**Software Requirements Specification**

**for**

Phoenix

**Version 2.0**

**Prepared by**

**Group Name: Ombrello**

| **Roberti Cristiano** | **student 1** | [**roberti.cristiano.stud@itisgalileiroma.it**](mailto:roberti.cristiano.stud@itisgalileiroma.it) |
| --- | --- | --- |
| **Marcolini Andrea** | **student 2** | [**marcolini.andrea.stud@itisgalileiroma.it**](mailto:marcolini.andrea.stud@itisgalileiroma.it) |
| **Guarracino Emanuele** | **student 3** | [**guarracino.emanuele.stud@itisgalileiroma.it**](mailto:guarracino.emanuele.stud@itisgalileiroma.it) |
|  |  |  |
|  |  |  |

| **Instructor:** | **Pisani Massimo** |
| --- | --- |
| **Course:** | **Gestione di progetto e organizzazione di impresa (GPOI)** |
| **Lab Section:** |  |
| **Teaching Assistant:** | **Tartaglione Domenica** |
| **Date:** | **5/03/2025** |

**Contents**

[**1 Introduction 1**](#_1fob9te)

**1.1 Document Purpose 1**

**1.2 Product Scope 1**

**1.3 Intended Audience and Document Overview 1**

**1.4 Definitions, Acronyms and Abbreviations 2**

**1.5 Document Conventions 2**

**1.6 References and Acknowledgments 2**

**2 Overall Description 3**

[2.1 Product Overview 3](#_2bn6wsx)

**2.2 Product Functionality 3**

**2.3 Design and Implementation Constraints 3**

**2.4 Assumptions and Dependencies 4**

**3 Specific Requirements 4**

[3.1 External Interface Requirements 4](#_23ckvvd)

**3.2 Functional Requirements 5**

**3.3 Use Case Model 6**

**4 Other Non-functional Requirements 12**

[4.1 Performance Requirements 12](#_1mrcu09)

**4.2 Safety and Security Requirements 12**

**4.3 Software Quality Attributes 12**

**5 Other Requirements 13**

**Revisions**

| **Version** | **Primary Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| 1.0 | [Roberti Cristiano](mailto:roberti.cristiano.stud@itisgalileiroma.it)  [Marcolini Andrea](mailto:marcolini.andrea.stud@itisgalileiroma.it)  [Guarracino Emanuele](mailto:guarracino.emanuele.stud@itisgalileiroma.it) | Prima versione del documento | 18/12/2024 |

# **Introduction**

Questo sistema è progettato per sviluppare un’applicazione centralizzata per l’ottimizzazione della gestione dei pazienti all’interno di una clinica privata. In un contesto sanitario in continua evoluzione, è fondamentale migliorare l’efficienza operativa e la comunicazione nel processo di cura.

## **Document Purpose**

Questo documento specifica i requisiti software per il sistema Phoenix progettato per una clinica privata. La versione attuale del documento è la 1.0.

Questo SRS copre l'intero sistema, inclusi i requisiti funzionali e non funzionali, le interfacce utente e hardware, e le specifiche di sicurezza.

## **Product Scope**

Il sistema Phoenix è un'applicazione web che consente la registrazione, la pianificazione degli appuntamenti, la gestione delle cartelle cliniche e la comunicazione nel processo di cura.

I benefici derivati dall’uso di questo sistema sono:

* Un riduzione dei tempi di attesa per le visite;
* Una qualità superiore del servizio rispetto alla media.

Gli obiettivi principali sono:

* Una migliore facilità nei processi amministrativi;
* Creazione di un’ UI intuitiva.

## **Intended Audience and Document Overview**

Il presente documento è destinato ai membri del team Ombrello, ai professori e ad una eventuale clinica privata che vorrebbe utilizzare il sistema Phoenix. La struttura del documento è organizzata in sezioni: una parte introduttiva e discorsiva e una parte tecnica. Si suggerisce di leggere prima le sezioni introduttive così da comprendere le intenzioni dello sviluppo del sistema e, in generale, le funzionalità del sistema.

## **Definitions, Acronyms and Abbreviations**

Glossario:

* **Sistema**: Il software “Phoenix”, progettato per semplificare ed automatizzare le operazioni

all’interno di ambienti sanitari privati.

* **UI**: User Interface, punto di interazione tra utente e sistema. L’interfaccia utente contiene tutti gli elementi visivi e interattivi che consentono all’utente di comunicare con il sistema.
* **GDPR**: Garante per la protezione dei dati personali è una normativa dell'Unione Europea. Essa stabilisce le linee guida per la raccolta e il trattamento dei dati personali all'interno dell'UE e per le organizzazioni che trattano dati di cittadini europei, indipendentemente dalla loro ubicazione.
* **IEEE**: Institute of Electrical and Electronics Engineers si occupa dello sviluppo di standard tecnici in vari settori, come le telecomunicazioni, l'elettronica e l'informatica.
* **API**: Application programming interface, ovverosia l’insieme di regole che consentono a diverse applicazioni di comunicare fra di loro, condividendo dati e funzionalità senza conoscere l’interno reciproco di esse.
* **Database**: Collezione di dati in cui all’interno si possono modificare questi ultimi tramite operazioni di creazione,eliminazione,aggiornamento.
* **Database relazionale**: Un database relazionale è un sistema di gestione dei dati che organizza le informazioni in tabelle composte da righe e colonne, vi sono anche associazioni tra queste tabelle. Ogni tabella rappresenta un'entità e le colonne ne descrivono gli attributi.
* **Sessione Utente**: Periodo in cui un utente è autenticato e interagisce con un'applicazione, durante il quale il sistema tiene traccia delle sue informazioni e attività. Termina quando l'utente esce o dopo un periodo di inattività.

## **Document Conventions**

Il presente documento segue le linee guida IEEE per la formattazione di un SRS.

Il font, la definizione delle pagine, gli spazi, etc… è tutto descritto nel documento IEEE.

## **References and Acknowledgments**

IEEE:

* “IEEE Guide for Software Requirements Specifications," in IEEE Std 830-1984 , vol., no., pp.1-26, 10 Feb. 1984, doi: 10.1109/IEEESTD.1984.119205.

<https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=278253&isnumber=6883>

GDPR:

* <https://www.garanteprivacy.it/regolamentoue> (Principale)
* <https://www.garanteprivacy.it/temi/sanita-e-ricerca-scientifica>
* <https://www.garanteprivacy.it/il-testo-del-regolamento>
* <https://www.garanteprivacy.it/codice>

Manuali e guide tecniche

* MySQL 8.0 Reference Manual, disponibile online: <https://dev.mysql.com/doc/>
* MariaDb Reference Manual, disponibile online:

<https://mariadb.com/kb/it/documentazione-di-mariadb/>

# **Overall Description**

## **Product Overview**

Il sistema si propone come sostituto di diversi software di gestione di cliniche private e offre una migliore esperienza di utilizzo e una semplicità non banale, rispetto ai competitor che realizzano software frammentati Phoenix sarà compatto e con tutte le funzionalità in uno.

## **Product Functionality**

● Unire nell’ UI sia il numero di prenotazioni che i moduli al fine di ottimizzare i tempi;

● Gestire le cartelle cliniche dei pazienti

Il sistema sarà utilizzato dai seguenti attori:

● **Dottore** : Dopo aver effettuato le dovute diagnosi, richiede alla clinica la

cartella medica del paziente per aggiornare o aggiungere nuovi dettagli.

Inoltre dovrà rilasciare la fattura per il pagamento dopo ogni visita.

● **Infermiere** : Attraverso notifiche di sistema,inviate dalla clinica, esso dovrà

gestire la posizione dei letti disponibili nei vari reparti. Inoltre sarà addetto al

controllo della somministrazione dei medicinali.

● **Paziente** : Potranno prenotare le visite e richiedere la distribuzione dei propri dati personali.

● **Clinica** : Ente che gestirà tutte le richieste degli utenti, personale medico e non medico.

## **Design and Implementation Constraints**

### Vincoli Tecnologici

Il sistema dovrà supportare una web app su desktop utilizzando e supportando l’ultima versione di un browser moderno (ad esempio: Chrome,Firefox, Edge). Deve utilizzare un database relazionale (ad esempio: MySQL) per la gestione dei dati. Deve essere sviluppato utilizzando tecnologie open source (es. Java, HTML5, CSS,JavaScript,Python).

### Vincoli di Sicurezza

Utilizzo di un metodo sicuro per la trasmissione dei dati che offra la possibilità di cifrare il contenuto del messaggio sulla rete (ad esempio, tramite HTTPS).

L’accesso al sistema deve essere protetto da un autenticazione (nome utente e password), nel caso del ruolo del dottore ci dovrà essere anche un autenticazione della sua professione durante la fase di registrazione.

I dati degli utenti e amministratori devono essere gestiti in conformità con il Regolamento Generale sulla Protezione dei Dati (GDPR).

### Vincoli di Usabilità

L' UI deve essere semplice e intuitiva, progettata per utenti e amministratori con competenze tecniche limitate.

Il sistema dovrà supportare almeno due lingue, italiano e inglese.

### Vincoli Temporali

Il sistema deve essere operativo e funzionante entro la fine dell’anno scolastico 2024/2025

(Giugno), in più durante il percorso bisognerà rispettare tutte le deadlines (SAL).

### Vincoli Economici

La realizzazione del sistema non prevede un budget.

### Vincoli Legali

Il sistema deve rispettare tutte le normative nazionali e europee sulla gestione delle strutture sanitarie e dei dati personali, rispettando così la trasparenza e il diritto alla privacy soprattutto per i dati più sensibili come la cartella clinica.

## **Assumptions and Dependencies**

Per adesso il sistema non ha dipendenze teoriche

# **Specific Requirements**

## **External Interface Requirements**

### **User Interfaces**

L’ UI sarà semplice e intuitiva, avrà una schermata home composta da un'immagine in alto a destra che sarà un collegamento con il proprio account, inoltre per ogni ruolo questa schermata cambierà mostrando funzionalità diverse, come ad esempio il sistema di prenotazioni. Ci saranno diversi pulsanti che avranno tutti funzionalità differenti, come ad esempio la stampa della ricevuta da pagare oppure il sistema di notifiche che terrà aggiornato l’utente o l’amministratore in ogni momento.

Nella propria area priva vi sarà un menù dove si potrà richiedere la propria cartella clinica e , in caso di dimenticanza, si potrà cambiare il proprio username o la propria password.

### **Hardware Interfaces**

Il sistema non riscontra utilizzi di interfacce hardware (ad esempio sensori).

### **Software Interfaces**

Il sistema avrà al suo interno diverse API, tra cui Gmail,Maps,Google,etc…

Che serviranno per sviluppare un ambiente più user friendly, ma soprattutto per integrare al meglio determinate funzioni, come ad esempio le conferme delle prenotazioni.

## **Functional Requirements**

I requisiti funzionali sono divisi da:

1.1 fino 1.4 sono i RF di Gestione degli account;

2.1 fino 2.3 sono i RF di Gestione prenotazioni;

3.1 fino 3.2 sono i RF di Gestione delle richieste di prenotazione(Amministratori);

4.1 fino 4.2 sono i RF di Generazione Report.

### RF 1.1

Il sistema richiede che tutti gli utenti siano registrati per utilizzarlo. Ciò avviene tramite una pagina di SignUp.

### RF 1.2

La pagina Home è la prima visualizzata, ma avrà funzionalità molto limitate, dunque l’utente dovrà o fare l’accesso tramite la pagina LogIn oppure creare un proprio account nella pagina SignUp.

### RF 1.3

Nella fase di registrazione verranno richieste le seguenti informazioni:  
nome, cognome, email, telefono, password, data di nascita, codice fiscale, ruolo (Paziente, infermiere, dottore)(Se nel ruolo si seleziona una professione sanitaria bisognerà immettere un codice id\_numerico che funzionerà come da badge).

### RF 1.4

Le funzionalità sono dipendenti dal ruolo dell’utente, ovviamente un dottore avrà più funzioni da svolgere e gestire a differenza di un paziente.

### RF 2.1

Un utente una volta registratosi dovrà avere accesso ad una schermata, avviata dopo una pressione di un pulsante, di prenotazione per visite mediche.

### RF 2.2

L’utente dopo aver fatto con successo l’operazione di prenotazione dovrà ricevere una notifica per email che annunci la buona riuscita dell’operazione.

### RF 2.3

Gli utenti avranno un periodo di tempo in cui potranno eliminare e annullare la propria richiesta di visita medica.

### RF 3.1

Gli amministratori devono poter accettare le richieste di prenotazioni.

### RF 3.2

Gli amministratori devono poter declinare la richiesta di prenotazione

### RF 4.1

Gli amministratori devono poter generare report settimanali sullo stato dei pazienti nelle cartelle cliniche.

### RF 4.2

Gli amministratori devono poter visualizzare le cartelle cliniche dei pazienti.

## **Use Case Model**

Il diagramma Use Case del sistema è già stato consegnato in precedenza.

### Registrazione

**Author –** Ombrello

**Purpose** - Permettere agli utenti di registrarsi nel sistema per accedere ai servizi offerti.

**Requirements Traceability –** 1

**Priority** - High

**Preconditions** - L'utente deve avere accesso a Internet, un indirizzo email e password validi.

**Post conditions** - L'utente è registrato nel sistema e può accedere con le proprie credenziali.

**Actors** – Utente, Sistema

**Extends –** Error output

**Flow of Events**

* 1. L'utente accede alla pagina di registrazione.
  2. L'utente compila il modulo di registrazione.
  3. L'utente invia il modulo.
  4. Il sistema verifica i dati e crea un nuovo account.
  5. Il sistema invia un'email di conferma all'utente.

**Includes** (Nessuno)

**Notes/Issues** - verificare la validità dell'email.

### Login

**Author –** Ombrello

**Purpose** - Consentire agli utenti di accedere al sistema utilizzando le proprie credenziali.

**Requirements Traceability –** 1

**Priority** - High

**Preconditions** - L'utente deve essere registrato.

**Post conditions** - L'utente è autenticato e ha accesso al sistema.

**Actors** – Utente, Sistema

**Extends –** Error output

**Flow of Events**

* 1. L'utente accede alla pagina di login.
  2. L'utente inserisce le credenziali.
  3. Il sistema verifica le credenziali.
  4. L'utente viene reindirizzato alla home.

**Includes** (Nessuno)

**Notes/Issues** - Implementare la funzionalità di recupero password.

* + 1. Richiesta cartella clinica

**Author –** Ombrello

**Purpose** - Permettere agli utenti di richiedere la propria cartella clinica.

**Requirements Traceability –** 1

**Priority** - Medium

**Preconditions** - L'utente deve essere autenticato.

**Post conditions** -La richiesta di cartella clinica è registrata nel sistema.

**Actors** – Utente, Sistema

**Extends –** Error output

**Flow of Events**

* 1. L'utente accede alla sezione "Richiesta Cartella Clinica".
  2. L'utente compila il modulo di richiesta.
  3. L'utente invia la richiesta.
  4. Il sistema conferma la ricezione della richiesta.

**Includes** (Nessuno)

**Notes/Issues** - verificare i diritti di accesso.

* + 1. Prenotazione visita medica

**Author –** Ombrello

**Purpose** -Consentire agli utenti di prenotare visite mediche.

**Requirements Traceability –** 1

**Priority** - High

**Preconditions** - L'utente deve essere autenticato e avere accesso a un elenco di medici disponibili.

**Post conditions** - La visita medica è prenotata e registrata nel sistema.

**Actors** – Utente, Sistema, Medico

**Extends –** Error output

**Flow of Events**

* 1. L'utente accede alla sezione "Prenotazione Visita".
  2. L'utente seleziona il medico e la data.
  3. L'utente conferma la prenotazione.
  4. Il sistema registra la prenotazione e invia una conferma all'utente.

**Includes** (Nessuno)

**Notes/Issues** - Nessuno

* + 1. Rilascio fattura

**Author –** Ombrello

**Purpose** - Generare e rilasciare fatture per i servizi medici forniti.

**Requirements Traceability –** 1

**Priority** - Medium

**Preconditions** - L'utente deve avere completato una visita medica.

**Post conditions** - La fattura è generata e inviata all'utente.

**Actors** – Utente, Sistema, Medico

**Extends –** Error output

**Flow of Events**

* 1. Il sistema identifica la visita medica completata.
  2. Il sistema genera la fattura basata sui servizi forniti.
  3. Il sistema invia la fattura all'utente via email.

**Includes** (Nessuno)

**Notes/Issues** - Verificare le informazioni di pagamento.

* + 1. Inserimento dati nella cartella clinica

**Author –** Ombrello

**Purpose** - Consentire ai medici di inserire dati nella cartella clinica dei pazienti.

**Requirements Traceability –** 1

**Priority** - High

**Preconditions** - Il medico deve essere autenticato e avere accesso alla cartella clinica del paziente.

**Post conditions** - I dati sono aggiornati nella cartella clinica del paziente.

**Actors** – Medico, Sistema

**Extends –** Error output

**Flow of Events**

* 1. Il medico accede alla cartella clinica del paziente.
  2. Il medico inserisce i dati richiesti.
  3. Il medico salva le modifiche.
  4. Il sistema conferma l'aggiornamento.

**Includes** (Nessuno)

**Notes/Issues** - Assicurarsi che i dati siano conformi alle normative sulla privacy.

* + 1. Gestione dei letti nei reparti

**Author –** Ombrello

**Purpose** - Gestire la disponibilità e l'assegnazione dei letti nei reparti.

**Requirements Traceability –** 1

**Priority** - High

**Preconditions** - Il personale deve essere autenticato e avere accesso al sistema di gestione dei letti.

**Post conditions** - La disponibilità dei letti è aggiornata nel sistema.

**Actors** – Infermiere, Sistema

**Extends –** Error output

**Flow of Events**

* 1. Il personale accede alla sezione di gestione dei letti.
  2. Il personale visualizza la disponibilità attuale dei letti.
  3. Il personale assegna un letto a un paziente.
  4. Il sistema aggiorna la disponibilità dei letti.

**Includes** (Nessuno)

**Notes/Issues** - Monitorare costantemente la disponibilità dei letti.

* + 1. Controllo farmaci

**Author –** Ombrello

**Purpose** - Monitorare e gestire l'inventario dei farmaci disponibili nel sistema.

**Requirements Traceability –** 1

**Priority** - Medium

**Preconditions** - Il personale deve essere autenticato e avere accesso al sistema di gestione dei farmaci.

**Post conditions** - L'inventario dei farmaci è aggiornato e monitorato.

**Actors** – Infermiere, Sistema

**Extends –** Error output

**Flow of Events**

* 1. Il personale accede alla sezione di controllo farmaci.
  2. Il personale visualizza l'inventario attuale.
  3. Il personale aggiorna le quantità dei farmaci.
  4. Il sistema conferma l'aggiornamento dell'inventario.

**Includes** (Nessuno)

**Notes/Issues** - Assicurarsi che le scorte siano sempre sufficienti e pianificare ordini di rifornimento.

# **Other Non-functional Requirements**

## **Performance Requirements**

Per questo progetto le performance non sono indispensabili essendo un progetto che ha come finalità la corretta funzionamento. Ma sicuramente non vi sarà un tempo di attesa infinito.

## **Safety and Security Requirements**

Per far risultare il sistema sicuro, da un punto di vista informatico, dovrà essere necessario l’utilizzo del protocollo HTTPS,delle tecniche di cifratura per conservare i dati, anche i più sensibili, in totale sicurezza e di un algoritmo di autenticazione di nome e password. Bisognerà inoltre sviluppare un piano di recovery nel caso di data breach o di errori di memoria del database. Perciò dovrà essere formata una persona per ricoprire il ruolo di Responsabile della Protezione dei Dati (RPD).

In aggiunta il sistema dovrà avere un algoritmo di verifica di un id\_numerico che sarà di vitale importanza nel momento della registrazione degli account, così che solamente il personale sanitario autorizzato, in possesso di questo id\_numerico, sarà in grado di avere ruoli da amministratore.

## **Software Quality Attributes**

### Sicurezza

RNF 1.1: L’accesso al sistema deve essere protetto da autenticazione tramite nome utente e password.

RNF 1.2: Tutte le comunicazioni tra il sistema e i client devono avvenire tramite il protocollo HTTPS.

### Usabilità

RNF 2.1: L’interfaccia utente deve essere intuitiva e accessibile anche a utenti con

competenze tecniche limitate.

### Manutennibilità

RNF 3.1: Il codice del sistema deve essere documentato in modo da garantire la facilità di manutenzione.

RNF 3.2: eventuali bug devono poter essere risolti con rilascio di patch entro 168 ore dalla segnalazione.

### Compatibilità

RNF 4.1: il sistema deve essere compatibile con i principali browser nelle loro versioni più recenti.

RNF 4.2: Deve supportare sia dispositivi desktop che mobile.

### Localizzazione

RNF 5.1: Il sistema deve essere disponibile in italiano e inglese.

RNF 5.2: La data e l’ora visualizzate devono seguire il formato locale selezionato.

### Conformità Normativa

RNF 6.1:Il sistema deve rispettare il Regolamento Generale sulla Protezione dei Dati(GDPR) per quanto riguarda la gestione dei dati personali.

RNF 6.2: Deve rispettare le normative locali relative alla gestione dei servizi sanitari pubblici.

# **Other Requirements**

Sezione per altri requisiti, anche futuri, non specificati in questo documento.

**Appendix A – Data Dictionary**

**Appendix B - Group Log**

19/12/2024 tempo speso per la realizzazione del documento SRS equivale sù per giù a 4 ore