



Laurea Magistrale in informatica
Università degli Studi di Salerno

Test Incident Report

Post-CR

QuantuMoonLight

Team

Matteo Cicalese - 0522501516
Luca Contrasto – 0522501512

Repo

github.com/CicaMatt/QML-IGES
github.com/Robertales/QuantuMoonLight

Sommario

- 1. Introduzione 3
- 2. Test di sistema aggiunti 3
- 3. Test di integrazione/unità aggiunti 3
- 4. Incident Report..... 4
 - 4.1. Incident Report UTI1 4
 - 4.2. Incident Report UTI2 5

1. Introduzione

Nel presente report saranno dettagliati i difetti e le problematiche riscontrate durante l'esecuzione dei casi di test a livello di unità, integrazione e sistema, riguardanti le modifiche introdotte.

L'obiettivo del documento è quindi fornire un'analisi dei fault emersi durante i test delle change request applicate al sistema e di illustrare le azioni risolutive intraprese al fine di risolverli.

2. Test di sistema aggiunti

- Test suite contenuta in *test_download*
- Test suite contenuta in *test_password_reset*

3. Test di integrazione/unità aggiunti

- *Test_UtenteControl*
 - *Test_SetNewPW*
 - *Test_sendCode*
 - *Test_sendCode_emailNotFoundError*
 - *Test_email_sending_error*
 - *Test_Download*
 - *testDownload_invalidPath*
- *Test_utils*
 - *Test_encrypt*
 - *Test_decrypt*
 - *Test_encrypt_fail*
 - *Test_decrypt_fail*

La specifica dei test con relativi input e oracolo atteso è presente nei documenti **Unit Test Plan** e **Post-CR System Testing**.

4. Incident Report

4.1. Incident Report UTI1

- **Informazioni:**

- Progetto: "QuantuMoonLight"
- Ambiente: Ambiente di test
- Nome Incident: UTI1 (Unit Testing Incident 1)
- Test Case: *test_routes1, test_routes2, test_routes3, test_routes4*
- Priorità: Alta
- Gravità: Bassa

- **Descrizione dell'Incidente:**

- Descrizione: Durante il testing di unità del sistema abbiamo riscontrato il seguente problema:

I test specificati risultano essere fleaky in quanto gli assert presenti non sempre sono corretti, nonostante vengano immessi gli stessi input.

- Passi per Riprodurre:

- Eseguire i test specificati diverse volte

- Comportamento Attuale: i test a volte falliscono e a volte hanno successo.

- **Risoluzione e Pianificazione:**

- Stato: chiuso

- Cause Identificate: i test a volte falliscono in quanto gli assert specificati controllano dei file generati all'interno di un thread, per cui il risultato del test dipende esclusivamente dal termine del thread(flakiness)

- Componente Coinvolto: *UtenteControl.py*

- Pianificazione: la modifica è minima e coinvolge una singola classe quindi non riteniamo di dover eseguire Impact Analysis.

- Soluzione: sono state inserite le istruzioni, necessarie solo all'interno dei metodi di test, utili ad attendere la fine del thread di interesse attivo in modo da controllare correttamente i file generati.

4.2. Incident Report UTI2

- **Informazioni:**

- Progetto: "QuantuMoonLight"
- Ambiente: Ambiente di test
- Nome Incident: UTI2 (Unit Testing Incident 2)
- Test Case: *test_sendCode*
- Priorità: Alta
- Gravità: Critica

- **Descrizione dell'Incidente:**

○ Descrizione: Durante il testing di sistema volto a testare l'effettiva correttezza della change request che riguarda il reset della password, abbiamo riscontrato il seguente problema:

Durante l'esecuzione del test che riguarda l'invio del codice di recupero via email il sistema si blocca e termina

○ Passi per Riprodurre:

- Disconnettere il pc dalla rete
- Andare nella pagina di login
- Cliccare su "Forgot your Password?"
- Inserire una mail esistente nel database
- Cliccare su "SEND"

○ Comportamento Attuale: l'applicazione va in stato d'errore e smette di funzionare

- **Risoluzione e Pianificazione:**

○ Stato: chiuso

○ Cause Identificate: non è presente un controllo riguardo eventuali stati d'errore dovuti per esempio a connessione assente o errori di tipo generale esterni all'applicazione per cui nel caso in cui si verifica un'eccezione del genere il sistema smette di funzionare.

○ Componente Coinvolto: *UtenteControl*

○ Pianificazione: la modifica è minima e coinvolge una singola classe quindi non riteniamo di dover eseguire Impact analysis.

○ Soluzione: viene inserito un controllo sull'eventuale ricezione di questo tipo di eccezione identificata con id 500 e nel caso venga lanciata il sistema mostra

un messaggio d'errore a video chiedendo di ripetere l'operazione. Inoltre, è stato aggiunto un ulteriore caso di test che verifica questo stato del sistema.