



Test Summary Report

E-Balance

Riferimento	E-Balance_C17_TSR
Versione	0.1
Data	06/02/2024
Destinatario	F. Ferrucci, F. Palomba
Presentato da	Simone Cirma, Mario De Luca, Antonio Di Giorgio, Donato Folgieri, Daniela Palma, Emanuele Vitale
Approvato da	Matteo Ercolino, Simone Silvestri



Revision History

Data	Versione	Descrizione	Autori
06/02/2024	0.1	Prima stesura del documento	Tutti i team member



Sommario

Revision History	2
1. Introduzione	4
2. Relazione con altri documenti	4
3. Testing Unitario e di Integrazione	4
4. Testing di sistema	5



1. Introduzione

E-Balance si propone di migliorare la gestione energetica dell'Università degli Studi di Salerno, creando un sistema in grado di monitorare e ottimizzare l'utilizzo delle risorse energetiche con l'aiuto dell'intelligenza artificiale.

Il documento di Test Plan ha l'obiettivo di descrivere ed analizzare le attività di Testing per il software di E-Balance. Il fine è quello di garantire che ogni aspetto funzioni in modo corretto. All'interno del documento sono riportate le strategie di testing adottate, quali funzionalità saranno testate e gli strumenti scelti per la rilevazione degli errori, con lo scopo di presentare un sistema privo di malfunzionamenti.

Sono state pianificate attività di testing per le seguenti gestioni:

- Gestione Amministratore
- Gestione Contratto
- Gestione Accesso
- Gestione IA

2. Relazione con altri documenti

Per la corretta esecuzione dei test di sistema progettati, si fa riferimento ad altri documenti prodotti:

- Relazioni con il Test Plan ([TP](#)): In esso sono state specificate le varie modalità di esecuzione dei vari test.
- Relazioni con il Test Case Specification ([TCS](#)): In esso sono descritti i vari test da effettuare.
- Relazioni con il Test Incident Report ([TIR](#)): In esso è rappresentato, sottoforma tabellare in foglio di calcolo, il report relativo ai risultati dell'esecuzioni dei test specificati nel TCS e TP.

3. Testing Unitario e di Integrazione

I test unitari e di integrazione sono stati descritti ed effettuati per ogni classe presente all'interno del sistema in modo tale da garantire il corretto funzionamento di ogni componente. Durante la fase di test, ogni team member, dopo aver compilato la classe di test ne effettuava il commit solo dopo averne controllato il corretto funzionamento.

Per la raccolta delle informazioni relative al branch coverage è stato utilizzato il tool JaCoCo.

Branche Coverage	Line Coverage
43%	52%



4. Testing di sistema

Per ciò che concerne il test di sistema si sono andate a definire vari test suites tramite il tool Selenium IDE, per Chrome. Nello specifico, si è generata una Test Suite per Gestione e ad ogni esecuzione di una test suite si effettuava il riavvio del sistema e refresh del DataBase.

Di seguito vengono riportati i risultati delle esecuzioni dei test:

Esecuzione	#Fallimenti	#Successi
Esecuzione 1 06/02/2024	0	33