



Laurea Magistrale in informatica
Università di Salerno

Test Incident Report

Pre-Unit and Integration

Testing

Team member

Saverio Napolitano - 0522501400

Gerardo Festa - 0522501452

Alessandra Parziale - 0522501422

<https://github.com/GerardoFesta/infozilla>

Sommario

1. Introduzione.....	3
2. Incident Report.....	3
2.1. Incident Report STI1.....	3
2.2. Incident Report STI2.....	4
2.3. Incident Report STI3.....	5
2.4. Incident Report STI4.....	6

1. Introduzione

Nel presente report saranno dettagliati i difetti e le problematiche riscontrate nell'ambito dell'esecuzione dei casi di test.

L'obiettivo principale di questo documento è quello di fornire un'analisi dei fault emersi durante il processo di test di sistema preliminare.

Attraverso una valutazione dettagliata dei problemi riscontrati, si mira a migliorare il processo complessivo di sviluppo.

2. Incident Report

2.1. Incident Report STI1

- **Informazioni di Base:**
 - Progetto: "InfoZilla".
 - Ambiente: Ambiente di test.
 - Nome Incident: STI1 (System Testing Incident 1).
 - Test Case: 7/8/9/10.
 - Priorità: Alta.
 - Gravità: Critica.
- **Descrizione dell'Incidente:**
 - Descrizione: Durante il testing preliminare di sistema volto a comprenderne il funzionamento, abbiamo riscontrato il seguente problema:
Settando a false almeno uno dei parametri (*withStackTraces*, *whithPatches*, *whithCode*, *whithLists*) il sistema riporta una *NullPointerException*.
 - Passi per Riprodurre:
 1. Settare a false almeno uno dei parametri (*withStackTraces*, *whithPatches*, *whithCode*, *whithLists*).
 2. Eseguire il tool a linea di comando.
 - Comportamento Attuale: L'applicazione riporta una *NullPointerException*.
- **Risoluzione e Pianificazione:**
 - Stato: chiuso.
 - Cause Identificate: Il problema si trova nel costruttore di *FilterChainEclipse* che richiama il metodo *getOutputText* anche per i filtri non utilizzati. Senza l'esecuzione del metodo *runFilter* la variabile *textRemover* non è istanziata e il *getOutputText* la utilizza facendo riferimento a null.
 - Componente Coinvolto: *Filters*.

- Pianificazione: la modifica è minima e coinvolge una singola classe quindi non riteniamo di dover eseguire Impact analysis.
- Soluzione: La chiamata a *getOutputText* viene spostata all'interno dell'if che valuta se il filtro deve essere eseguito o meno.
Quindi la chiamata viene fatta solo se il filtro viene eseguito, risolvendo la *NullPointerException*.

2.2. Incident Report STI2

- **Informazioni di Base:**

- Progetto: "InfoZilla".
- Ambiente: Ambiente di test.
- Nome Incident: STI2 (System Testing Incident 2).
- Test Case: 7/9/10/11.
- Priorità: Alta.
- Gravità: Critica.

- **Descrizione dell'Incidente:**

- Descrizione: Durante il testing preliminare di sistema volto a comprenderne il funzionamento, abbiamo riscontrato il seguente problema:
Il codice presente nelle patches viene riportato anche all'interno del source code questo è un comportamento inatteso data la struttura dei filtri che una volta individuati gli elementi li eliminano.
È causa del warning *IndexForElimination* out of bounds.
- Passi per Riprodurre:
 1. Creare un file di testo con delle patch.
 2. Eseguire il tool a linea di comando.
 3. Verificare l'output di "*Patches*" e "*Source Code Fragments*".
- Comportamento Attuale: L'applicazione riporta valori errati in "*Source Code Fragments*".

- **Risoluzione e Pianificazione:**

- Stato: Aperto.
- Cause Identificate: Il problema è causato dall'individuazione della patch poiché l'inizio non viene segnalato correttamente.
- Componente Coinvolto: *PatchParser*.
- Pianificazione: la modifica potrebbe coinvolgere più classi quindi riteniamo di dover eseguire Impact analysis.

- Soluzione: un possibile approccio che pensiamo di adottare consiste nell'analizzare il metodo *parseForPatches* della classe *PatchParser* per individuare l'errore e risolverlo.

2.3. Incident Report STI3

- **Informazioni di Base:**

- Progetto: "InfoZilla".
- Ambiente: Ambiente di test.
- Nome Incident: STI3 (System Testing Incident 3).
- Test Case: 7/9/10/11.
- Priorità: Alta.
- Gravità: Critica.

- **Descrizione dell'Incidente:**

- Descrizione: Durante il testing preliminare di sistema volto a comprenderne il funzionamento, abbiamo riscontrato il seguente problema:
Il codice degli hunks all'interno delle patches non viene riconosciuto del tutto in quanto si ferma al primo spazio e viene riportato con degli "r" in formato anomalo.
- Passi per Riprodurre:
 1. Creare un file di testo con degli hunks all'interno delle patches intervallati da spazi.
 2. Eseguire il tool a linea di comando.
 3. Verificare il file xml di output.
- Comportamento Attuale: L'applicazione non riconosce gli hunks in modo adeguato.

- **Risoluzione e Pianificazione:**

- Stato: Aperto.
- Cause Identificate: Il problema è causato dall'individuazione degli hunks poiché si ferma al primo spazio.
- Componente Coinvolto: *PatchParser*.
- Pianificazione: la modifica potrebbe coinvolgere più classi quindi riteniamo di dover eseguire Impact analysis.
- Soluzione: un possibile approccio che pensiamo di adottare consiste nell'iniziare ad analizzare il metodo *findAllHunks* e verificare se coinvolge altri metodi della classe *PatchParser* per individuare l'errore e risolverlo.

2.4. Incident Report STI4

- **Informazioni di Base:**
 - Progetto: "InfoZilla".
 - Ambiente: Ambiente di test.
 - Nome Incident: STI4 (System Testing Incident 4).
 - Test Case: 11.
 - Priorità: Alta.
 - Gravità: Critica.
- **Descrizione dell'Incidente:**
 - Descrizione: Durante il testing preliminare di sistema volto a comprenderne il funzionamento, abbiamo riscontrato il seguente problema:
Il formato delle patches riconosciuto del sistema è *Unified Diff Format with Index Line*, tuttavia non si tratta del formato più diffuso.
 - Passi per Riprodurre:
 1. Creare un file con delle patches con formato differente da *Unified Diff Format with Index Line*.
 2. Eseguire il tool a linea di comando.
 3. Verificare l'output prodotto.
 - Comportamento Attuale: L'applicazione non riconosce un formato differente da *Unified Diff Format with Index Line*.
- **Risoluzione e Pianificazione:**
 - Stato: Aperto.
 - Cause Identificate: Il problema è causato dalla regex utilizzata dal *PatchParser* per riconoscere una patch.
 - Componente Coinvolto: *PatchParser*.
 - Pianificazione: la modifica potrebbe coinvolgere più classi quindi riteniamo di dover eseguire Impact analysis.
 - Soluzione: un possibile approccio che pensiamo di adottare consiste nel modificare la regex utilizzata dal *PatchParser* per riconoscere anche il formato *Unified Diff Format* (quello che si ottiene da git facendo un diff).