



Università degli Studi di Salerno

Corso di Ingegneria del Software



He♥lthyLife

HealthyLife

System Design Document

Versione 1.0

PARTECIPANTI AL PROGETTO

Nome	Matricola
Cataletti Raffaele	0512102372
D'Auria Antonio	0512102582
Nunziata Vincenzo	0512102594
Rotolo Westley	0512103310

Sommario

Sommario	2
1. Introduzione	3
1.1 Scopo del sistema	3
1.2 Design Goals	3
1.2.1 Uso del web	3
1.2.2 Open Source.....	4
1.2.3 Usabilità	4
1.2.4 Robustezza	4
1.2.5 Tempi di risposta.....	4
1.2.6 Sicurezza	4
1.2.7 Estendibilità	4
1.2.8 Tracciabilità dei requisiti	5
1.2.9 Memoria	5
1.3 Definizioni, Acronimi ed Abbreviazioni	5
1.4 Riferimenti.....	5
2 Sistema corrente.....	5
3. Architettura del Sistema	5
3.1 Overview.....	5
2 Decomposizione in sottosistemi	6
3.2.1 Presentation Layer	6
3.2.2 Logica applicativa	7
3.2.3 Dati Persistenti.....	8
3.3 Mapping HW/SW	8
3.3.1 Component Diagram	9
3.3.2 Deployment Diagram	10
3.4 Gestione dei dati persistenti	10
3.5 Controllo degli accessi e sicurezza.....	11
3.6 Controllo Software.....	13

3.7 Boundary conditions	13
3.7.1 Descrizione.....	13
3.7.2 Scenari e casi d'uso	14
4 Servizi dei sottosistemi	15
4.1 Accounting.....	15
4.2 Gestione Medico.....	15
4.3 Gestione Visite	15
4.4 Gestione Messaggi.....	16
4.5 Gestione Dieta	16

1. Introduzione

1.1 Scopo del sistema

HealthyLife vuole essere uno strumento che consente di migliorare l'efficienza della cura e la comunicazione tra paziente e dottore.

Esso offre strumenti e funzionalità che aiutano il paziente a migliorare le proprie abitudini ed il proprio stile di vita, oltre che a semplificare la comunicazione con il dottore e ad avere un andamento globale della propria cura.

Dal punto di vista del medico, il software permette di gestire al meglio i pazienti, dandogli la possibilità di aggiornare il menù, stabilire una visita, condividere materiale e suggerimenti utili per migliorare le abitudini dei propri pazienti, facilitare l'archiviazione dei risultati dell'andamento delle diete dei singoli utenti.

1.2 Design Goals

I design goals rappresentano le caratteristiche desiderate per l'applicazione e prevedono un insieme di criteri che devono essere considerati durante la creazione del sistema. Gli obiettivi vengono così identificati:

1.2.1 Uso del web

Per garantire un'esperienza ottimale del sistema, abbiamo deciso di utilizzare il web come piattaforma di comunicazione tra paziente e dottore. Per l'implementazione

useremo i seguenti linguaggi: HTML, PHP, CSS, JavaScript. Essi risultano traducibili ed eseguibili da qualunque dispositivo capace di collegarsi alla rete, quindi accessibili ovunque.

1.2.2 Open Source

Il servizio verrà fornito senza particolari licenze, non dovrà contenere componenti a pagamento.

1.2.3 Usabilità

L'uso del sistema deve risultare intuitivo. L'interfaccia grafica deve essere progettata in maniera tale che ogni tipo di utente possa facilmente capirne il funzionamento senza possedere particolari competenze informatiche.

1.2.4 Robustezza

HealthyLife presenta solo una piccola serie di interazioni tra l'utente ed il sistema, e deve essere in grado di riscontrare input errati da parte degli utenti comunicando in modo rapido e chiaro il problema verificatosi.

1.2.5 Tempi di risposta

I tempi di risposta dipenderanno in buona parte dalla qualità ed il tipo di connessione usata per interfacciarsi al sistema, ma in ogni caso dovranno essere previsti tempi ridotti. Nello specifico in condizioni di carico medio del sistema i tempi di risposta dovranno essere di 5 secondi medi e 10 secondi assoluti.

1.2.6 Sicurezza

HealthyLife prevede un sistema di autenticazione tramite username e password. Grazie a questo sistema ogni utente ha l'accesso soltanto alle funzionalità a lui offerte.

1.2.7 Estendibilità

Il sistema verrà sviluppato in modo modulare, di conseguenza esso permette l'aggiunta di nuove funzionalità in futuro.

1.2.8 Tracciabilità dei requisiti

La tracciabilità dei requisiti risulta relativamente semplice grazie alla coerenza tra gli artefatti realizzati, sia per quanto riguarda la terminologia sia per i concetti contenuti. Effettuando la tracciabilità dei requisiti sarà possibile ridurre i costi e i rischi, valutando l'impatto delle modifiche effettuate.

1.2.9 Memoria

Il sistema non necessiterà di grossi quantitativi di memoria, in quanto non si dovranno gestire file di grossa taglia, e l'utenza non sarà molto grande. Tutti gli utenti, divisi per le rispettive categorie, apparterranno ad un DBMS

1.3 Definizioni, Acronimi ed Abbreviazioni

TERMINE	DESCRIZIONE
HTML	HyperText Markup Language
PHP	PHP: HyperText Preprocessor
JavaScript	Linguaggio di scripting orientato agli oggetti e agli eventi
Css	Cascading Style Sheets
RAD	Requirements Analysis Document
DBMS	Data Base Management System

1.4 Riferimenti

RAD – HealthyLife Requirements Analysis Document

2 Sistema corrente

Il sistema deve ancora essere implementato e rilasciato. Non esistono altre versioni.

3. Architettura del Sistema

3.1 Overview

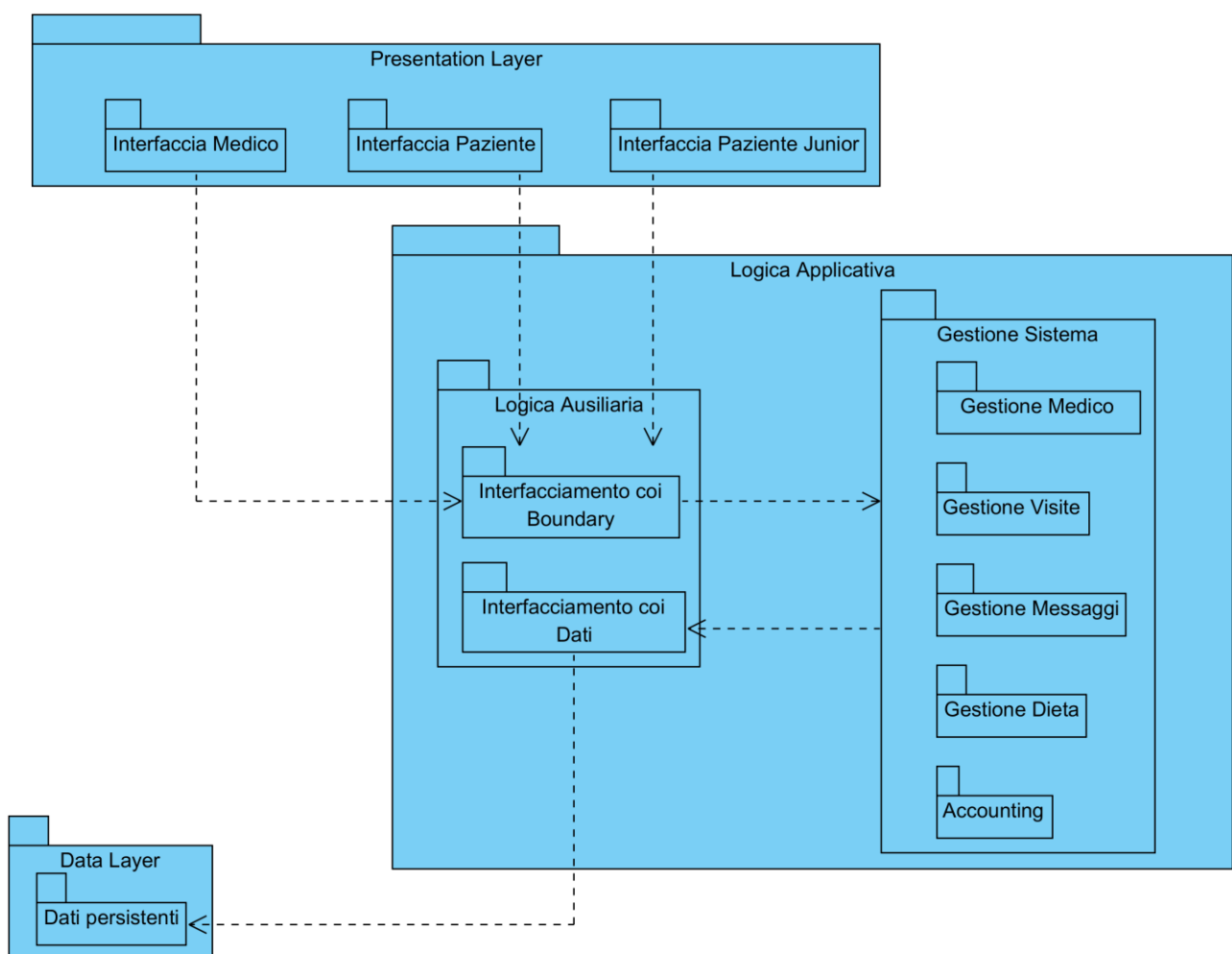
Il sistema sarà sviluppato secondo un'architettura Three-Tier: livello di presentazione, livello applicativo e livello dei dati.

Il livello di presentazione è composto da diverse interfacce grafiche, divise in base agli utenti del sistema:

- Paziente
- Paziente Junior
- Medico

Il livello di gestione della persistenza è costituito da un DBMS, che si occupa della conservazione di dati e metadati propri del sistema.

2 Decomposizione in sottosistemi



3.2.1 Presentation Layer

3.2.1.1 Interfaccia Medico

Permette al medico di essere autenticato e avvia l'accesso alle funzionalità del sistema attraverso un menu descritto nel punto [3.4.5.14] del Rad.

3.2.1.2 Interfaccia Paziente

Permette al paziente di essere autenticato e avvia l'accesso alle funzionalità del sistema attraverso un menu descritto nel punto [3.4.5.4] del Rad.

3.2.1.3 Interfaccia Paziente Junior

Permette al paziente Junior di essere autenticato e avvia l'accesso alle funzionalità del sistema attraverso un menu descritto nel punto [3.4.5.12] del Rad.

3.2.2 Logica applicativa

3.2.2.1 Accounting

Orchestra l'associazione di Account utente ai dati del relativo utente, occupandosi, quindi, dell'autenticazione degli utenti e della garanzia dei privilegi dovuti. Si occupa, inoltre, della gestione di tutte le utenze registrate nel sistema, permettendone l'aggiunta, la modifica e l'eliminazione.

3.2.2.2 Gestione Medico

Si occupa di gestire le correlazioni tra medici e pazienti. Permette ad un paziente di richiedere l'associazione ad un medico ed al medico di accettare tale richiesta e rendere quindi effettiva tale associazione. Un paziente infine può anche cambiare medico inviando una richiesta ad un nuovo tale.

3.2.2.3 Gestione Visite

Permette la gestione delle visite di un medico. Rende possibile l'aggiunta di nuove visite, o la modifica/eliminazione di visite già esistenti. Inoltre notifica tramite un avviso importante il paziente della creazione/modifica della visita indicandogli data ed ora dell'appuntamento.

3.2.2.4 Gestione Avvisi e messaggi

Permette di gestire l'invio e la ricezione di messaggi ed avvisi.

3.2.2.5 Gestione Dieta

Consente di creare nuovi menù per nuovi pazienti o anche di modificare quelli già esistenti. Gestisce due tipi diversi di modifiche, settimanale e giornaliera, la prima riguardo il menù riproposto settimana per settimana, la seconda il menù proposto invece in un singolo giorno. Inoltre consente di gestire il peso di un paziente.

3.2.2.6 Interfacciamento con i boundary

Sistema di inoltro delle richieste ai control da parte dei boundary. Ha anche una controparte tramite la quale i control istanziano i nuovi boundary da mandare in display per restituire il feedback delle operazioni.

3.2.2.7 Interfacciamento con i dati

Sistema che si occupa di accedere ai dati memorizzati all'interno dei dati persistenti (DBMS), e di costruire le informazioni da utilizzare in tutti gli altri sottosistemi.

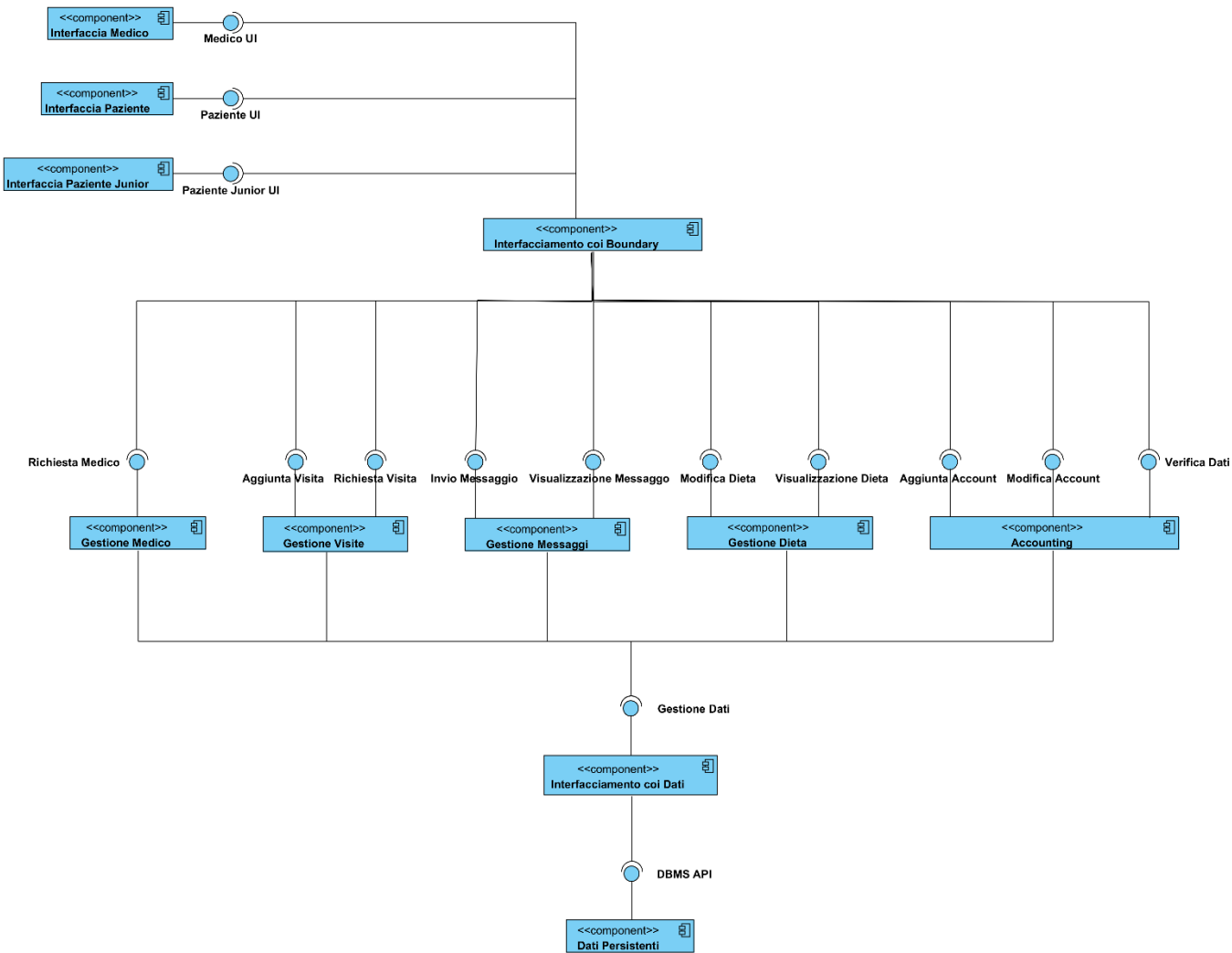
3.2.3 Dati Persistenti

E' responsabile di mantenere i dati persistenti e di assicurare la coerenza dei dati attraverso il controllo dei riferimenti e l'atomicità delle operazioni.

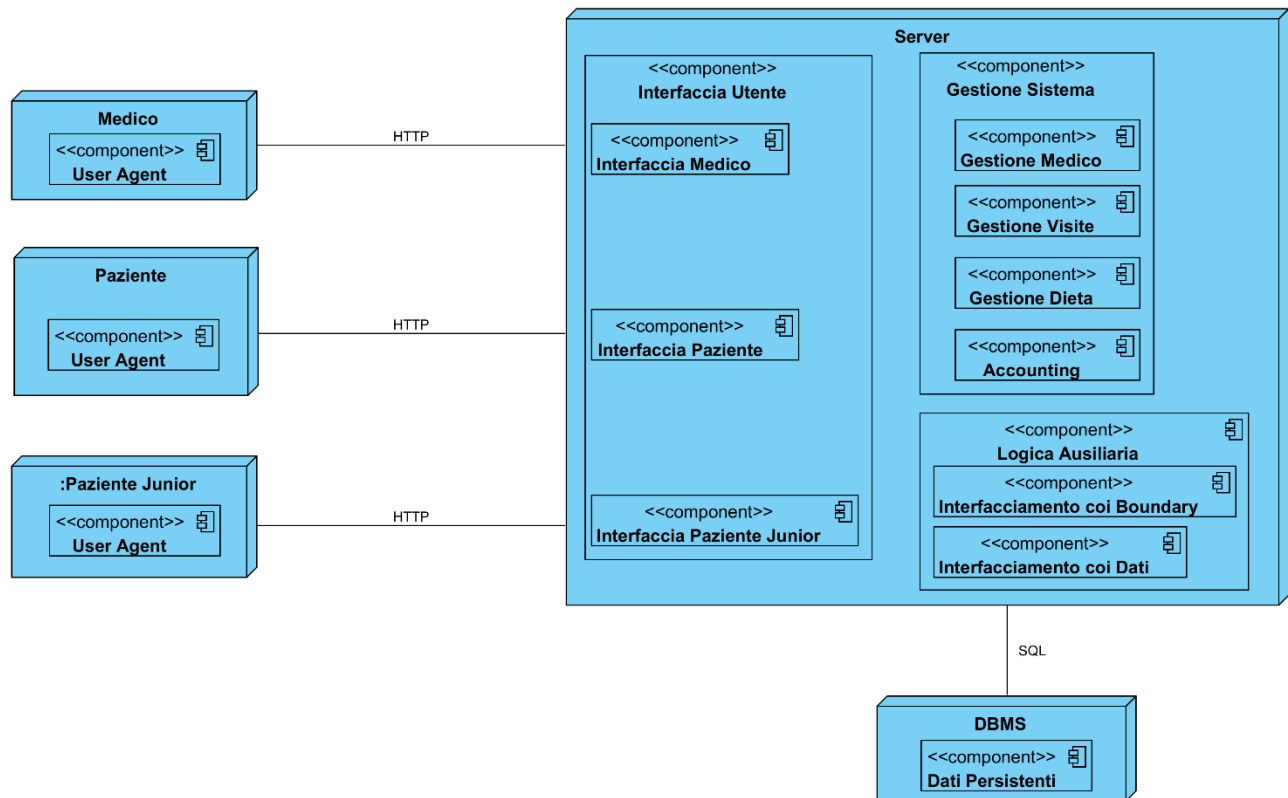
3.3 Mapping HW/SW

Il sistema HealthyLife sarà web-based per tutti gli utenti. Le funzionalità verranno quindi implementare sul server e accedute via HTTP dagli utenti.

3.3.1 Component Diagram



3.3.2 Deployment Diagram



3.4 Gestione dei dati persistenti

Analizzando il Requirements Analysis Document, sono stati identificati i seguenti oggetti persistenti:

Entità	Definizione
Medico	Persona che offre il servizio ai vari pazienti. Dopo averli accettati nell'apposita sezione, può seguire la cura di un numero limitato di Pazienti aggiornando il menù, pianificando visite, mandando messaggi di avviso o di consiglio ecc.
Paziente	Persona che ha bisogno di seguire una cura presso un medico. Può cercare un medico nella zona ed inviargli una richiesta di visita, una volta effettuata tale visita, il paziente potrà controllare/aggiornare il proprio peso, leggere eventuali consigli/avvisi inviatigli dal medico, cercare punti d'interesse, visualizzare l'andamento della cura e nel caso

	in cui il paziente sia Junior avrà anche la possibilità di controllare lo stato del supereroe.
Menù	Ogni paziente ne ha uno, contiene indicazioni riguardanti cosa può e cosa non può mangiare quella persona giorno per giorno.
Messaggio	Messaggio contenente informazioni importanti, inviato da un medico ad un paziente. Può essere di due tipi: <ul style="list-style-type: none">• Avviso importante• Consiglio utile
Visita	La visita si tiene in un determinato giorno ed una determinata ora presso il proprio Medico curante. Viene creata dal Medico per sostenere la cura del proprio paziente.
Supereroe	Il Supereroe ha un suo stato che può essere aggiornato solo ed esclusivamente dal medico curante. Un paziente Junior ha la possibilità di visualizzare lo stato del Supereroe.

3.5 Controllo degli accessi e sicurezza

Il sistema prevede un sistema di autenticazione mediante l'inserimento di coppie email-password per l'accesso alle funzionalità offerte a ogni attore.

Il sistema prevede l'utilizzo di un sistema di tipo Access Control List per la verifica dei privilegi previsti per ciascun attore: per ogni oggetto sono presenti delle coppie (attore, operazione) che permettono di stabilire se l'attore può eseguire una determinata operazione.

Paziente

(Utente Guest, Registrazione)

(Paziente, Visualizzazione Dati Personali)

(Medico, Visualizzazione Dati)

(Paziente, Modifica dati personali)

Paziente Junior

(Utente Guest, Registrazione)

(Paziente Junior, Visualizzazione dati personali)

(Paziente Junior, Modifica dati personali)

Medico

(Utente Guest, Registrazione)

(Medico, Visualizzazione dati personali)

(Medico, Modifica dati personali)

Comunicazione

(Medico, Invio)

(Paziente, Visualizzazione)

(Medico, Visualizzazione)

Visita

(Paziente, Invio Richiesta)

(Medico, Accettazione Richiesta)

(Medico, Aggiunta Visita)

(Medico, Rimozione Visita)

Menu

(Medico, Aggiunta Menu)

(Medico, Modifica Menu)

(Paziente, Visualizzazione Menu)

Peso

(Paziente, Aggiunta Peso)

(Medico, Visualizzazione Peso)

Supereroe

(Medico, Modifica)

(Paziente Junior, Visualizzazione)

3.6 Controllo Software

Il sistema HealthyLife sarà realizzato con un controllo esplicito con paradigma procedurale. Il flusso di controllo sarà gestito nella maniera più consona ad un'applicazione web, quindi sarà di tipo event-driven. La gestione della concorrenza inoltre è un problema corposo da gestire in un approccio di questa tipologia a causa dell'imprevedibilità degli eventi provenienti dai numerosi utenti con cui il sistema si troverà a dialogare: l'integrità dei dati è tuttavia garantita dal DBMS MySQL, tramite la semantica delle transazioni.

3.7 Boundary conditions

3.7.1 Descrizione

Le operazioni di start, stop ed installazione dell'applicazione HealthyLife definiscono le boundary conditions:

3.7.1.1 Installazione

HealthyLife non richiede una installazione esplicita, e funziona su tutti i sistemi operativi. Lato server è richiesta la presenza del DBMS e del web server. In tale fase, inoltre deve essere gestito(manualmente) il file di configurazione riguardante l'inserimento dell'Amministratore Globale. Lato client è richiesta la presenza di connessione ad Internet e di un Browser compatibile con HTML e Javascript. Il servizio è interamente usufruibile sul web.

3.7.1.2 Configurazione

Si necessita inserire i file del sistema HealthyLife all'interno di una directory raggiungibile tramite il Web Server attivo. Successivamente si inserirà un file di configurazione per l'amministratore del sistema secondo un template fornito nel pacchetto di installazione. Una volta configurato il file, si dovrà effettuare il primo avvio per configurare i nuovi dati dell'account dell'amministratore del sistema; infatti il sistema caricherà i dati dell'amministratore generale nel DBMS in forma protetta e rimuoverà il file per garantire la sicurezza

3.7.1.3 Avvio

Lato server, l'avvio richiede l'avvio dello script di monitoraggio del trasporto, e, poi, del web server.

3.7.1.4 Arresto

Lato server, l'arresto richiede lo stop del web server, e quindi dello script di monitoraggio del trasporto

3.7.2 Scenari e casi d'uso

3.7.2.1 Configurazione del Server

Nome	Configurazione del Server
Partecipanti	Amministratore Generale
Precondizioni	Il sistema non è ancora configurato per il primo avvio
Flusso di eventi	<ol style="list-style-type: none">1. Amministratore Generale: Copia i file del sistema HealthyLife in una directory raggiungibile dal Web Server;2. Amministratore Generale: Configura il file di configurazione dell'Amministratore Generale3. Amministratore Generale: Avvia per la prima volta il server. <<include Startup del Server >>4. Sistema: Legge i dati dal file di configurazione e li salva nel DB5. Sistema: Genera una username ed una password per l'Amministratore Generale e crea l'account.6. Sistema: Invia una mail all'Amministratore Generale per comunicargli i dati circa il suo account.7. Sistema: Cancella il file di configurazione.
Postcondizioni	Il Sistema può essere avviato

3.7.2.2 Avvio del Server

Nome	Avvio del Server
Partecipanti	Amministratore Generale
Precondizioni	Il server non è stato ancora avviato
Flusso di eventi	<ol style="list-style-type: none">1. Amministratore Generale: Avvia MySql.2. Amministratore Generale: Avvia lo script di monitoraggio trasporti.3. Amministratore Generale: Avvia il WebServer.
Postcondizioni	Il Server è avviato

3.7.2.3 Arresto del Server

Nome	Arresto del Server
Partecipanti	Amministratore Generale
Precondizioni	Il Server è attivo

Flusso di eventi	1. Amministratore Generale: Stoppa il Web Server. 2. Amministratore Generale: Stoppa lo script di monitoraggio trasporti. 3. Amministratore Generale: Stoppa il servizio MySql.
Postcondizioni	Il Server non è più attivo

4 Servizi dei sottosistemi

4.1 Accounting

Operazione	Interfaccia	Descrizione
Registrazione	Aggiunta	Permette ad un Utente di registrarsi ad HealthyLife
Login	Verifica dati	Permette all'utente di loggarsi al sistema
Modifica dati personali	Modifica	Permette all'utente di modificare i proprio dati personali

4.2 Gestione Medico

Operazione	Interfaccia	Descrizione
Invio richiesta prima visita ad un medico	Richiesta medico	Permette ad un paziente di richiedere la prima visita ad un medico
Visualizzazione delle richieste di visita di un medico	Richiesta medico	Permette ad un medico di visualizzare tutte le richieste di visita ricevute
Accettazione/rifiuto delle richieste di visita	Richiesta medico	Permette ad un medico di accettare/rifiutare una richiesta di visita

4.3 Gestione Visite

Operazione	Interfaccia	Descrizione
Invio richiesta Visita	Richiesta visita	Permette ad un paziente di richiedere una visita al proprio medico

Aggiungi Visita	Aggiunta visita	Permette al medico di aggiungere una visita per un determinato paziente
Accetta richiesta di visita	Aggiunta visita	Permette al medico di accettare ed aggiungere una visita ad un paziente

4.4 Gestione Messaggi

Operazione	Interfaccia	Descrizione
Invio Messaggio	Invio Messaggio	Permette al medico di inviare un messaggio ad un paziente
Lettura Messaggio	Visualizzazione Messaggio	Permette al paziente di leggere un messaggio a lui destinato
Lista Messaggi	Visualizzazione Messaggio	Permette al paziente di visualizzare la lista di tutti i messaggi ricevuti

4.5 Gestione Dieta

Operazione	Interfaccia	Descrizione
Aggiungi/Modifica Menù	Modifica dieta	Permette al medico di modificare il menù di un paziente
Aggiorna Stato del Supereroe	Modifica dieta	Permette al medico di modificare lo stato di un Supereroe
Visualizza andamento del peso del Paziente	Visualizza dieta	Permette sia al medico che al paziente di visualizzare l'andamento del peso
Visualizza Menù	Visualizza dieta	Permette sia al medico che al paziente di visualizzare il menù
Aggiorna Peso	Modifica dieta	Permette al paziente di aggiornare il proprio peso giornaliero