



# INTEGRATION TEST CASE DOCUMENT

## Nefrapp

---

|              |                                                                                                                                                              |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Riferimento  | Nefrapp_RAD_Vers.1.8, Nefrapp_SDD_Vers.1.0,<br>Nefrapp_TCS_Vers.0.7, Nefrapp_TP_Vers.1.0                                                                     |
| Versione     | 1.0                                                                                                                                                          |
| Data         | 13/12/2019                                                                                                                                                   |
| Destinatario | Professoressa F.Ferrucci                                                                                                                                     |
| Proposto da  | Eugenio Corbisiero, Sara Corrente, Silvio Di<br>Martino, Antonio Donnarumma, Luca Esposito,<br>Matteo Falco, Domenico Musone, Davide<br>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                                   |
|------------|----------|----------------------------------------------------------------------------|------------------------------------------|
| 10/12/2019 | 0.1      | Stesura template,<br>Riferimenti                                           | Sara Corrente                            |
| 13/12/2019 | 0.2      | Introduzione,<br>Approccio di<br>Integration Testing,<br>Criteri Pass/Fail | Matteo Falco                             |
| 13/12/2019 | 0.3      | Componenti da<br>testare                                                   | Sara Corrente,<br>Domenico<br>Musone     |
| 13/12/2019 | 0.9      | Definizioni,<br>acronimi e<br>abbreviazioni,<br>Glossario                  | Eugenio<br>Corbiserio                    |
| 15/12/2019 | 1.0      | Revisione e<br>Validazione PM                                              | Antonio Fasulo,<br>Francesco<br>Garofalo |



## Sommario

|                                           |          |
|-------------------------------------------|----------|
| <b>1.Introduzione</b>                     | <b>5</b> |
| 1.1 Definizioni, acronimi e abbreviazioni | 5        |
| <b>2. Riferimenti</b>                     | <b>5</b> |
| <b>3.Test di Integrazione</b>             | <b>6</b> |
| 3.1 Approccio di Integration Testing      | 6        |
| 3.2 Componenti da testare                 | 6        |
| <b>4. Criteri Pass/Fail</b>               | <b>7</b> |
| <b>5. Glossario</b>                       | <b>7</b> |



## 1.Introduzione

Lo scopo di questo documento è quello di pianificare l'integration testing della piattaforma Nefrapp, che ha l'obiettivo di integrare le componenti del sistema per testarle nel complesso, verificando che non ci siano differenze tra il comportamento atteso e quello effettivo del sistema. Questa procedura consente di rilevare e risolvere quanti più errori possibile, diminuendo le probabilità che questi si presentino durante l'effettivo utilizzo del sistema da parte degli utenti finali.

### 1.1 Definizioni, acronimi e abbreviazioni

| Acronimi | Descrizione                    |
|----------|--------------------------------|
| ITCD     | Integration Test Case Document |
| RAD      | Requirements Analysis Document |
| TP       | Test Plan                      |
| TCS      | Test Case Specification        |
| SDD      | System Design Document         |

## 2. Riferimenti

- Documentazione di progetto (Nefrapp\_RAD\_Vers.1.8, Nefrapp\_SDD\_Vers.1.0, Nefrapp\_TP\_Vers.1.0, Nefrapp\_TCS\_Vers.0.7)
- Bruegge, Dutoit, Object-Oriented Software Engineering.



## 3. Test di Integrazione

### 3.1 Approccio di Integration Testing

La strategia adottata per il testing di integrazione è di tipo **Bottom-up**. Essa consiste nel testare i sottosistemi nei layer del livello più basso della gerarchia individualmente; successivamente vengono integrati e testati i layer di livello superiore, in congiunzione ai sottosistemi già testati precedentemente. Questa procedura viene ripetuta finché non sono stati testati tutti i sottosistemi.

### 3.2 Componenti da testare

Seguendo la strategia **Bottom-up**, le componenti da testare sono le seguenti:

DataStorageLayer:

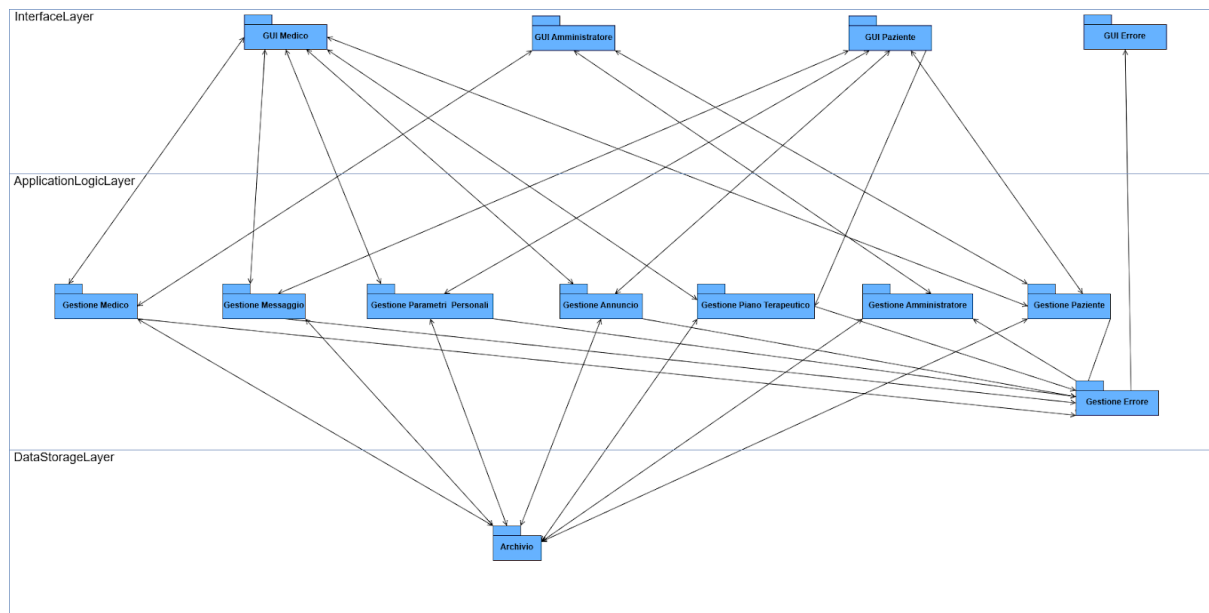
- Archivio

ApplicationLogicLayer:

- GestioneMedico
- GestionePaziente
- GestioneAmministratore
- GestioneMessaggio
- GestioneAnnuncio
- GestionePianoTerapeutico
- GestioneParametriPersonali
- GestioneErrore

InterfaceLayer:

- GUIMedico
- GUIPaziente
- GUIAmministratore
- GUIErrore



## 4. Criteri Pass/Fail

La fase di testing avrà successo se individuerà una o più failure, cioè se l'output osservato sarà diverso da quello atteso. Qualora venisse individuata una failure, questa verrà analizzata e, se essa è legata ad un fault, si procederà alla sua correzione. Una volta completata la correzione, si procederà ad una nuova iterazione della fase di testing per verificare che la modifica non abbia impatto su altri componenti del sistema.

Al contrario, il testing fallirà se l'output osservato sarà uguale all'oracolo.



## 5. Glossario

- **Integration testing:** è la fase del testing nella quale i moduli individuali vengono combinati e testati in gruppo.
- **Bottom-up:** è una strategia nella quale parti individuali del sistema sono specificate in dettaglio, e poi connesse tra loro in modo da formare componenti più grandi, a loro volta interconnesse fino a realizzare un sistema completo.
- **Layer:** raggruppamento di sottosistemi che forniscono servizi correlati, eventualmente realizzati utilizzando servizi di altri layer.
- **Fault:** errore di implementazione in un sistema software che determina l'insorgere di failure.