



Università degli Studi di Salerno – Dipartimento di Informatica  
Ingegneria del Software – Prof. A. De Lucia

---



SDD

**System Design Document**

---

# Partecipanti

Cognome	Nome	Matricola
Campofreda	Alessio	05121 05492
Caccia	Raffaele	05121 05514
Pincivalle	Rolando Antonio	05121 05192
Spera	Francesco	05121 05408

---

# Indice

1. INTRODUZIONE .....	4
1.1 OBIETTIVO DEL SISTEMA .....	3
1.2 DESIGN GOALS .....	4
1.3 DEFINIZIONI, ABBREVIAZIONE E ACRONIMI .....	5
1.4 RIFERIMENTI .....	5
1.5 PANORAMICA .....	5
2. ARCHITETTURA DEL SISTEMA CORRENTE .....	6
3. ARCHITETTURA DEL SISTEMA PROPOSTO .....	6
3.ARCHITETTURA DEL SISTEMA PROPOSTO	
3.1 DECOMPOSIZIONE DEL SISTEMA IN SOTTOSISTEMI .....	7
3.2 MAPPING HARDWARE/SOFTWARE .....	9
3.3 GESTIONE DEI DATI PERSISTENTI .....	9
3.4 CONTROLLO DEGLI ACCESSI E DELLA SICUREZZA .....	10
3.5 CONTROLLO DEL FLUSSO GLOBALE DEL SISTEMA .....	10
3.6 CONDIZIONI BOUNDARY .....	11
4. SERVIZIO DEI SOTTOSISTEMI .....	13

---

# 1. INTRODUZIONE

## 1.1 OBIETTIVO DEL SISTEMA

Lo scopo del sistema è quello di facilitare l'interazione fra paziente e medico digitalizzandone le operazioni.

Il sistema, da parte del paziente, si riserva di sostituire la cartella clinica cartacea in una totalmente digitale

creando un rapido accesso ad essa; da parte del medico, si riserva di digitalizzare la propria agenda.

Il cliente potrà scegliere il medico che più gli aggrada collegandosi ad esso, potrà richiedere appuntamenti e ricette. Dall'altra parte, il medico, con l'agenda digitale, avrà la possibilità di gestire al meglio gli appuntamenti tenendo conto di eventuali impegni esterni. Il medico potrà ottimizzare al meglio l'uso della carta poiché il materiale da lui prodotto sarà possibile caricarlo in formato pdf.

## 1.2 DESIGN GOALS

Usabilità:

l'interfaccia del sistema sarà intuitiva in modo da renderne l'utilizzo semplice e veloce, l'utente verrà guidato in tutte le azioni che eseguirà non generando dubbi o incertezze. Le interfacce del sistema saranno gestite in modo tale che l'utente percepirà il cambiamento successivo all'azione.

Tempi di risposta:

il sistema sarà progettato garantendo una rapida risposta (in un tempo non superiore ai 3 secondi) e di supportare un elevato carico di utenza (almeno 1000 utenti).

Sicurezza:

il sistema utilizzerà dei protocolli sicuri e controllati in modo da garantire sicurezza all'utente. I parametri sensibili come la password verranno criptati.

Manutenibilità:

il sistema risulterà semplice da mantenere per l'implementazione di facile comprensione. Risulteranno ridotti i periodi di discontinuità.

Affidabilità:

il sistema dovrà garantire il successo dell'operazione effettuata dall'utente, in caso di errore sarà guidato attraverso messaggi che riporteranno l'errore commesso.

---

Disponibilità:

il sistema dovrà essere disponibile h24 in modo da permettere agli utenti di usufruirne costantemente.

## 1.3 DEFINIZIONI, ABBREVIAZIONI E ACRONIMI

SDD: System Design Document

MVC: Model-View-Controller

JSP: Java Servlet Page

DBMS: Database Management System

## 1.4 RIFERIMENTI

Si rimanda ai documenti "RAD\_MEDASSISTANT" e "SDD\_DATIPERSISTENTI\_MEDASSISTANT"

## 1.5 PANORAMICA

Il seguente SDD mostra i dettagli tecnici del sistema MedAssistant.

Il documento si compone di 4 sezioni:

1. Introduzione:

viene descritto il sistema proposto, descrivendo il motivo per cui è stato pensato e le funzionalità;

2. Architettura del sistema corrente:

viene descritto il sistema che si sta utilizzando correntemente;

3. Architettura del sistema proposto:

viene descritto l'architettura del sistema, in dettaglio:

3.1 Decomposizione del sistema in sottosistemi;

3.2 Mapping Hardware/Software;

3.3 Gestione dei dati persistenti;

3.4 Controllo degli accessi e della sicurezza;

3.5 Controllo del flusso globale del sistema;

3.6 Condizioni boundary;

4. Servizi dei sottosistemi:

vengono descritti i sottosistemi individuati indicando i servizi offerti.

---

## 2. ARCHITETTURA DEL SISTEMA CORRENTE

Attualmente non esiste un sistema software.

## 3. ARCHITETTURA DEL SISTEMA PROPOSTO

La sezione riguardante l'architettura del sistema proposto è così divisa:

### Decomposizione del sistema in sottosistemi

Si suddivide il sistema in sottosistemi per diminuire la difficoltà, permette lo sviluppo individuale dei vari sottosistemi in maniera più semplice;

### Mapping Hardware/Software

Si individua quale configurazione software verrà impiegata; verranno definiti i nodi responsabili delle funzionalità; la comunicazione tra questi; l'eventuale utilizzo di tecnologie già esistenti e come queste vengono incapsulate.

### Gestione dei dati persistenti

Si individuano le informazioni da persistere e come accedervi;

### Controllo degli accessi e della sicurezza

Si identificano gli attori che possono accedere alle informazioni, come è realizzato e specificato l'accesso e se quest'ultimo può cambiare;

### Controllo del flusso globale del Sistema

Si definisce il flusso di operazioni del sistema e come il sistema gestisce le interazioni con l'utente;

### Condizioni boundary

Si descrive come il sistema viene avviato, arrestato e come il sistema gestisce i casi di errore.

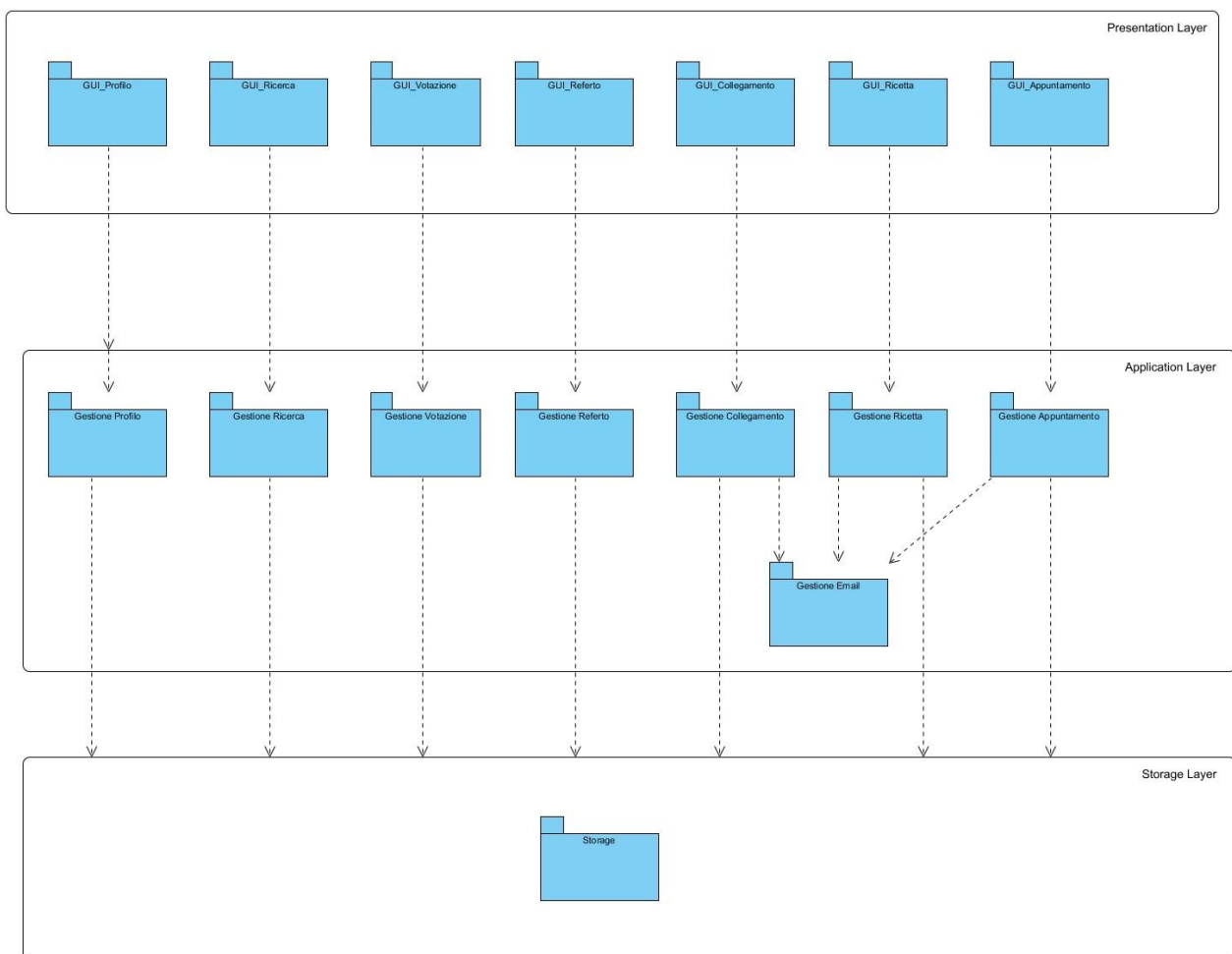
## 3.1 DECOMPOSIZIONE DEL SISTEMA IN SOTTOSISTEMI

L'architettura scelta per questo sistema è MVC strutturata in tre componenti:

- il Model che si occupa di gestire i dati persistenti;
- la View che si occupa dell'interazione tra utente ed il sistema;
- il Controller che si occupa della logica di controllo dell'applicazione interagendo con gli altri componenti.

L'architettura MVC verrà implementata con la tecnologia Servlet-JSP.

### *Decomposizione in sottosistemi*



---

GUI_Profilo	È responsabile dell'interazione con l'utente per la gestione del profilo utente.
GestioneProfilo	Permette di effettuare l'autenticazione, logout, registrarsi e modificare il profilo.
GUI_Ricerca	È responsabile dell'interazione con l'utente per la gestione della ricerca di medici e pazienti.
GestioneRicerca	Permette di cercare medici in base alla zona, tipo e nome; un paziente per nome.
GUI_Votazione	È responsabile dell'interazione con l'utente per la gestione della votazione di un medico.
GestioneVotazione	Permette di votare un medico.
GUI_Referto	È responsabile dell'interazione con l'utente per la gestione del referto.
GestioneReferto	Permette di compilare e caricare un referto.
GUI_Collegamento	È responsabile dell'interazione con l'utente per la gestione del collegamento con un medico
GestioneCollegamento	Permette di richiedere e accettare un collegamento.
GUI_Ricetta	È responsabile dell'interazione con l'utente per la gestione della ricetta.
GestioneRicetta	Permette di richiedere, caricare e scaricare una ricetta.
GUI_Appuntamento	È responsabile dell'interazione con l'utente per la gestione dell'appuntamento
GestioneAppuntamento	Permette di prenotare, modificare e accettare un appuntamento.
GestioneEmail	Permette di gestire le notifiche generate dagli utenti inviando email.

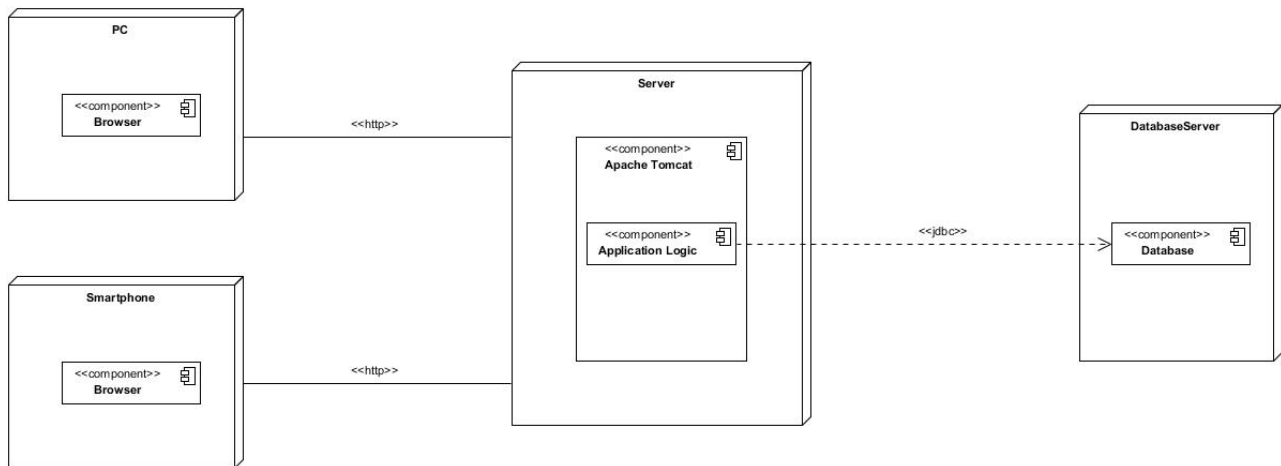


---

## 3.2 MAPPING HARDWARE/SOFTWARE

Il sistema utilizza un'architettura Client/Server. Il WebServer è rappresentato da Apache Tomcat 9 ed è situato su una singola macchina, la logica del sistema è costituita da Java Servlet mentre l'interfaccia utente è realizzata utilizzando pagine JSP. Il Client è rappresentato dal Web Browser utilizzato dall'utente.

La comunicazione tra i nodi è rappresentata da richieste e risposte http tra client e server, e da query in JDBC tra server e database.



## 3.3 GESTIONE DEI DATI PERSISTENTI

Si rimanda al documento "SDD\_DATIPERSISTENTI\_MEDASSISTANT".

---

## 3.4 CONTROLLO DEGLI ACCESSI E DELLA SICUREZZA

MedAssistant è un sistema gestionale che permette una semplice e intelligente interazione fra paziente e medico, i quali svolgeranno le tipiche azioni che tipicamente vengono eseguite nelle modalità che tutti conosciamo. È stata disegnata una matrice per il controllo degli accessi che mostra le operazioni consentite da ogni attore su ogni oggetto. Il controllo degli accessi è garantito tramite l'utilizzo di una coppia di credenziali personali (email e password) univoche. Queste devono essere inserite ogni volta che si desidera utilizzare il sistema per iniziare una nuova sessione, la quale terminerà quando l'utente effettuerà il logout o quando chiuderà il browser.

Matrice degli accessi

Attori oggetto	Appuntamento	Referto	Ricetta	Votazione	Area Personale
Utente non registrato				Visualizza	Crea
Paziente	Richiede Visualizza Modifica	Visualizza	Richiede Visualizza Scarica	Crea Visualizza Modifica	Visualizza Modifica
Medico di base	Approva Visualizza Modifica	Crea Visualizza	Crea Visualizza	Visualizza	Visualizza Modifica
Medico specialista	Approva Visualizza Modifica	Crea Visualizza	Visualizza	Visualizza	Visualizza Modifica

## 3.5 CONTROLLO DEL FLUSSO GLOBALE DEL SISTEMA

Il Web Server si occupa di gestire le varie richieste effettuate dal client. Il server smista le richieste alle classi Java Servlet opportune che si occuperanno di gestire la richiesta, eventualmente interagire con il model, e dare una risposta. Dopodiché il server crea la pagina jsp che verrà poi convertita in pagina html e visualizzata dall'utente.

---

## 3.6 CONDIZIONI BOUNDARY

Le condizioni limite riguardano l'accensione e lo spegnimento del sistema per quanto riguarda il lato Server.

Dal lato Client si riferiscono agli errori di connessione al server.

Avvio del sistema

Il sistema dopo essersi avviato presenta un'interfaccia ai client. Dopo aver effettuato l'autenticazione ogni utente può accedere alle funzionalità disponibili.

Name	Startup Server	
Actors	Amministratore	
Flows of events	Amministratore  Decide di avviare il server, accede al pc dove risiede il server e lo avvia	Sistema  Attiva il server e tutti i servizi in remoto, rendendosi disponibile ad eventuali richieste da parte degli utenti

---

## Terminazione del sistema

È possibile terminare il sistema se e solo se tutti i sottosistemi sono stati disattivati in precedenza.

Prima della disattivazione totale del database-server e dell'application-server verranno disconnessi tutti

i client connessi al sistema.

Name	Shutdown Server	
Actors	Amministratore	
Flows of events	<p>Amministratore</p> <p>Decide di arrestare il server, accede al pc dove risiede il server e lo arresta</p>	<p>Sistema</p> <p>Termina tutte le connessioni attive. Successivamente termina tutte le operazioni in corso e il server</p>

## Fallimento del sistema

Nel caso si verifichi un errore dovuto all'hardware o al software si cercherà di ripristinare una configurazione del sistema precedente allo stato d'errore. Poiché i dati sono gestiti dal DBMS non c'è alcun rischio di perderli. Tuttavia, non è da escludere la perdita dei dati, se si verifica un guasto al supporto di memorizzazione dei dati nel database-server.

---

## 4. SERVIZIO DEI SOTTOSISTEMI

Di seguito, saranno mostrati i servizi offerti dai sottosistemi con una breve descrizione.

### Gestione profilo

Servizi offerti	
Servizio	Descrizione
Registrazione paziente	Consente di registrare un nuovo account di tipo paziente
Registrazione medico	Consente di registrare un nuovo account di tipo medico
Autenticazione	Consente di accedere alla propria area personale
Logout	Consente di effettuare il logout dal sito
Modificare profilo paziente	Consente al paziente di modificare il proprio profilo
Modificare profilo medico	Consente al medico di modificare il proprio profilo

### Gestione appuntamento

Servizi offerti	
Servizio	Descrizione
Prenotazione appuntamento	Consente di prenotare un nuovo appuntamento
Modificare appuntamento paziente	Consente al paziente di modificare un appuntamento
Modificare appuntamento medico	Consente al medico di modificare un appuntamento
Accettare appuntamento	Consente di accettare un appuntamento
Rinviare appuntamento	Consente di rinviare un appuntamento

### Gestione ricetta

Servizi offerti	
Servizio	Descrizione
Richiedere ricetta	Consente di chiedere una nuova ricetta
Scaricare ricetta	Consente di scaricare una ricetta
Caricare ricetta	Consente di caricare una ricetta
Rifiuto rinvio ricetta	Consente di rifiutare una richiesta di rinvio di una ricetta

### Gestione ricerca

Servizi offerti	
Servizio	Descrizione
Cercare medico in base alla zona e al tipo	Consente di cercare un medico in base alla zona e al tipo
Cercare medico	Consente di cercare un medico in base al nome e cognome
Cercare paziente	Consente di cercare un paziente in base al nome e cognome

---

### Gestione collegamento

Servizi offerti	
Servizio	Descrizione
Richiedere collegamento	Consente di richiedere un collegamento con un medico
Accettare collegamento	Consente di accettare una richiesta di collegamento da parte di un paziente

### Gestione votazione

Servizi offerti	
Servizio	Descrizione
Votare medico	Consente di votare un medico

### Gestione referto

Servizi offerti	
Servizio	Descrizione
Compilare referto	Consente di compilare e caricare un referto