

# 1) Introduzione

Il seguente documento si pone come obbiettivo quello di descrivere le linee guida in fase di implementazione del sistema NetPharma. In questo documento verranno descritti i trade-off generali e le convenzioni per la nomenclatura ed implementazione di classi ed interfacce.

## 1.1) Object Design Trade-offs

### Interfaccia vs Usabilità:

L'interfaccia deve essere realizzata in maniera chiara e semplice in modo da poter essere utilizzata da diversi tipi di utenti, facendo uso di semplici e ben visibili form e pulsanti con descrizioni esplicative.

### Comprensibilità vs Tempo:

Essendo i tempi di sviluppo del sistema limitati, il codice NON sarà accompagnato da commenti che ne semplificherebbero la comprensione, per permettere l'implementazione del sistema in tempi più contenuti. Di conseguenza, la fase di testing ed eventuali future modifiche saranno più complesse.

### Response Time vs Hardware:

Il sistema deve garantire tempi di risposta rapidi nonostante alto carico di utenza, per cui l'hardware utilizzato dovrà essere adeguato per poterlo permettere.

### Prestazioni vs Costi:

Lo sviluppo del sistema non prevede l'utilizzo di librerie o componenti a pagamento, essendo il progetto privo di budget economico.

### Sicurezza vs Efficienza:

Dati i tempi piuttosto ristretti per lo sviluppo, ci limiteremo ad implementare sistemi di sicurezza basati su username e password crittografata, garantendo un controllo sugli accessi basato su determinati ruoli.

## 1.2) Linee Guida per la Documentazione delle Interfacce

Di seguito le linee guida che gli sviluppatori rispetteranno durante l'implementazione dell'interfaccia:

### Naming Conventions:

Per la nomenclatura è necessario utilizzare nomi di lunghezza medio-corta, che non siano abbreviati e che contengano solo caratteri alfanumerici.

I nomi delle variabili e dei metodi devono seguire la cosiddetta "dromedaryCase" (iniziare con una lettera minuscola e le parole successive con una lettera maiuscola).

Per i metodi in particolare è necessario, salvo particolari casi, definirne il nome con un verbo che specifica l'azione del metodo seguito dall'oggetto su cui l'azione si verifica.

I metodi d'accesso e di modifica delle variabili devono sempre iniziare rispettivamente con get/set prima del nome della variabile.

I commenti, ove utilizzati, devono essere scritti sulla stessa riga per le variabili e prima della dichiarazione per i metodi, descrivendone l'utilizzo.

I nomi delle classi e delle interfacce devono seguire la cosiddetta "CamelCase" (iniziare ogni parola con una lettera maiuscola).

### **1.3) Definizioni, acronimi e abbreviazioni**

RAD: Requirements Analysis Document.

SDD: System Design Document.

ODD: Object Design Document.

### **1.4) Riferimenti**

Requirements Analysis Document (RAD).

System Design Document (SDD).