un server RabbitMQ; nel momento di implementazione del prototipo è stato possibile ottenere tale servizio

dalla piattaforma IBM cloud, ma dal 4 dicembre 2020 non è più supportato da questa, quindi per utilizzare tale servizio sarà necessario registrarsi ed accedere presso CloudAMQP e creare una nuova istanza. Tale servizio è stato utilizzato per permettere la comunicazione tra i componenti situati negli strati di integrazione e dei microservizi, come precedentemente descritto;

- Cloudant, è un Database as a service (DAAS) NOSQL e nel prototipo è stato utilizzato per memorizzare le configurazioni ed i dati raccolti;
- API gateway, è un servizio che permette di creare, proteggere e gestire le API che fanno da tramite tra i servizi interni alla piattaforma e le richieste provenienti dall'esterno, funge da proxy di API e nello sviluppo prototipale è utilizzato come intermediario tra il servizio Merge service e l'esterno.

Infine vi è un altro servizio nella piattaforma IBM cloud, chiamato Cloud functions, che si ritiene utile anche se non è stato utilizzato; esso è un Function as a service (FAAS) che permette l'esecuzione di codice al presentarsi di una condizione, potrebbe essere utilizzato per la traduzione dei dati dal database NOSQL a quello SQL. Il motivo per il quale è stata individuata questa soluzione risiede nel fatto che tale servizio è facilmente integrabile con Cloudant, in particolare è possibile definire funzioni che eseguano codice ogni volta che vengono aggiornati i dati all'interno del database NOSQL.

Capitolo 6

Conclusioni

Il telemonitoraggio risulta essere importante per la medicina, infatti, la possibilità di monitorare i pazienti a distanza può aiutare i medici a prendere decisioni più consapevoli sullo stato di salute del paziente e può rendere più efficaci altre tipologie di telemedicina, come la televisita.

Attraverso questa tesi si è cercato di introdurre il tema della telemedicina evidenziandone l'importanza in un periodo come quello attuale; in particolare si è voluto approfondire il campo di applicazione del telemonitoraggio, fornendo una possibile architettura di riferimento per la realizzazione di piattaforme in grado di integrare un elevato numero di dispositivi sanitari, con lo scopo di raccoglierne i dati misurati e riutilizzarli per diverse tipologie di applicazioni. A tal fine, è stato condotto uno studio introduttivo alla telemedicina approfondendo il ramo del telemonitoraggio, cercando di esplorare le tecniche, le tecnologie e le architetture utilizzate nelle piattaforme sfruttate a tale scopo. In seguito sono stati analizzati ed elaborati i requisiti proposti, soffermandosi maggiormente sulle metodologie di raccolta dati e di realizzazione della piattaforma e, sulla base di queste operazioni e degli studi precedentemente effettuati, è stata progettata come soluzione una possibile architettura; su di essa è stato sviluppato un prototipo come Proof of Concept calato nel caso d'uso Covid-19, con lo scopo di validarne la progettazione e verificarne l'attuabilità.

Attraverso lo sviluppo prototipale si è voluto verificare principalmente il flusso dei dati, a partire dalla loro raccolta, passando alla fase di trasformazione per renderli con-

formi al modello della piattaforma, fino alla loro fruizione da parte di servizi interni o esterni ad essa. Nello sviluppo non è stato possibile verificare interamente l'architettura a causa del tempo insufficiente a disposizione, ci si è dovuti quindi limitare a simulare la raccolta dati dallo smartphone e la validazione delle API attraverso l'uso di Postman; questi aspetti sono comunque esposti nella sezione seguente che contenente gli sviluppi futuri proposti.

6.1 Sviluppi futuri

In seguito vengono illustrati gli sviluppi futuri e le parti sulle quali è necessario lavorare per terminare lo sviluppo dell'intera piattaforma:

- realizzare l'applicazione nativa Android come presentata in fase di analisi, facendo riferimento ai mockup in sezione 4.6.1 ed integrando la parte relativa a questionari e test, inoltre si consiglia di utilizzare alcuni dispositivi per verificarne il funzionamento;
- progettare e realizzare il componente edge utilizzando alcuni dispositivi per verificarne il funzionamento, in questo caso ci si può limitare a seguire l'analisi effettuata nella sezione 4.1.3 o esplorare una soluzione alternativa;
- sostituire la piattaforma Apache Kafka con una tecnologia alternativa tra quelle proposte nella sezione 5.1.1;
- implementare la parte di sicurezza in tutta la piattaforma seguendo quanto detto in sezione 4.4, modificando quindi i componenti già realizzati ed implementando il servizio che gestisce i dati dei pazienti;
- realizzare la parte che si occupa della traduzione dei dati per memorizzarli in un database relazionale;
- implementare l'interfaccia del medico come mostrato in fase di analisi e seguendo il mockup mostrato in sezione 4.6.2;

- realizzare il servizio che si occupa della gestione dei token per il meccanismo di autenticazione OAuth 2.0;
- implementare i servizi e le funzionalità necessarie ad esporre i dati, prendendo come riferimento Measurements provider e Activities provider;
- eseguire il deploy attraverso servizi cloud di tutta la piattaforma invece di eseguirne parte in locale come avviene attualmente.

Inoltre vi sono aspetti che non sono stati trattati nello sviluppo prototipale ma che potrebbe essere utile esplorare in futuro, tra questi possiamo trovare le varie tipologie di telemonitoraggio e di applicazioni che fanno uso dei dati raccolti; queste sono state solamente accennate nella tesi ma non sono state approfondite in quanto ritenute secondarie rispetto l'obiettivo del progetto. I tipi di telemonitoraggio non esplorati sono quello continuo e quello predittivo; per il primo è necessario progettare ed implementare il componente che si occupa della generazione degli allarmi, per il quale potrebbe essere pensata anche una soluzione che comprende il componente edge. Per la seconda tipologia invece è necessario esplorare le applicazioni di processamento ed analitica dei dati per ottenerne inferenza, in particolare si vogliono sviluppare algoritmi in grado di prevedere l'evoluzione della patologia.

Bibliography

- [1] Sonu Bhaskar et al. “Telemedicine Across the Globe-Position Paper From the COVID-19 Pandemic Health System Resilience PROGRAM (REPROGRAM) International Consortium (Part 1)”. In: *Frontiers in Public Health* 8 (2020), p. 644. ISSN: 2296-2565. DOI: 10.3389/fpubh.2020.556720. URL: <https://www.frontiersin.org/article/10.3389/fpubh.2020.556720>.
- [2] Swagata Biswas et al. “Smartphone-Based Telemedicine Service at Palliative Care Unit during Nationwide Lockdown: Our Initial Experience at a Tertiary Care Cancer Hospital”. In: *Indian Journal of Palliative Care* 26 (giu. 2020), p. 31. DOI: 10.4103/IJPC.IJPC_161_20.
- [3] Flavio Boraso et al. “Monografia Telemedicina - Mi curo a casa”. In: (2020), p. 61.
- [4] S. Durga, R. Nag e E. Daniel. “Survey on Machine Learning and Deep Learning Algorithms used in Internet of Things (IoT) Healthcare”. In: *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*. 2019, pp. 1018–1022. DOI: 10.1109/ICCMC.2019.8819806.
- [5] H. Habibzadeh et al. “A Survey of Healthcare Internet of Things (HIoT): A Clinical Perspective”. In: *IEEE Internet of Things Journal* 7.1 (2020), pp. 53–71. DOI: 10.1109/JIOT.2019.2946359.
- [6] WHO Group Consultation on Health Telematics. *A health telematics policy in support of WHO’s Health-for-all strategy for global health development : report of the WHO Group Consultation on Health Telematics, 11-16 December, Geneva, Switzerland, 1997*. 1998.

- [7] “IEEE Draft Standard for Service-Oriented Medical Device Exchange Architecture & Protocol Binding”. In: *IEEE P11073-20701/D5, July 2018* (2018), pp. 1–47.
- [8] Kenneth Lam et al. “Assessing Telemedicine Unreadiness Among Older Adults in the United States During the COVID-19 Pandemic”. In: *JAMA Internal Medicine* 180.10 (1 ott. 2020), pp. 1389–1391. ISSN: 2168-6106. DOI: 10.1001/jamainternmed.2020.2671. URL: <https://doi.org/10.1001/jamainternmed.2020.2671> (visitato il 02/07/2021).
- [9] Y. A. Qadri et al. “The Future of Healthcare Internet of Things: A Survey of Emerging Technologies”. In: *IEEE Communications Surveys Tutorials* 22.2 (2020), pp. 1121–1167. DOI: 10.1109/COMST.2020.2973314.
- [10] Andrew M Rizzi et al. “The new ‘normal’: Rapid adoption of telemedicine in orthopaedics during the COVID-19 pandemic”. In: *Injury* 51.12 (dic. 2020). Edition: 2020/09/16 Publisher: Elsevier, pp. 2816–2821. ISSN: 1879-0267. DOI: 10.1016/j.injury.2020.09.009. PMID: 32951916. URL: <https://pubmed.ncbi.nlm.nih.gov/32951916>.
- [11] Matteo Runfola et al. “Telemedicine Implementation on a Bariatric Outpatient Clinic During COVID-19 Pandemic in Italy: an Unexpected Hill-Start”. In: *Obesity surgery* 30.12 (dic. 2020). Edition: 2020/09/27 Publisher: Springer US, pp. 5145–5149. ISSN: 1708-0428. DOI: 10.1007/s11695-020-05007-z. PMID: 32981003. URL: <https://pubmed.ncbi.nlm.nih.gov/32981003>.
- [12] G. Saha, R. Singh e S. Saini. “A Survey Paper on the impact of “Internet of Things” in Healthcare”. In: *2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA)*. 2019, pp. 331–334. DOI: 10.1109/ICECA.2019.8822225.
- [13] Ministero della Salute (Italia). *TELEMEDICINA - Linee di indirizzo nazionali*. 2012.
- [14] M Suguna et al. “A Survey on Cloud and Internet of Things Based Healthcare Diagnosis”. In: dic. 2018, pp. 1–4. DOI: 10.1109/CCAA.2018.8777606.

-
- [15] *Telemonitoraggio dei pazienti Covid-19: l'esperienza dell'ASL Toscana Nord-Ovest.*
URL: <http://www.nbst.it/661-telemedicina-coronavirus-pazienti-covid-19-esperienza-asl-toscana-nord-ovest.html>. (accessed: 17.02.2020).