



# TEST UNIT REPORT

## Nefrapp

---

Riferimento	Nefrapp_RAD_Vers.1.8, Nefrapp_TP_Vers.1.1, Nefrapp_TUR.Vers.1.0	Nefrapp_TCS_Vers.0.7, Nefrapp_TER.Vers.1.0,
Versione	1.0	
Data	14/01/2020	
Destinatario	Professoressa F.Ferrucci	
Proposto da	Eugenio Corbisiero, Sara Corrente, Silvio Di Martino, Antonio Donnarumma, Luca Esposito, Matteo Falco, Domenico Musone, Davide 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
12/01/2020	0.1	Stesura iniziali	Sara Corrente
13/01/2020	0.9	Inserimento TC e schermata	Tutti
14/01/2020	1.0	Revisione	Francesco Garofalo, Antonio Fasulo



<b>1. Introduzione</b>	<b>6</b>
1.1 Scopo del Test	6
1.2 Strumentazione per il Test di unità	6
<b>2. Report dei test</b>	<b>7</b>
2.1 Test Bean	7
2.1.1 Bean Amministratore (Success)	7
2.1.2 Bean AnnuncioCompleto (Success)	7
2.1.3 Bean AnnuncioProxy (Success)	8
2.1.4 Bean Medico (Success)	8
2.1.5 Bean Messaggio Completo (Success)	9
2.1.6 Bean Messaggio Proxy (Success)	9
2.1.7 Bean Paziente (Success)	10
2.1.8 Bean Piano Terapeutico (Success)	10
2.1.9 Bean Scheda Parametri (Success)	11
2.1.10 Bean Utente (Success)	12
2.2 Test Model	13
2.2.1 AmministratoreModel (Success)	13
2.2.4 MessaggioModel (Success)	14
2.2.5 PazienteModel (Success)	14
2.2.6 PianoTerapeuticoModel (Success)	14
2.2.7 SchedaParametriModel (Success)	15
2.2.8 UtenteModel (Success)	15
2.3 Test Control	15
2.3.1 GestioneAccessoTest (rif. TC_GP_1, TC_GA_1) (Success)	15
2.3.2 GestioneAmministratoreTest (rif. TC_GA_3, TC_GA_5, TC_GA_6, TC_GA_7, TC_GA_8) (Failure)	16
2.3.3 GestioneAmministratoreTest (Success)	17
2.3.4 GestioneParametriTest (rif. TC_GM_6, TC_GP_6, TC_GP_7) (Success)	18



2.3.5 GestioneRegistrazioneTest (rif. TC_GM_10) (Failure)	18
2.3.6 GestioneRegistrazioneTest (Success)	19
2.3.7 GestionePazienteTest (rif. TC_GP_4) (Failure)	20
2.3.8 GestioneComunicazioneTest (Success)	21
2.3.9 GestioneMessaggiTest (rif. TC_GM_7, TC_GM_8, TC_GP_9, TC_GP_10) (Success)	21
2.3.10 GestionePianoTerapeuticoTest (rif. TC_GM_11, TC_GP_8) (Success)	22
2.3.11 GestioneAnnunciTest (rif. TC_GM_9, TC_GP_10) (Success)	22



## 1. Introduzione

Il testing di unità rappresenta la fase di testing in cui si assicura che le componenti sviluppate funzionino in isolamento. Questo documento contiene i risultati del testing di unità, le failure rilevate e le eventuali soluzioni.

### 1.1 Scopo del Test

Lo scopo dei Test è quello di riportare il risultato dei test di unità programmati nel documento e stilare una statistica dei casi di success, failure ed error.

### 1.2 Strumentazione per il Test di unità

I test saranno effettuati con JUnit integrato con Eclipse. Saranno riportati in questo documento i risultati della console di JUnit con il numero di failure, success ed error e l'eventuale errore apparso nella console di Eclipse se dovessero verificarsi error.

Sono stati preparati diversi JUnit Test Case per singoli metodi o classi e sono stati poi inglobati in un JUnit Test Suite per effettuare il test dell'intera classe o dell'insieme di classi.

Il testing di unità sarà suddiviso in tre parti, una relativa al package Bean, una relativa al package Model e una relativa al package Control. In ogni parte saranno testate le classi Java contenute nel package.

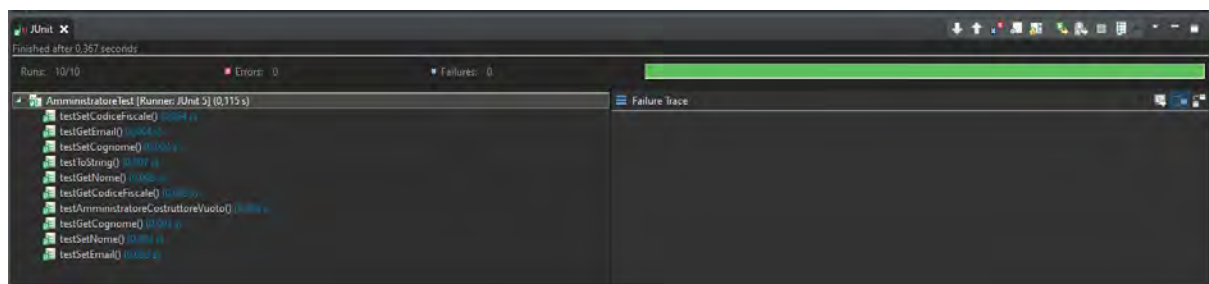


## 2. Report dei test

### 2.1 Test Bean

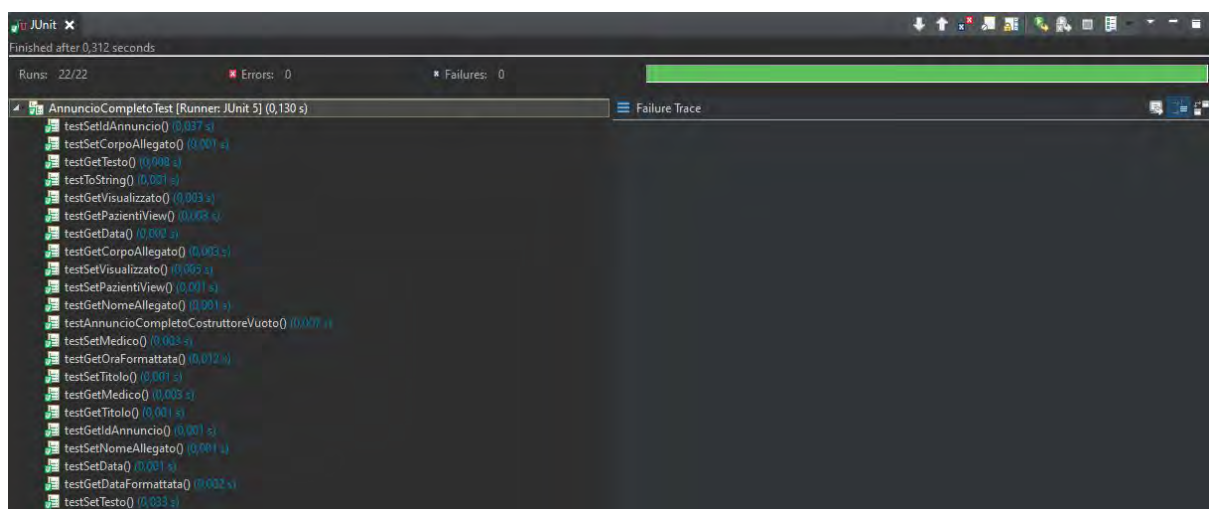
Questa sezione contiene tutti i test di unità relativi ai bean.

#### 2.1.1 Bean Amministratore (Success)



L'esecuzione della classe di test AmministratoreTest ha testato tutti i metodi del bean Amministratore, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

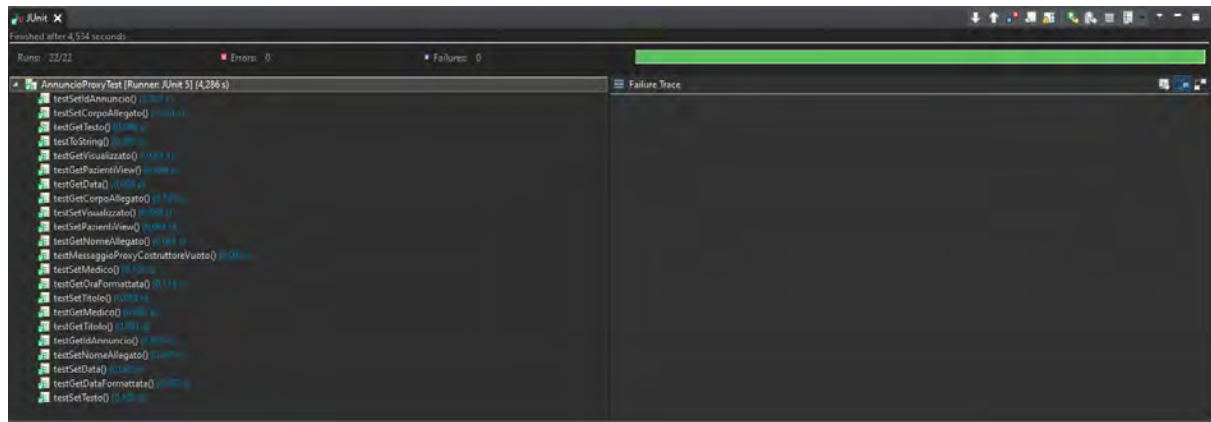
#### 2.1.2 Bean AnnuncioCompleto (Success)





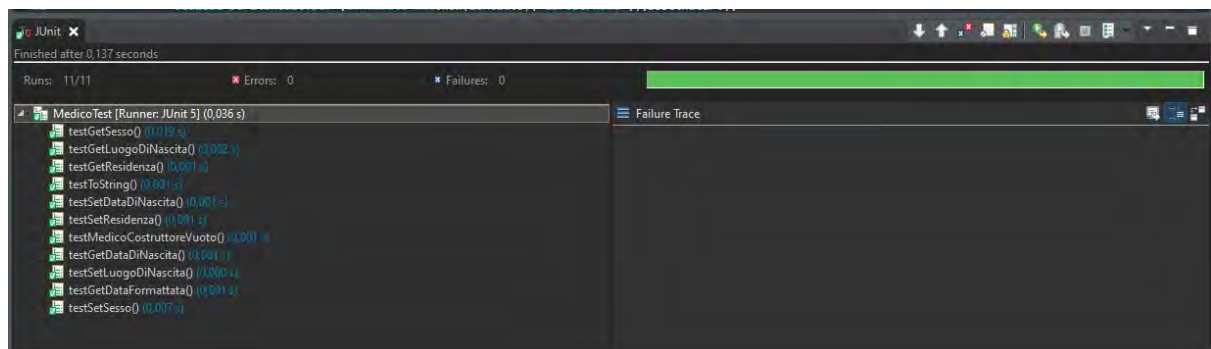
L'esecuzione della classe di test `AnnuncioCompletoTest` ha testato tutti i metodi del bean `AnnuncioCompleto`, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

### 2.1.3 Bean `AnnuncioProxy` (Success)



L'esecuzione della classe di test `AnnuncioProxyTest` ha testato tutti i metodi del bean `AnnuncioProxy`, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

### 2.1.4 Bean `Medico` (Success)

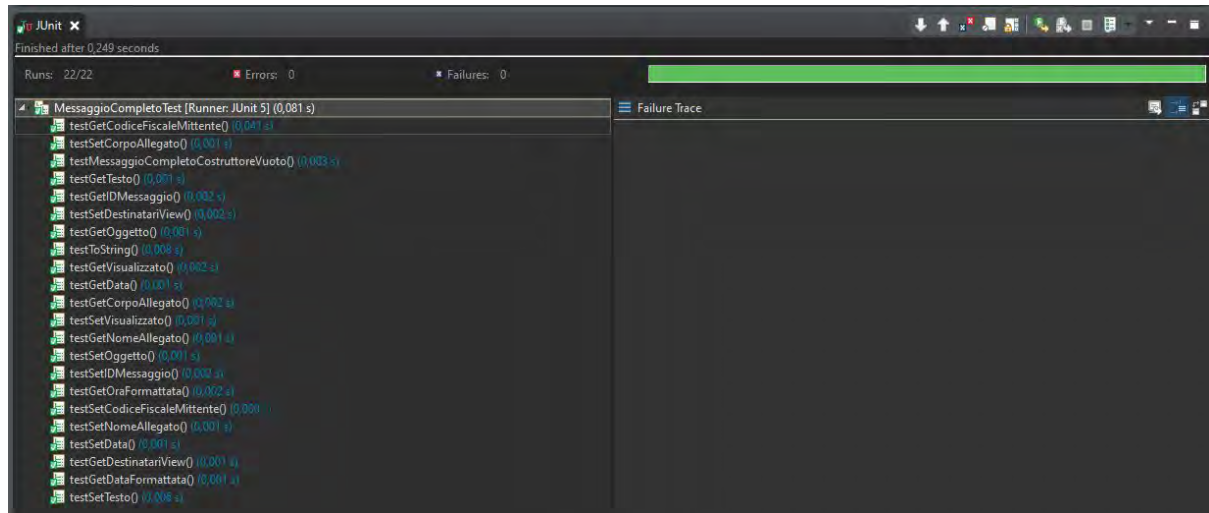


L'esecuzione della classe di test `MedicoTest` ha testato tutti i metodi del bean `Medico`, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.



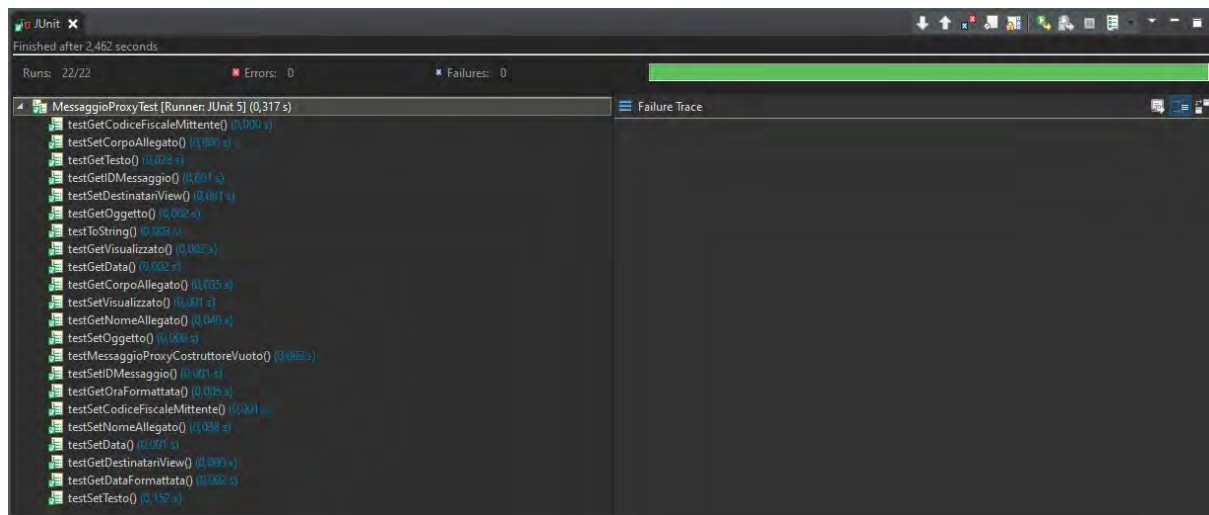


## 2.1.5 Bean Messaggio Completo (Success)



L'esecuzione della classe di test `MessaggioCompletoTest` ha testato tutti i metodi del bean `MessaggioCompleto`, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

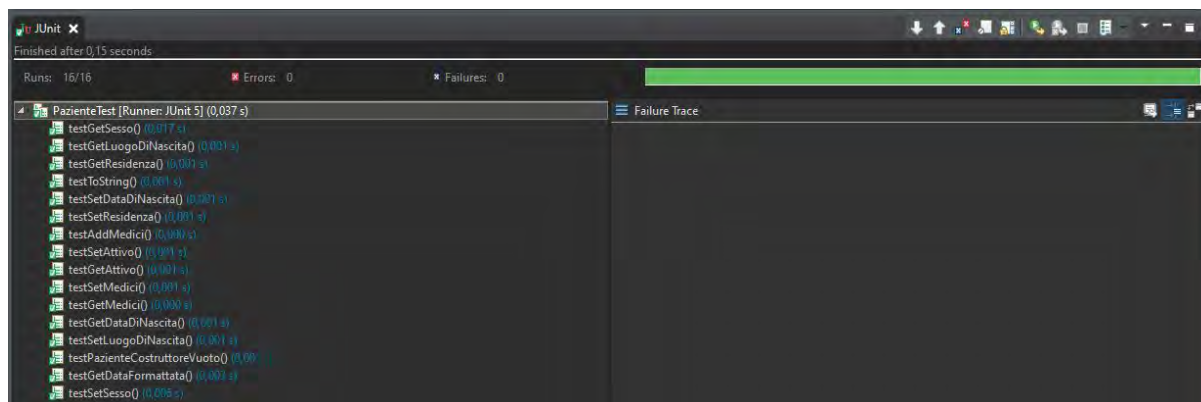
## 2.1.6 Bean Messaggio Proxy (Success)



L'esecuzione della classe di test `MessaggioProxyTest` ha testato tutti i metodi del bean `MessaggioProxy`, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

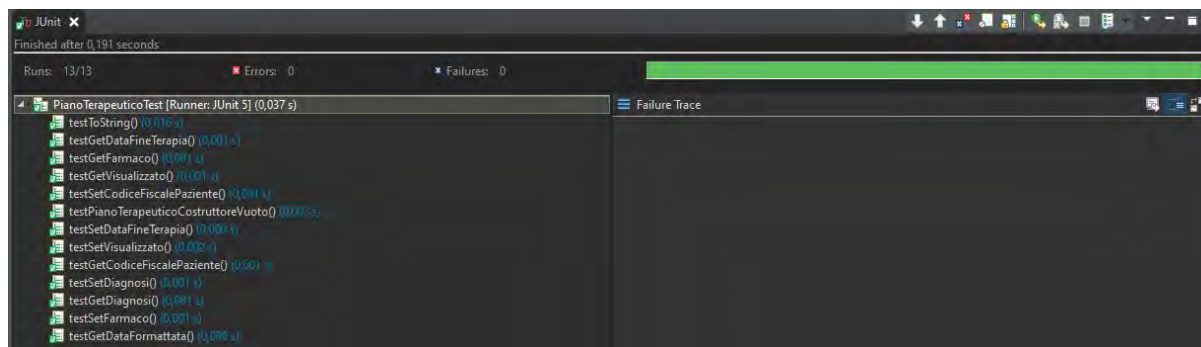


### 2.1.7 Bean Paziente (Success)



L'esecuzione della classe di test PazienteTest ha testato tutti i metodi del bean Paziente, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

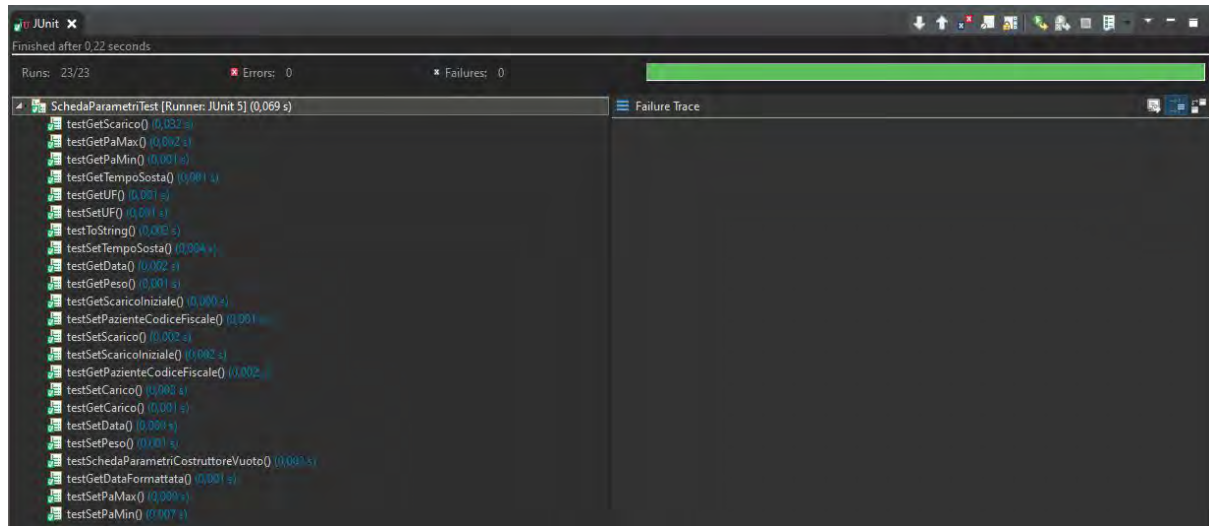
### 2.1.8 Bean Piano Terapeutico (Success)



L'esecuzione della classe di test PianoTerapeuticoTest ha testato tutti i metodi del bean PianoTerapeutico, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.



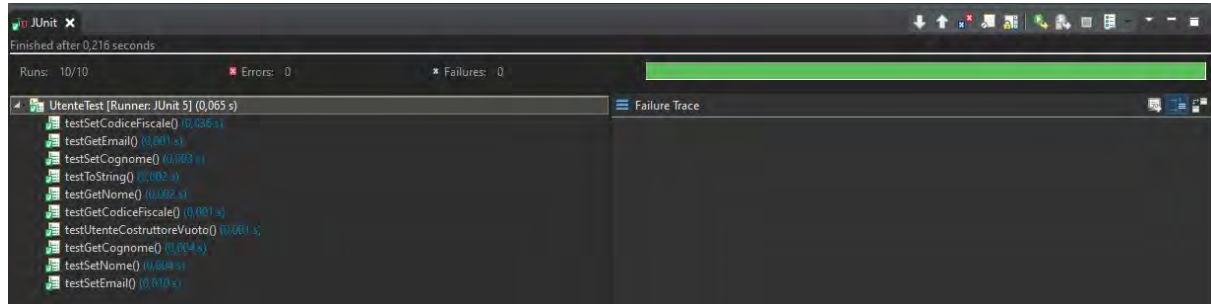
### 2.1.9 Bean Scheda Parametri (Success)



L'esecuzione della classe di test SchedaParametriTest ha testato tutti i metodi del bean SchedaParametri, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.



### 2.1.10 Bean Utente (Success)



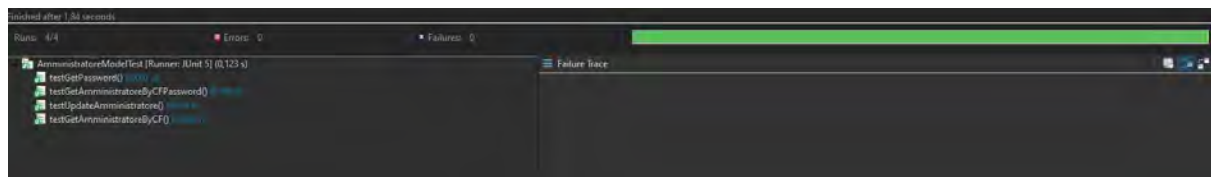
L'esecuzione della classe di test `UtenteTest` ha testato tutti i metodi del bean `Utente`, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.



## 2.2 Test Model

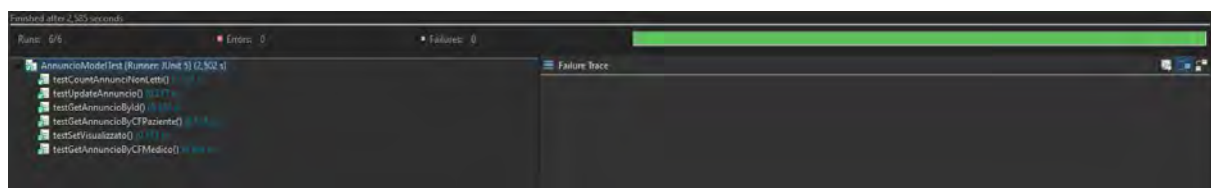
Questa sezione contiene tutti i test di unità relativi ai Model.

### 2.2.1 AmministratoreModel (Success)



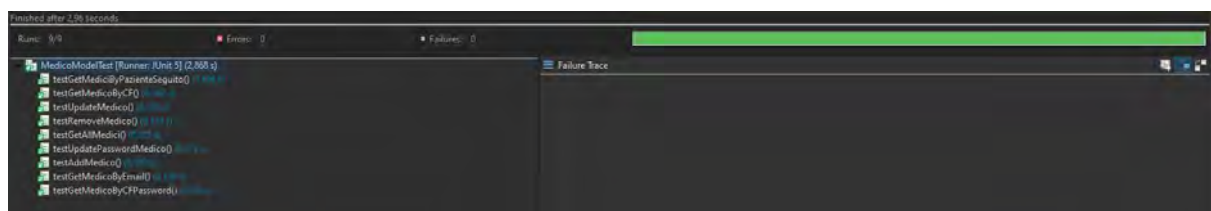
L'esecuzione della classe di test AmministratoreModelTest ha testato tutti i metodi della classe AmministratoreModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

### 2.2.2 AnnuncioModel (Success)



L'esecuzione della classe di test AnnuncioModelTest ha testato tutti i metodi della classe AnnuncioModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

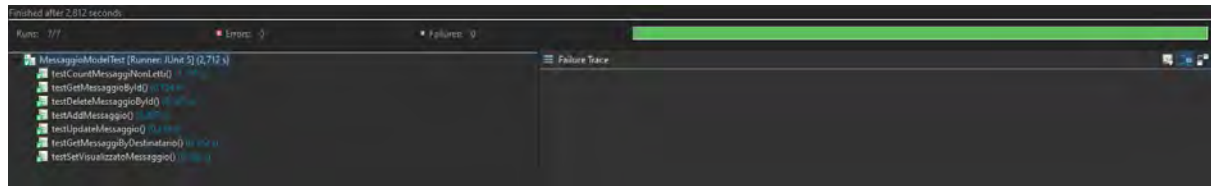
### 2.2.3 MedicoModel (Success)



L'esecuzione della classe di test MedicoModelTest ha testato tutti i metodi della classe MedicoModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

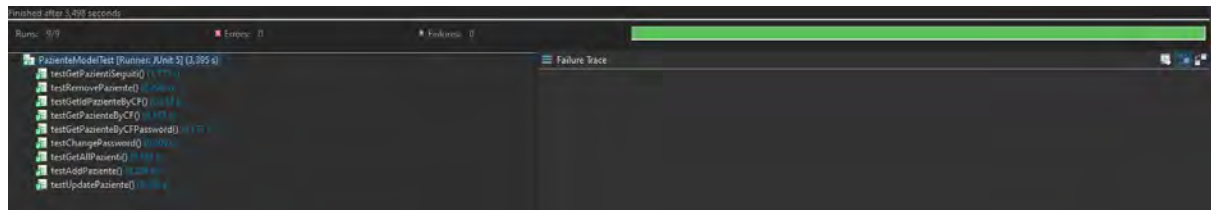


## 2.2.4 MessaggioModel (Success)



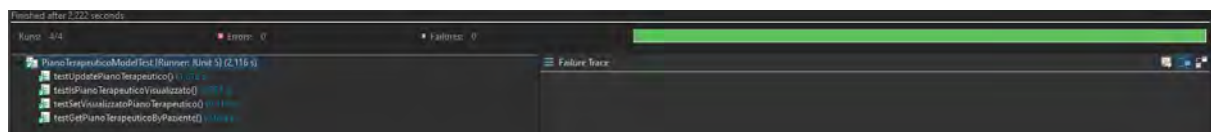
L'esecuzione della classe di test MessaggioModelTest ha testato tutti i metodi della classe MessaggioModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

## 2.2.5 PazienteModel (Success)



L'esecuzione della classe di test PazienteModelTest ha testato tutti i metodi della classe PazienteModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

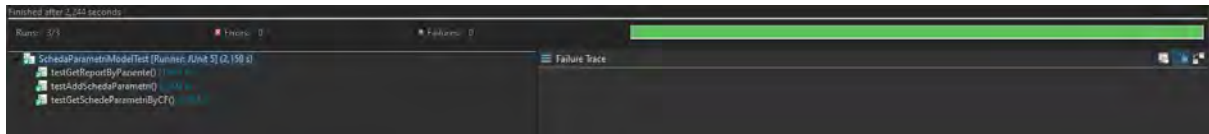
## 2.2.6 PianoTerapeuticoModel (Success)



L'esecuzione della classe di test PianoTerapeuticoModelTest ha testato tutti i metodi della classe PianoTerapeuticoModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

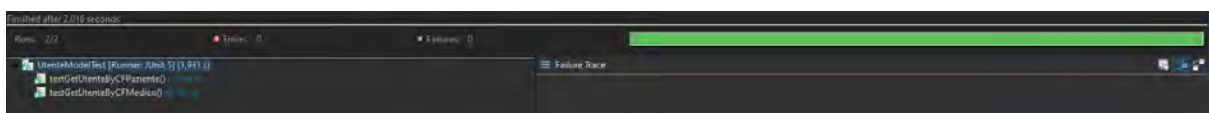


### 2.2.7 SchedaParametriModel (Success)



L'esecuzione della classe di test SchedaParametriModelTest ha testato tutti i metodi della classe SchedaParametriModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

### 2.2.8 UtenteModel (Success)

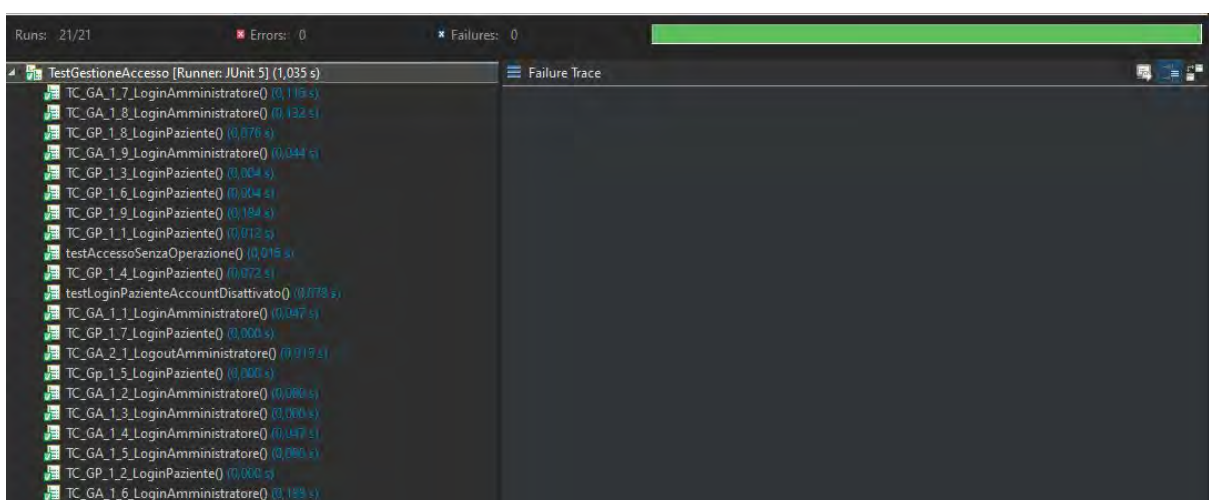


L'esecuzione della classe di test UtenteModelTest ha testato tutti i metodi della classe UtenteModel, eseguendo i metodi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

## 2.3 Test Control

Questa sezione contiene tutti i test di unità relativi ai Control.

### 2.3.1 GestioneAccessoTest (rif. TC\_GP\_1, TC\_GA\_1) (Success)







L'esecuzione della classe di test `GestioneAccessoTest` ha eseguito i 21 casi di test riportati nello screen. Tutte le esecuzioni hanno avuto successo.

### 2.3.2 `GestioneAmministratoreTest` (rif. `TC_GA_3`, `TC_GA_5`, `TC_GA_6`, `TC_GA_7`, `TC_GA_8`) (Failure)

```
Runs: 62/62 Errors: 0 Failures: 3

Failure Trace
org.opentest4j.AssertionFailedError: expected: <./ModificaAccountMedicoView.jsp?notifica=ParamErr> but was: <./dashboard.jsp?notifica=ModificaMedRiuscita>
    at test.control.GestioneAmministratoreTest.TC_GA_8_17_ModificaAccountMedico(GestioneAmministratoreTest.java:1019)
    at java.util.ArrayList.forEach(Unknown Source)
    at java.util.ArrayList.forEach(Unknown Source)
```

```
Runs: 62/62 Errors: 0 Failures: 3

Failure Trace
org.opentest4j.AssertionFailedError: expected: <./ModificaAccountMedicoView.jsp?notifica=ParamErr> but was: <./dashboard.jsp?notifica=ModificaMedRiuscita>
    at test.control.GestioneAmministratoreTest.TC_GA_8_17_ModificaAccountMedico(GestioneAmministratoreTest.java:1019)
    at java.util.ArrayList.forEach(Unknown Source)
    at java.util.ArrayList.forEach(Unknown Source)
```

Il test di `GestioneAmministratore` ha avuto 3 failure. In particolare, le failure sono state rilevate nei seguenti test case:





- TC\_GA\_7\_17\_ModificaAccountMedico() è un metodo che si occupa di effettuare la modifica del paziente con una mail troppo lunga per essere registrata. La failure rilevata è dovuta ad un controllo mancato nella servlet il che consente di registrare un paziente con una mail più lunga del normale.

```
org.opentest4j.AssertionFailedError: expected: <./ModificaAccountMedicoView.jsp?notifica=ParamErr> but was: <./dashboard.jsp?notifica=ModificaMedRiuscita>  
at TestControlFailureAmministratoreTest.TC_GA_8_17_ModificaAccountMedico() (GestioneAmministratoreTest.java:1010)
```

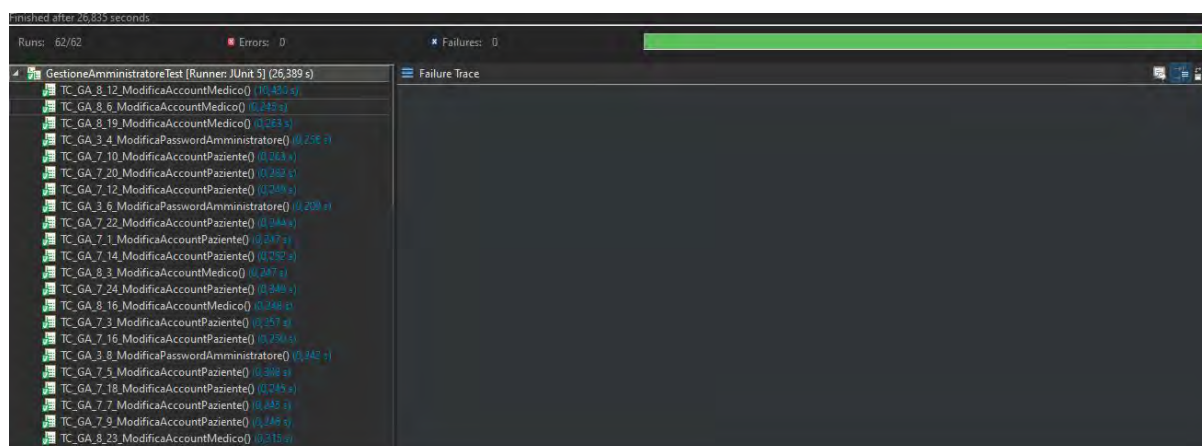
- TC\_GA\_8\_15\_ModificaAccountMedico() è un metodo che si occupa di effettuare la modifica del medico con una mail troppo lunga per essere registrata. La failure rilevata è dovuta ad un controllo mancato nella servlet il che consente di registrare un paziente con una mail più lunga del normale.

```
org.opentest4j.AssertionFailedError: expected: <./ModificaAccountMedicoView.jsp?notifica=ParamErr> but was: <./dashboard.jsp?notifica=ModificaMedRiuscita>  
at TestControlFailureAmministratoreTest.TC_GA_8_17_ModificaAccountMedico() (GestioneAmministratoreTest.java:1010)
```

- TC\_GA\_8\_17\_ModificaAccountMedico() è un metodo che si occupa di effettuare la modifica del medico con un email già presente. La failure è dovuta al fatto che un medico non può essere registrato con una mail già presente nel database mentre la registrazione avviene poichè non è stato effettuato il controllo sulle mail presenti nel DB.

```
org.opentest4j.AssertionFailedError: expected: <./ModificaAccountMedicoView.jsp?notifica=ParamErr> but was: <./dashboard.jsp?notifica=ModificaMedRiuscita>  
at TestControlFailureAmministratoreTest.TC_GA_8_17_ModificaAccountMedico() (GestioneAmministratoreTest.java:1010)
```

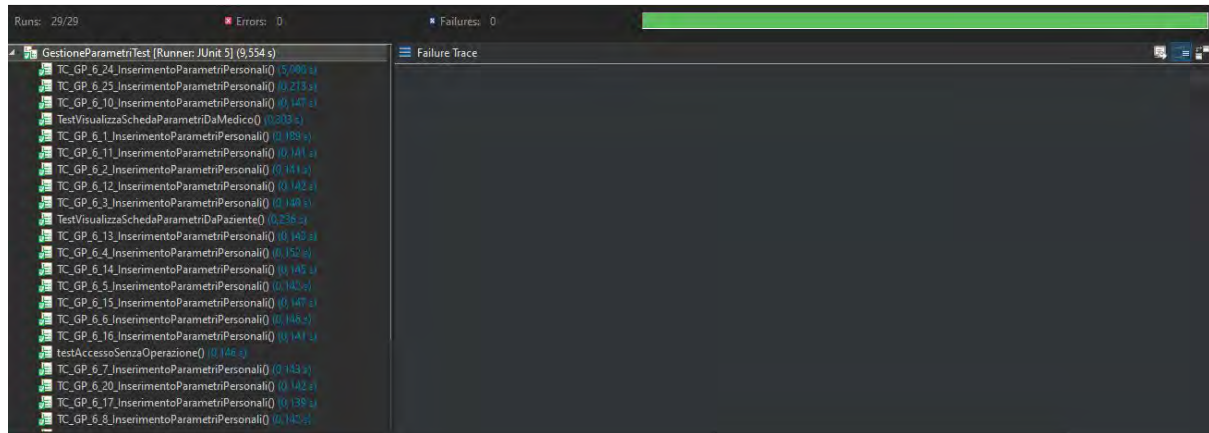
### 2.3.3 GestioneAmministratoreTest (Success)





L'esecuzione della classe di test GestioneAmministratoreTest ha eseguito tutti i 62 casi di test riportati nello screen e tutti hanno avuto successo dopo la correzione degli errori rilevati.

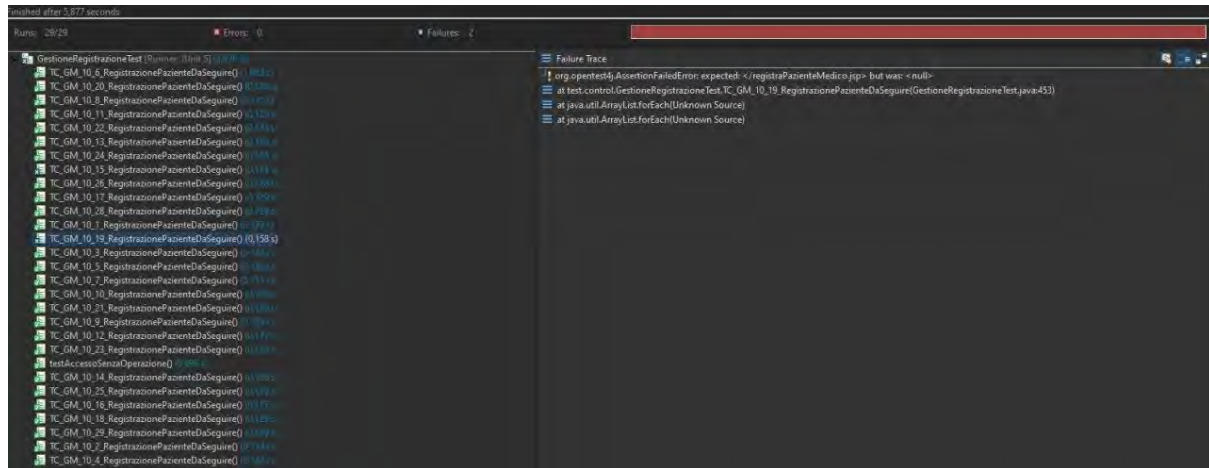
#### 2.3.4 GestioneParametriTest (rif. TC\_GM\_6, TC\_GP\_6, TC\_GP\_7) (Success)



L'esecuzione della classe di test GestioneParametriTest ha eseguito tutti i 29 casi di test riportati nello screen e tutti hanno avuto successo.

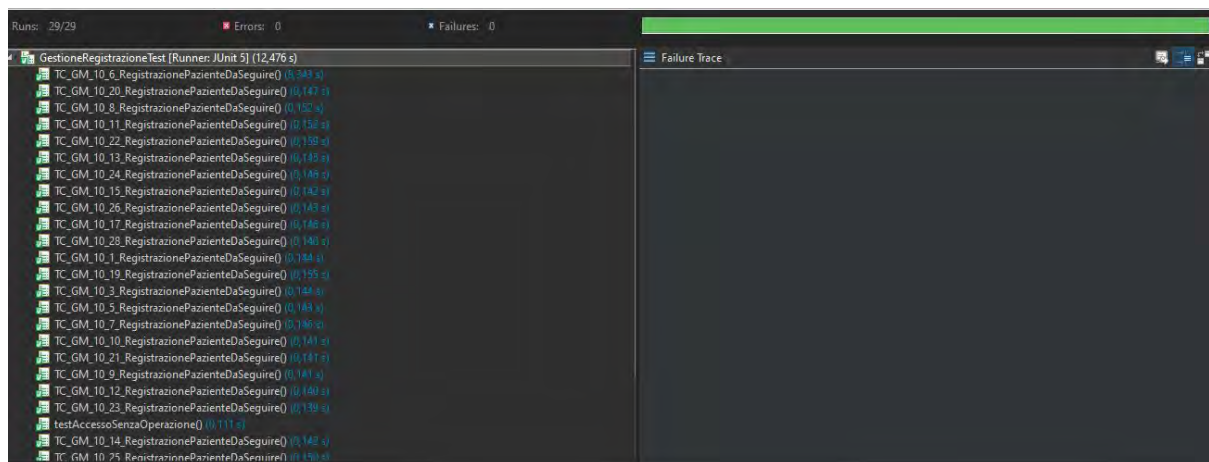


### 2.3.5 GestioneRegistrazioneTest (rif. TC\_GM\_10) (Failure)



Il test di GestioneRegistrazione ha avuto una failure. In particolare, la failure è stata rilevata in TC\_GM\_10\_19\_RegistrazionePazienteDaSeguire(), ed è dovuta ad un mancato controllo sulla lunghezza della mail.

### 2.3.6 GestioneRegistrazioneTest (Success)



L'esecuzione della classe di test GestioneRegistrazioneTest ha eseguito tutti i 29 casi di test riportati nello screen e tutti hanno avuto successo dopo la correzione del bug in TC\_GM\_10\_19\_RegistrazionePazienteDaSeguire().



### 2.3.7 GestionePazienteTest (rif. TC\_GP\_4) (Failure)

```
Finished after 5,241 seconds
Runs: 26/26      Errors: 0      Failures: 1

GestionePazienteTest [Runner: JUnit 5] (4,381 s)
  TC_GP_4_9_ModificaAccountPaziente() (0,142 s)
  testDisattivaAccount() (0,132 s)
  TC_GP_4_10_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_20_ModificaAccountPaziente() (0,088 s)
  TC_GP_4_12_ModificaAccountPaziente() (0,082 s)
  TC_GP_4_22_ModificaAccountPaziente() (0,084 s)
  TC_GP_4_14_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_24_ModificaAccountPaziente() (0,119 s)
  TC_GP_4_16_ModificaAccountPaziente() (0,081 s)
  TC_GP_4_2_ModificaAccountPaziente() (0,092 s)
  TC_GP_4_18_ModificaAccountPaziente() (0,091 s)
  TC_GP_4_4_ModificaAccountPaziente() (0,081 s)
  TC_GP_4_6_ModificaAccountPaziente() (0,089 s)
  TC_GP_4_8_ModificaAccountPaziente() (0,093 s)
  testVisualizzaProfilo() (0,086 s)
  TC_GP_4_11_ModificaAccountPaziente() (0,082 s)
  TC_GP_4_21_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_13_ModificaAccountPaziente() (0,084 s)
  TC_GP_4_23_ModificaAccountPaziente() (0,081 s)
  TC_GP_4_15_ModificaAccountPaziente() (0,174 s)
  TC_GP_4_1_ModificaAccountPaziente() (0,088 s)
  TC_GP_4_17_ModificaAccountPaziente() (0,084 s)
  TC_GP_4_3_ModificaAccountPaziente() (0,091 s)
  TC_GP_4_19_ModificaAccountPaziente() (0,087 s)
  TC_GP_4_5_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_7_ModificaAccountPaziente() (0,101 s)

Failure Trace
  org.opentest4j.AssertionFailedError: expected: <_/ModificaAccountPazienteView.jsp?notifica=ParamErr> but was: <_/ModificaAccountPazienteView.jsp?notifica=ParamErr>
    at test.control.GestionePazienteTest.TC_GP_4_15_ModificaAccountPaziente(GestionePazienteTest.java:369)
    at java.util.ArrayList.forEach(Unknown Source)
    at java.util.ArrayList.forEach(Unknown Source)
```

Il test di GestionePaziente ha avuto una failure dovuto al controllo sulla mail. In particolare il fail in TC\_GP\_4\_15\_ModificaAccountPaziente() è dovuto ad un mancato controllo sulla lunghezza della mail.

#### 2.3.7.1 GestionePazienteTest (Success)

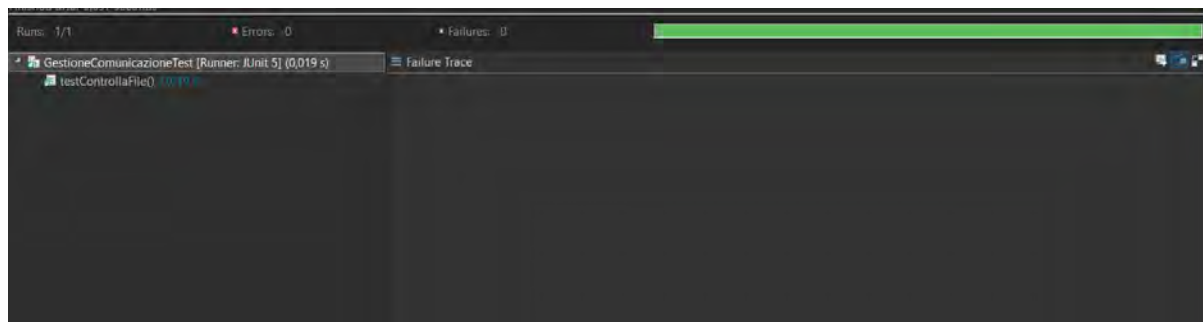
```
Runs: 26/26      Errors: 0      Failures: 0

GestionePazienteTest [Runner: JUnit 5] (3,967 s)
  TC_GP_4_9_ModificaAccountPaziente() (0,143 s)
  testDisattivaAccount() (0,138 s)
  TC_GP_4_10_ModificaAccountPaziente() (0,084 s)
  TC_GP_4_20_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_12_ModificaAccountPaziente() (0,089 s)
  TC_GP_4_22_ModificaAccountPaziente() (0,085 s)
  TC_GP_4_14_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_24_ModificaAccountPaziente() (0,167 s)
  TC_GP_4_16_ModificaAccountPaziente() (0,087 s)
  TC_GP_4_2_ModificaAccountPaziente() (0,099 s)
  TC_GP_4_18_ModificaAccountPaziente() (0,084 s)
  TC_GP_4_4_ModificaAccountPaziente() (0,081 s)
  TC_GP_4_6_ModificaAccountPaziente() (0,089 s)
  TC_GP_4_8_ModificaAccountPaziente() (0,081 s)
  testVisualizzaProfilo() (0,085 s)
  TC_GP_4_11_ModificaAccountPaziente() (0,082 s)
  TC_GP_4_21_ModificaAccountPaziente() (0,081 s)
  TC_GP_4_13_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_23_ModificaAccountPaziente() (0,089 s)
  TC_GP_4_15_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_1_ModificaAccountPaziente() (0,080 s)
  TC_GP_4_17_ModificaAccountPaziente() (0,081 s)
  TC_GP_4_3_ModificaAccountPaziente() (0,081 s)
  TC_GP_4_19_ModificaAccountPaziente() (0,077 s)
  TC_GP_4_5_ModificaAccountPaziente() (0,088 s)
  TC_GP_4_7_ModificaAccountPaziente() (0,089 s)
```

L'esecuzione della classe di test GestionePazienteTest ha eseguito tutti i 26 casi di test riportati nello screen e tutti hanno avuto successo dopo la correzione dell'errore in TC\_GP\_4\_15\_ModificaAccountPaziente().

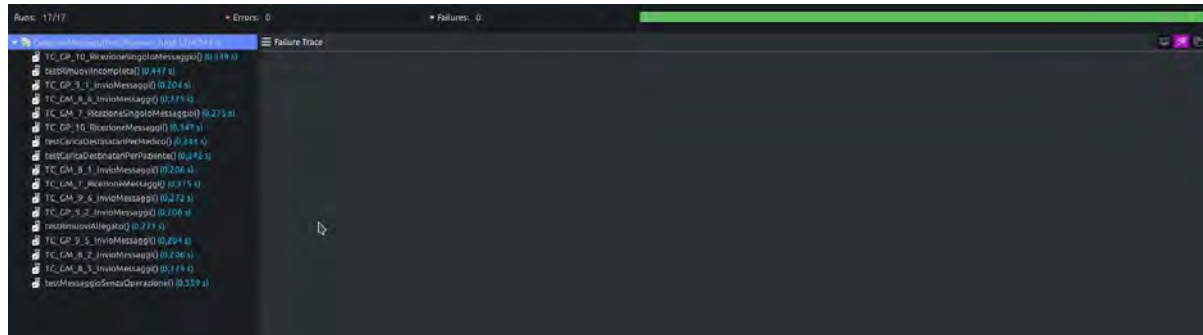


### 2.3.8 GestioneComunicazioneTest (Success)



L'esecuzione della classe di test GestioneComunicazioneTest ha eseguito il caso di test riportato nello screen e ha avuto successo.

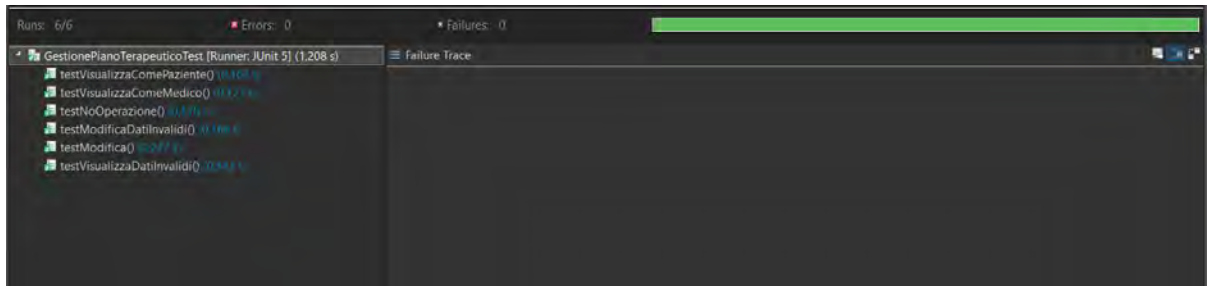
### 2.3.9 GestioneMessaggiTest (rif. TC\_GM\_7, TC\_GM\_8, TC\_GP\_9, TC\_GP\_10) (Success)



L'esecuzione della classe di test GestioneMessaggiTest ha eseguito tutti i 17 casi di test riportati nello screen e tutti hanno avuto successo.

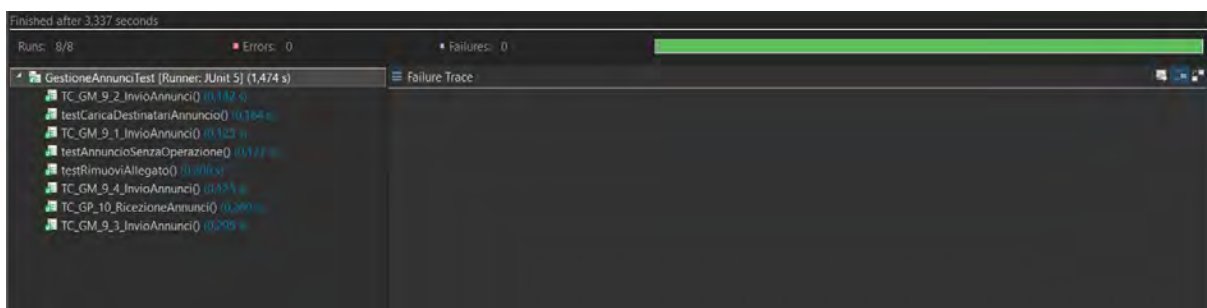


### 2.3.10 GestionePianoTerapeuticoTest (rif. TC\_GM\_11, TC\_GP\_8) (Success)



L'esecuzione della classe di test GestionePianoTerapeuticoTest ha eseguito tutti i 6 casi di test riportati nello screen e tutti hanno avuto successo.

### 2.3.11 GestioneAnnunciTest (rif. TC\_GM\_9, TC\_GP\_10) (Success)



L'esecuzione della classe di test GestioneAnnunciTest ha eseguito tutti i 8 casi di test riportati nello screen e tutti hanno avuto successo.