



SYSTEM DESIGN DOCUMENT

Nefrapp

Riferimento	Nefrapp_RAD_Vers.1.8
Versione	1.0
Data	05/12/2019
Destinatario	Professoressa F.Ferrucci
Proposto da	Eugenio Corbisiero, Sara Corrente, Silvio Di Martino, Antonio Donnarumma, Luca Esposito, Matteo Falco, Domenico Musone, Davide Benedetto Strianese.
Approvato da	Antonio Fasulo, Francesco Garofalo



Project Manager

Nome	Cognome	Matricola
Francesco	Garofalo	0522500615
Antonio	Fasulo	0522500627

Partecipanti

Nome	Cognome	Matricola
Eugenio	Corbisiero	0512105449
Sara	Corrente	0512105695
Silvio	Di Martino	0512105629
Antonio	Donnarumma	0512105083
Luca	Esposito	0512105123
Matteo	Falco	0512109201
Domenico	Musone	0512105689
Davide Benedetto	Strianese	0512105257



Revision History

Data	Versione	Descrizione	Autori
14/11/2019	0.1	Inizio stesura punti iniziali	Sara Corrente
19/11/2019	0.2	Mapping HW/SW	Antonio Donnarumma
22/11/2019	0.3	Design goals, trade-off	Domenico Musone
22/11/2019	0.4	Panoramica architettura del sistema proposto, decomposizione in Sottosistemi	Sara Corrente
22/11/2019	0.5	stesura dei punti 3.6 e 3.7 , con informazioni relative al controllo del flusso e le condizioni limite con relativa start-up , terminazione e fallimento	Eugenio Corbisiero
22/11/2019	0.6	Servizi dei Sottosistemi	Luca Esposito
24/11/2019	0.7	Aggiunta schema 3.4	Davide Benedetto Strianese
26/11/2019	0.8	Revisione e controllo qualità	Eugenio Corbisiero, Domenico Musone
29/11/2019	0.9	Correzioni post controllo qualità	Eugenio Corbisiero
05/12/2019	1.0	Revisione PM	Francesco Garofalo, Antonio Fasulo



Sommario

1.Introduzione	5
1.1 Obiettivi del sistema	5
1.2 Design Goals	5
Criteri di performance	5
Criteri di affidabilità	6
Criteri di manutenibilità	6
Criteri di usabilità	7
1.2.1 Trade-off	7
1.3 Definizioni, acronimi e abbreviazioni	8
1.4 Riferimenti	8
1.5 Panoramica	8
2. Architettura del Sistema Corrente	10
3. Architettura del Sistema Proposto	11
3.1 Panoramica	11
3.2 Decomposizione in Sottosistemi	11
3.2.1 Decomposizione in Layer	12
3.2.2 Schema Generale	13
3.3 Mapping hardware/software	13
3.4 Gestione dati persistenti	14
3.5 Controllo degli accessi e sicurezza	14
3.6 Controllo flusso globale del sistema	17
3.7 Condizioni limite	17
3.7.1 Start-up	17
3.7.2 Terminazione	17
3.7.3 Fallimento	17
4. Servizi dei Sottosistemi	18
5. Glossario	22



1.Introduzione

Con l'obiettivo di ottimizzare il servizio medico offerto dal reparto di Nefrologia dell'ospedale "Primo Policlinico Azienda Ospedaliera Universitaria Luigi Vanvitelli" di Napoli, si vuole semplificare la comunicazione e l'interazione medico/paziente per quanto concerne la dialisi peritoneale per i pazienti in dialisi presso la struttura.

1.1 Obiettivi del sistema

I medici del reparto di Nefrologia hanno problemi a comunicare con i loro pazienti, i quali vengono contattati su diversi social come Facebook o WhatsApp, o tramite sms, chiamata e altri mezzi di comunicazione. Questi mezzi di comunicazione che vengono attualmente utilizzati risultano inappropriati per la condivisione di foto dello stato dell'Exit Site.

Il problema principale è la necessità dei pazienti di monitorare giornalmente peso, pressione arteriosa, numero di scarichi al giorno, UF 24h e Diuresi 24h. I medici devono inserire mensilmente questi dati su un file Excel perdendo molto tempo a digitalizzarli e analizzarli.

L'obiettivo che si vuole raggiungere è la creazione di un'unica piattaforma centralizzata per la soluzione di problemi di comunicazione riguardanti sia i medici sia i pazienti.

1.2 Design Goals

Il processo di progettazione del sistema sarà guidato da rigorosi criteri di design, per favorire lo sviluppo di un prodotto finale che abbia le caratteristiche necessarie per inserirsi agevolmente nell'ambito di utilizzo preposto. I criteri di design scelti per il sistema saranno divisi nelle seguenti categorie: performance, affidabilità, manutenibilità, usabilità.

Criteri di performance

- **Tempo di risposta (priorità alta):** la risposta del sistema alle richieste degli utenti dovrà essere rapida, evitando di generare lunghe attese anche nel caso in cui giungessero al

sistema più richieste simultanee [in accordo con il Requisito non Funzionale **RNF_6: Prestazioni**].

Criteri di affidabilità

- **Sicurezza (priorità alta):** sarà garantito l'accesso al sistema soltanto agli utenti registrati, e i dati sensibili trattati dal sistema saranno sempre inaccessibili a parti non autorizzate. Una parte dei dati trattati sarà illeggibile anche per il personale tecnico del sistema [in accordo con i Requisiti non Funzionali **RNF_1: Sicurezza** e **RNF_8: Legali**].
- **Consistenza dei dati (priorità medio/alta):** nella gestione dei dati sottomessi dagli utenti, il sistema eviterà stati di errore e inconsistenze che potrebbero ostacolare il corretto svolgersi delle attività degli utenti e metterne a repentaglio la salute [in accordo con il Requisito non Funzionale **RNF_3: Affidabilità**].
- **Tolleranza agli errori (priorità media):** il sistema sarà in grado di continuare ad operare nel caso in cui si presentino condizioni di errore, tollerando eventuali malfunzionamenti e tornando tempestivamente in uno stato operativo [in accordo con il Requisito non Funzionale **RNF_3: Affidabilità**].

Criteri di manutenibilità

- **Tolleranza alle modifiche (priorità medio/bassa):** nel caso in cui il sistema dovesse necessitare di cambiamenti, esso potrà essere modificato nel codice sorgente, nei modelli di progettazione e nell'interezza della sua documentazione [in accordo con il Requisito non Funzionale **RNF_5: Manutenibilità**].
- **Testabilità (priorità medio/alta):** sarà possibile, in ogni fase dello sviluppo e della manutenzione del sistema, validarne la funzionalità e verificarne l'aderenza ai requisiti tramite opportuni test [in accordo con il Requisito non Funzionale **RNF_5: Manutenibilità**].
- **Portabilità (priorità medio/alta):** sarà possibile installare e utilizzare il sistema in diversi ambienti software ed hardware. Per esempio, saranno supportati diversi browser

web e sarà garantita l'accessibilità da dispositivi mobile. [in accordo con il Requisito non Funzionale **RNF_4: Supportabilità**].

- **Analizzabilità (priorità bassa):** sarà possibile effettuare operazioni di diagnostica per determinare le cause di errori e malfunzionamenti del sistema [in accordo con il Requisito non Funzionale **RNF_5: Manutenibilità**].

Criteri di usabilità

- **Comprensibilità (priorità medio/alta):** le funzionalità offerte agli utenti dal sistema potranno essere comprese facilmente e intuitivamente, e sarà disponibile opportuna documentazione per migliorare ulteriormente tale comprensione [in accordo con il Requisito non Funzionale **RNF_2: Usabilità**].
- **Facilità di apprendimento (priorità medio/alta):** gli utenti apprenderanno in tempi brevi le modalità di utilizzo delle funzionalità del sistema. [in accordo con il Requisito non Funzionale **RNF_2: Usabilità**].
- **Idoneità (priorità media):** il sistema fornirà agli utenti un insieme di funzionalità che sia aderente alla natura delle attività che essi dovranno svolgere e agli obiettivi che essi dovranno raggiungere [in accordo con il Requisito non Funzionale **RNF_2: Usabilità**].

1.2.1 Trade-off

- **Funzionalità vs. Usabilità:** si eviterà di arricchire eccessivamente il numero e la complessità delle funzionalità disponibili, in modo da consentire anche a utenti meno avvezzi all'uso della tecnologia di comprendere il funzionamento del sistema e padroneggiarne completamente l'uso.
- **Tolleranza agli errori vs. Consistenza dei dati:** nel caso in cui il sistema, o parte di esso, dovesse trovarsi in uno stato di errore, si preferirà interromperne il funzionamento laddove esista il rischio di corrompere dati sensibili continuandone l'esecuzione.
- **Usabilità vs. Sicurezza:** entrambi saranno obiettivi cruciali del processo di design, ma, vista la natura sensibile dei dati trattati dal sistema, sarà opportuno che la semplificazione dell'interfaccia utente non comporti tagli su controlli e protocolli di sicurezza.

1.3 Definizioni, acronimi e abbreviazioni

Acronimi	Descrizione
DBMS	Data Base Management System
GUI	Graphic User Interface
SDD	System Design Document
UF	Ultra Filtrato

1.4 Riferimenti

- Documentazione di progetto
- Bruegge, Dutoit, Object-Oriented Software Engineering.
- Nefrapp_RAD_Vers.1.8

1.5 Panoramica

- **Capitolo 1:**
Contiene una breve introduzione e descrizione degli obiettivi del sistema proposto, per poi passare ai design goals. In seguito vengono chiariti definizioni, acronimi ed abbreviazioni utilizzate all'interno del documento. Infine vi è una sezione di riferimenti utilizzati come linee guida per lo sviluppo del progetto.
- **Capitolo 2:**
Contiene la descrizione dello stato attuale dell'ambito di utilizzo in cui andrà ad inserirsi il sistema proposto.
- **Capitolo 3:**
Contiene la descrizione dell'architettura del sistema proposto. Questa sezione comprende una spiegazione riguardo la decomposizione in sottosistemi, il mapping hardware/software, la gestione dei dati persistenti, il controllo degli accessi e sicurezza, il controllo del flusso globale del sistema e le condizioni limite.
- **Capitolo 4:**
Contiene la definizione dei servizi offerti dai sottosistemi individuati nel Capitolo 3 e dettaglia l'interazione tra sottosistemi tramite i suddetti servizi.



- **Capitolo 5:**

Contiene la spiegazione dei termini tecnici utilizzati all'interno del documento.

2. Architettura del Sistema Corrente

Attualmente non è presente alcuna piattaforma adibita alla comunicazione tra medico e paziente. Ciò si traduce in comunicazioni che avvengono in maniera rudimentale e poco professionale, principalmente attraverso Social, ad esempio Facebook Messenger o WhatsApp, che rendono difficoltoso qualsiasi scambio di informazioni o richieste.

In particolare, l'attuale sistema risulta di difficile gestione per il medico che deve far fronte alle numerose richieste che gli pervengono su piattaforme che risultano inadatte alle interazioni medico-paziente (non è possibile, ad esempio, consultare i progressi di un paziente in un dato periodo in maniera pratica e veloce).

L'attuale sistema può talvolta risultare di difficile utilizzo per il paziente, non sufficientemente guidato nelle operazioni da svolgere (in particolare per quanto concerne l'invio dei dati al medico), e presenta evidenti problemi di archiviazione e sistemazione dei dati: risulta difficile, ad esempio, ritrovare eventuali trattamenti a cui il paziente deve sottoporsi oppure delle semplici brochure informative che il medico si impegna ad inviare periodicamente. È in questo ambito che, per cercare di risolvere queste problematiche, si inserisce il sistema proposto.

3. Architettura del Sistema Proposto

3.1 Panoramica

Il sistema avrà un'architettura Three-layer: un layer di interfaccia presenterà all'utente i servizi del layer applicativo del sistema, generando diverse interfacce in base alle sue necessità e ai suoi permessi d'accesso; il terzo e ultimo layer si occuperà di gestire i dati persistenti, rispondendo alle richieste di consultazione e aggiornamento del layer applicativo.

Sul server risiede un DBMS che si occuperà di recuperare, memorizzare ed interrogare i dati presenti nel database elaborando, quindi, le richieste degli utenti. L'aspetto degli accessi concorrenti al database sarà pertanto gestito dal DBMS stesso, che dovrà evitare eventuali colli di bottiglia. Tale architettura conferirà all'intero sistema una manutenibilità apprezzabile e permetterà di gestire il problema degli accessi concorrenti ai dati in maniera semplice ed efficace.

3.2 Decomposizione in Sottosistemi

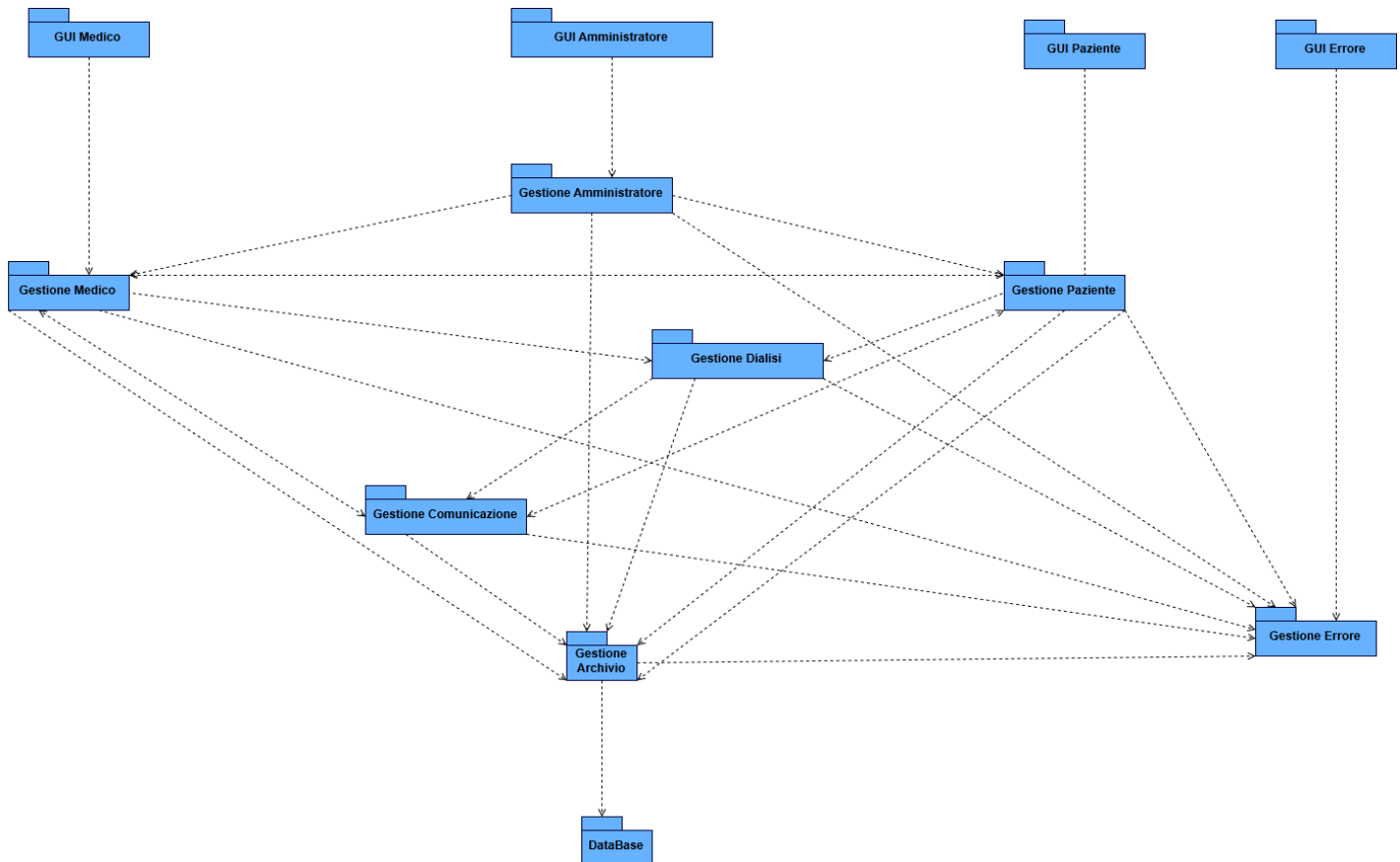
Dopo un'attenta analisi si è deciso di suddividere il sistema in sottosistemi in modo tale da gestire i singoli componenti con basso accoppiamento ed elevata coesione. Ciò permetterà in futuro di semplificare il processo di manutenzione dell'intero sistema.

La decomposizione in sottosistemi ha portato all'inserimento di un nuovo sottosistema chiamato "Gestione Errore". Questo sottosistema avrà il compito di gestire gli errori presenti all'interno del sistema.

Di seguito l'elenco dei sottosistemi e lo schema generale della suddivisione:

- Gestione Medico
- Gestione Paziente
- Gestione Amministratore
- Gestione Comunicazione
- Gestione Dialisi
- Gestione Errore
- GUI Medico
- GUI Paziente
- GUI Amministratore
- GUI Errore
- Gestione Archivio

- Database

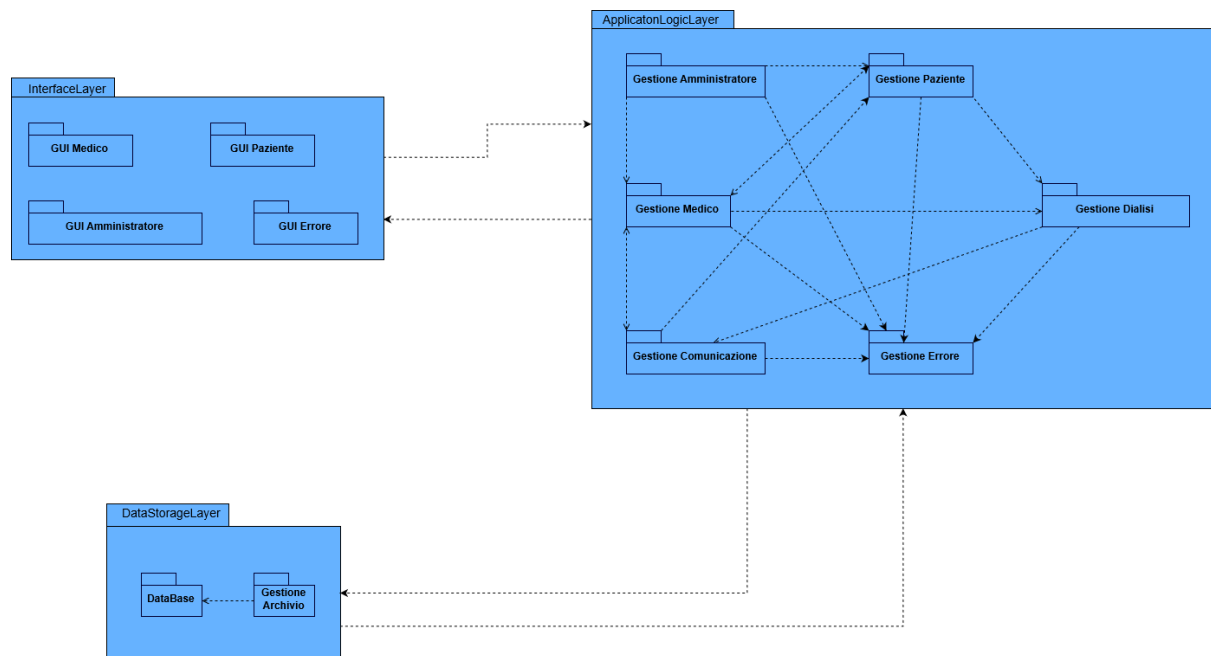


3.2.1 Decomposizione in Layer

- **InterfaceLayer:**
 - si occupa di gestire l'interfaccia dei diversi utenti (Medico, Paziente e Amministratore); include gli oggetti boundary con cui l'utente interagisce (finestre, form, liste, pulsanti, ecc.).
- **ApplicationLogicLayer:**
 - comprende le componenti logiche (oggetti control) responsabili del corretto funzionamento del sistema e interroga l'archivio per effettuare operazioni sui dati.
- **DataStorageLayer:**

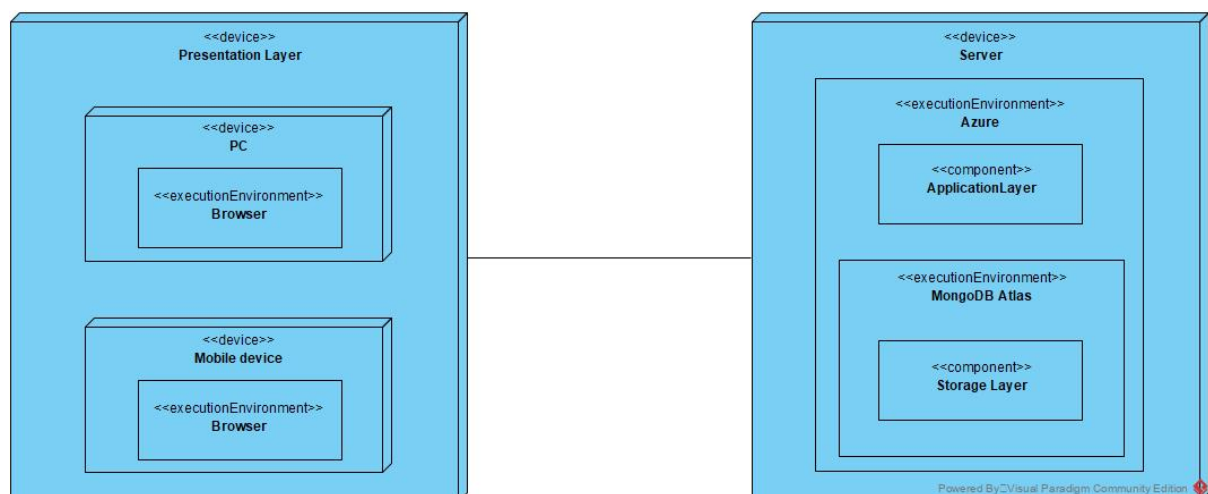
- comprende un gestore che si occupa di effettuare operazioni di inserimento, cancellazione e aggiornamento e si occupa di rendere disponibili i dati presenti all'interno dell'archivio.

3.2.2 Schema Generale



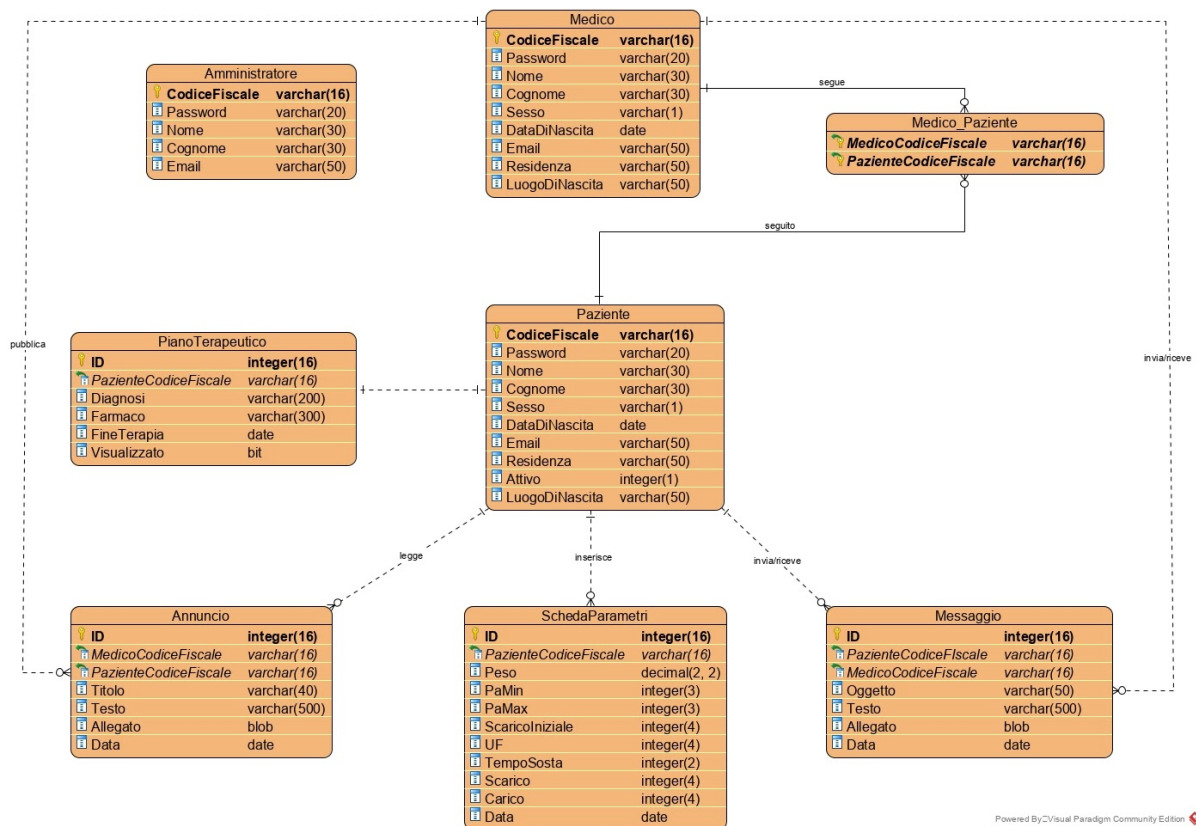
3.3 Mapping hardware/software

Il sistema sviluppato sarà installato su un server gestito da Microsoft Azure che manterrà sia la parte relativa alla logica dell'applicazione sia la parte relativa allo storage gestita attraverso MongoDB Atlas, un database NoSQL.



Il Deployment diagram mostra come sarà strutturata l'applicazione. Il paradigma utilizzato è quello tipico delle applicazioni WEB (client-server), mentre per quanto concerne l'architettura scelta sarà *two tier - three layer*. Dal lato client è possibile accedere (attraverso diversi dispositivi) all'applicazione, mentre dal lato server sarà possibile accedere alla logica dell'applicazione ed eventualmente allo storage.

3.4 Gestione dati persistenti



3.5 Controllo degli accessi e sicurezza

Il controllo degli accessi sarà alla base della sicurezza del nostro sistema, permettendo a medici, pazienti ed amministratori di collegarsi al sistema tramite l'utilizzo di codice fiscale e password, che verranno richieste ad ogni singolo accesso. La sessione terminerà quando l'utente effettua il logout. Nel caso in cui l'accesso non abbia successo, verrà notificato un errore nell'inserimento del codice fiscale o della password e sarà poi consentito all'utente di effettuare un nuovo tentativo. Il sistema fornirà più viste (interfacce grafiche) a seconda dell'attore che interagirà con esso, in modo che ognuno possa accedere solo a determinate funzionalità.



Per schematizzare al meglio il controllo degli accessi si è utilizzata una matrice degli accessi dove le righe rappresentano gli attori e le colonne rappresentano le classi. La cella relativa ad ogni coppia attore-classe contiene le operazioni consentite da quell'attore sulle istanze di quella classe.



Sottosistema ----- Attore	Gestione				
	Gestione utente	Gestione messaggi	Gestione annunci	Gestione parametri	Gestione piano terapeutico
Amministratore	<ul style="list-style-type: none"> • Login() • Logout() • Rimozione account medico() • Rimozione account paziente() • Visualizza lista pazienti() • Visualizza lista medici() • Modifica password() • Modifica account medico() • Modifica account paziente() • Registrazione medico() 				
Medico	<ul style="list-style-type: none"> • Login() • Logout() • Elimina account() • Reset password() • Modifica account() • Registra paziente da seguire() • Visualizza lista pazienti() 	<ul style="list-style-type: none"> • Invia messaggi() • Riceve messaggi() 	<ul style="list-style-type: none"> • Invia annunci() 	<ul style="list-style-type: none"> • Monitora parametri() • Download report() 	<ul style="list-style-type: none"> • Modifica piano terapeutico()
Paziente	<ul style="list-style-type: none"> • Login() • Logout() • Reset password() • Modifica account() • Disattiva account() 	<ul style="list-style-type: none"> • Invia messaggi() • Riceve messaggi() 	<ul style="list-style-type: none"> • Riceve annunci() 	<ul style="list-style-type: none"> • Inserisce parametri() • Visualizza parametri() 	<ul style="list-style-type: none"> • Visualizza piano terapeutico()

3.6 Controllo flusso globale del sistema

Nel flusso del sistema “Nefrapp”, il server sarà sempre in funzione (a meno di manutenzioni) in attesa di eventuali richieste da parte dell’utente, ragion per cui il controllo del flusso globale del sistema sarà di tipo event-driven, ovvero guidato da eventi.

3.7 Condizioni limite

Il sistema sarà sempre attivo permettendo l’utilizzo dei servizi in qualsiasi momento e, in presenza di un malfunzionamento, il sistema mostrerà una pagina di manutenzione. Nel caso si verifichi un’interruzione del sistema in seguito ad un errore, si tenterà un ripristino.

3.7.1 Start-up

Per il primo avvio del sistema “Nefrapp”, sarà necessario l’avvio di un web server che fornisca il servizio di un database MongoDB per la gestione dei dati persistenti e per l’interpretazione ed esecuzione del codice lato server. Prima dello start-up iniziale il database dovrà avere almeno un amministratore che una volta loggato può accedere a tutte le funzionalità a lui dedicate. A questo punto, medici e pazienti potranno essere registrati ed effettuare il login.

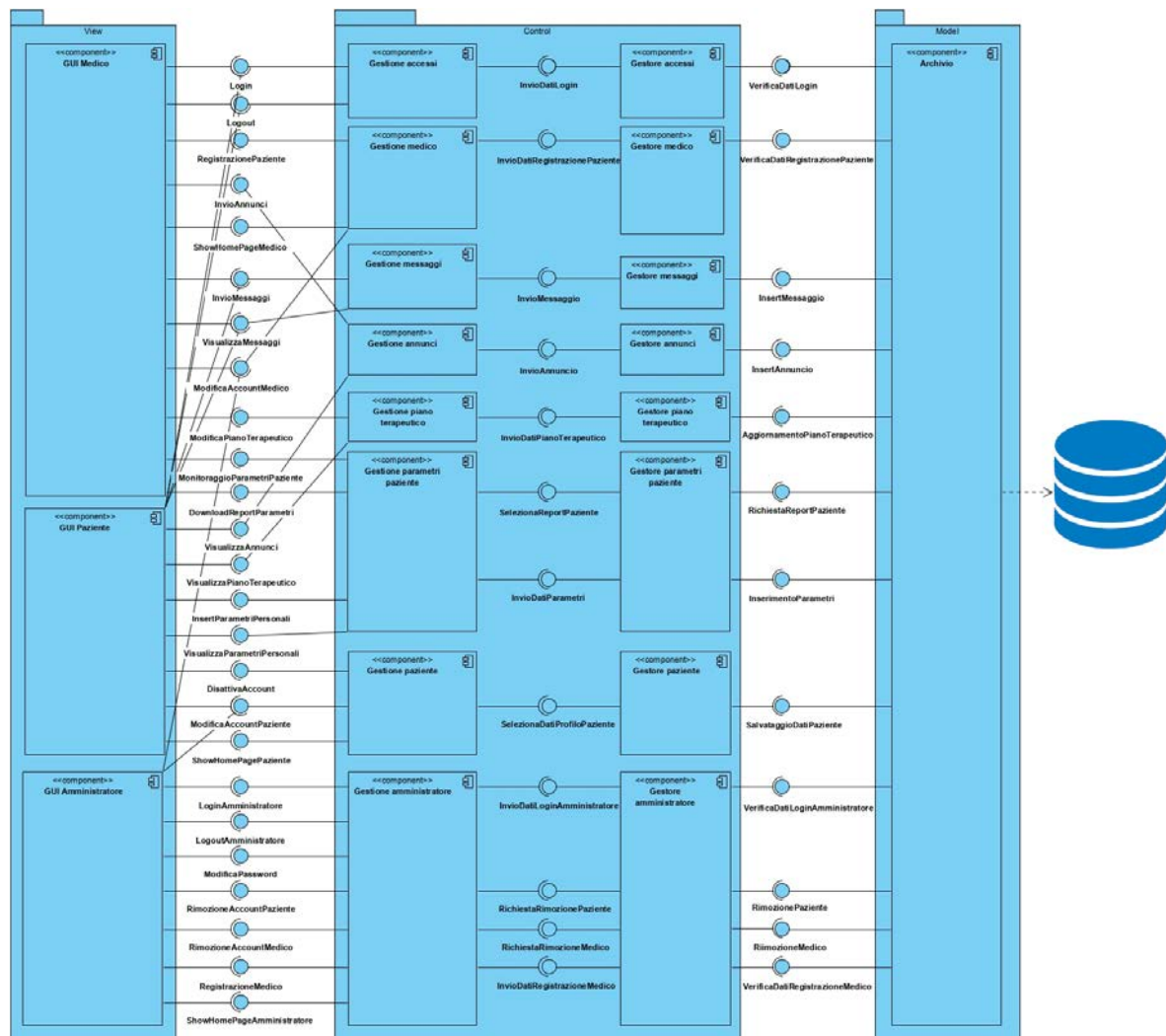
3.7.2 Terminazione

In caso di manutenzione il sistema sarà arrestato per un determinato quantitativo di ore. Dal punto di vista dell’utente, la terminazione avverrà nel caso in cui l’utente effettui il logout oppure nel caso in cui l’utente effettui la chiusura del web browser senza effettuare il logout.

3.7.3 Fallimento

Nel caso in cui si verifichi un’interruzione inaspettata dell’alimentazione non sono previsti metodi per ripristinare lo stato precedente del sistema. Un altro caso di fallimento potrebbe derivare dal software stesso che causa una chiusura inaspettata dovuta ad errori commessi durante la fase di implementazione. Non essendo previste politiche correttive, l’unica operazione consentita in questa particolare situazione è la visualizzazione di una semplice pagina di errore.

4. Servizi dei Sottosistemi



GUI Medico: Interfaccia che gestisce l'interazione del medico con il sistema.

GUI Paziente: Interfaccia che gestisce l'interazione del paziente con il sistema.

GUI Amministratore: Interfaccia che gestisce l'interazione dell'amministratore con il sistema.

Gestione medico: offre 3 servizi all'interfaccia GUI Medico:

- ModificaAccountMedico
- RegistrazionePaziente
- ShowHomePageMedico



e offre 1 servizio all'interfaccia GUI Amministratore:

- ModificaAccountMedico

Gestione accessi: offre 2 servizi all'interfaccia GUI Medico:

- Login
- Logout

e offre 2 servizi all'interfaccia GUI Paziente:

- Login
- Logout

Gestione messaggi: offre 2 servizi all'interfaccia GUI Medico:

- InvioMessaggi
- VisualizzaMessaggi

e offre 2 servizi all'interfaccia GUI Paziente:

- InvioMessaggi
- VisualizzaMessaggi

Gestione piano terapeutico: offre un unico servizio all'interfaccia GUI Medico:

- ModificaPianoTerapeutico

e offre un unico servizio all'interfaccia GUI Paziente:

- VisualizzaPianoTerapeutico

Gestione parametri paziente: offre 2 servizi all'interfaccia GUI Medico:

- MonitoraggioParametriPaziente
- DownloadReportParametri

e offre 2 servizi all'interfaccia GUI Paziente:

- VisualizzaPianoTerapeutico
- InsertParametriPersonalizzati

Gestione paziente: offre 3 servizi all'interfaccia GUI Paziente:

- DisattivaAccount
- ModificaAccountPaziente
- ShowHomePagePaziente



e offre 1 servizio all'interfaccia GUI Amministratore:

- ModificaAccountPaziente

Gestione amministratore: offre 7 servizi all'interfaccia GUI Amministratore:

- LoginAmministratore
- LogoutAmministratore
- ModificaPassword
- RimozioneAccountPaziente
- RimozioneAccountMedico
- RegistrazioneMedico
- ShowHomePageAmministratore

Gestore accessi: offre 1 servizio a Gestione accessi:

- InvioDatiLogin

Gestore medico: offre 1 servizio a Gestione medico:

- InvioDatiRegistrazionePaziente

Gestore messaggi: offre 1 servizio a Gestione messaggi:

- InvioMessaggio

Gestore annunci: offre 1 servizio a Gestione annunci:

- InvioAnnuncio

Gestore piano terapeutico: offre 1 servizio a Gestione piano terapeutico:

- InvioDatiPianoTerapeutico

Gestore parametri paziente: offre 2 servizi a Gestione parametri paziente:

- SelezionaReportPaziente
- InvioDatiParametri



Gestore paziente: offre 1 servizio a Gestione paziente:

- SelezionaDatiProfiloPaziente

Gestore amministratore: offre 4 servizi a Gestione amministratore:

- InvioDatiLoginAmministratore
- RichiestaRimozionePaziente
- RichiestaRimozioneMedico
- InvioDatiRegistrazioneMedico

Archivio: offre i servizi a Gestore accessi, Gestore medico, Gestore messaggi, Gestore annunci, Gestore piano terapeutico, Gestore parametri paziente, Gestore paziente, Gestore amministratore:

- VerificaDatiLogin
- VerificaDatiRegistrazionePaziente
- InsertMessaggio
- InsertAnnuncio
- AggiornamentoPianoTerapeutico
- RichiestaReportPaziente
- InserimentoParametri
- SalvataggioDatiPaziente
- VerificaDatiLoginAmministratore
- RimozionePaziente
- RimozioneMedico
- VerificaDatiRegistrazioneMedico

5. Glossario

- **Accoppiamento:** il grado con cui ciascuna componente di un sistema software dipende dalle altre componenti.
- **Coesione:** una misura di quanto strettamente correlate siano le varie funzionalità messe a disposizione da un singolo modulo di un sistema software.
- **DBMS:** software utilizzato per la gestione di un database.
- **Deployment diagram:** diagramma di tipo statico previsto dal linguaggio di modellazione object-oriented UML per descrivere un sistema in termini di risorse hardware, dette *nodi*, e di relazioni fra di esse.
- **Dialisi:** terapia fisica adottata per il trattamento dell'insufficienza renale cronica.
- **Diuresi:** processo di formazione delle urine nei reni.
- **Layer:** raggruppamento di sottosistemi che forniscono servizi correlati, eventualmente realizzati utilizzando servizi di altri layer.
- **Mapping hw/sw:** descrizione dei dispositivi e degli equipaggiamenti hardware e della loro interazione con le componenti software del sistema.
- **NoSQL:** movimento che promuove sistemi software dove la persistenza dei dati è in generale caratterizzata dal fatto di non utilizzare il modello relazionale, di solito usato dalle basi di dati tradizionali.
- **Sottosistema:** ciascun insieme strutturato di programmi che, nell'ambito di una rete di sistemi tra loro interconnessi, controlla un settore particolare delle risorse o delle funzioni del sistema più complesso.
- **Storage:** i dispositivi hardware, i supporti per la memorizzazione, le infrastrutture ed i software dedicati alla memorizzazione non volatile di grandi quantità di informazioni in formato elettronico.
- **Tier:** unità fisica di distribuzione delle funzionalità di un sistema software; un insieme di tier rappresenta la separazione fisica di presentazione, servizi e dati in apparati hardware differenti.
- **Ultrafiltrato:** parametro ottenuto mediante la differenza tra liquido “scaricato” e liquido “caricato”.



- **Web server:** applicazione software che, in esecuzione su un server, è in grado di gestire le richieste di trasferimento di pagine web di un client, tipicamente un web browser.