



UNIVERSITÀ DEGLI STUDI DI SALERNO

ANNO ACCADEMICO 2019/2020



UNIT TEST REPORT DOCUMENT

Version 2.0

TOP MANAGER

Prof. Andrea De Lucia

PROJECT MANAGER

Giuseppe De Michele

Partecipanti

Nome	Matricola
Mario Balbi	0512102944
Giuseppe De Michele	0512102642
Singh Karanbir	0512104924



Revision History

Data	Versione	Descrizione	Autore
17/02/2020	1.0	Introduzione; Relazione con altri documenti; Dettagli; Dettagli dei risultati dei test; Glossario;	De Michele Giuseppe



Indice

1. Introduzione	4
1. 1 Definizioni ed acronimi	4
2. Relazione con altri documenti	4
3. Dettagli	4
1. Overview dei risultati dei test	4
2. Dettagli dei risultati dei test	5
4. Conclusioni e raccomandazioni	7
5. Glossario	8



1. Introduzione

1.1 Definizioni ed acronimi

Il testing di unità rappresenta la fase di testing in cui si assicura che le componenti sviluppate funzionino in isolamento. Questo documento ha il compito di identificare la strategia di testing di unità per il sistema Nashira Custom Guitars. In particolare, saranno specificate le componenti da testare e il modo in cui il testing dovrà essere eseguito. Lo scopo del documento è quello di riportare l'esecuzione del testing unitario.

2. Relazione con altri documenti

Per verificare il corretto funzionamento del sistema Nashira Custom Guitars saranno utilizzati i test cases. I test cases sono basati sulle funzionalità individuate nel documento di raccolta ed analisi dei requisiti (RAD).

I documenti a cui facciamo riferimento sono:

C2_RAD: Requirement Analysis document

C2_SDD: System Design Document

C2_ODD: Object Design Document

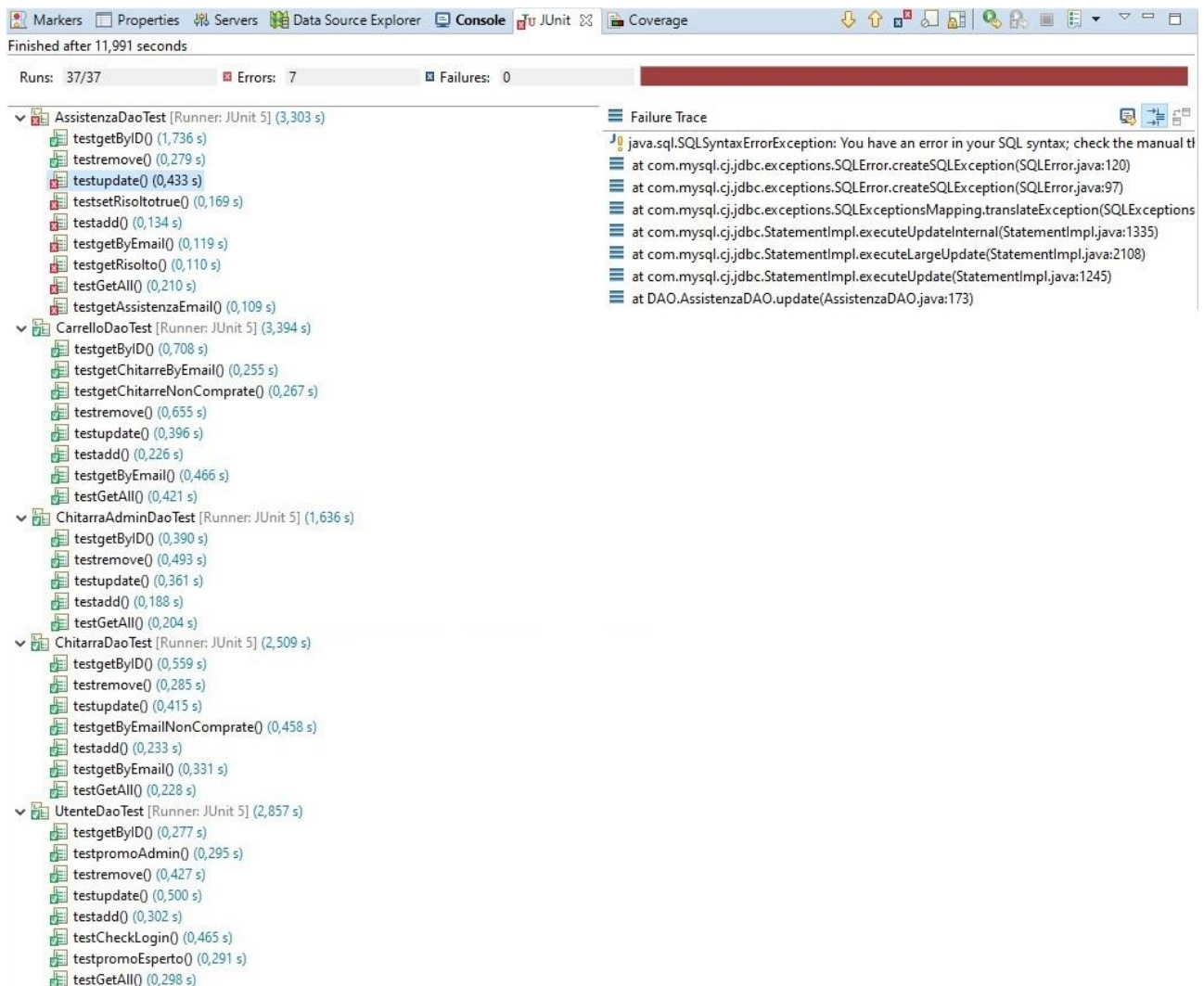
3. Details

3.1. Overview dei risultati dei test

Non sono stati riscontrati fault nelle componenti del sistema Nashira Custom Guitars eccetto per l'inserimento nella tabella Assistenza del DB . Per informazioni dettagliate sugli item testati e sui risultati si faccia riferimento al paragrafo successivo.

3.2 Dettagli dei risultati dei test

3.2.1 Fault durante il test



Markers Properties Servers Data Source Explorer Console JUnit Coverage

Finished after 11,991 seconds

Runs: 37/37 Errors: 7 Failures: 0

AssistenzaDaoTest [Runner: JUnit 5] (3,303 s)

- testGetById() (1,736 s)
- testRemove() (0,279 s)
- testUpdate() (0,433 s)
- testSetRisoltoTrue() (0,169 s)
- testAdd() (0,134 s)
- testGetByEmail() (0,119 s)
- testGetRisolto() (0,110 s)
- testGetAll() (0,210 s)
- testGetAssistenzaEmail() (0,109 s)

CarrelloDaoTest [Runner: JUnit 5] (3,394 s)

- testGetById() (0,708 s)
- testGetChitarraByEmail() (0,255 s)
- testGetChitarraNonComprate() (0,267 s)
- testRemove() (0,655 s)
- testUpdate() (0,396 s)
- testAdd() (0,226 s)
- testGetByEmail() (0,466 s)
- testGetAll() (0,421 s)

ChitarraAdminDaoTest [Runner: JUnit 5] (1,636 s)

- testGetById() (0,390 s)
- testRemove() (0,493 s)
- testUpdate() (0,361 s)
- testAdd() (0,188 s)
- testGetAll() (0,204 s)

ChitarraDaoTest [Runner: JUnit 5] (2,509 s)

- testGetById() (0,559 s)
- testRemove() (0,285 s)
- testUpdate() (0,415 s)
- testGetByEmailNonComprate() (0,458 s)
- testAdd() (0,233 s)
- testGetByEmail() (0,331 s)
- testGetAll() (0,228 s)

UtenteDaoTest [Runner: JUnit 5] (2,857 s)

- testGetById() (0,277 s)
- testPromoAdmin() (0,295 s)
- testRemove() (0,427 s)
- testUpdate() (0,500 s)
- testAdd() (0,302 s)
- testCheckLogin() (0,465 s)
- testPromoEsperto() (0,291 s)
- testGetAll() (0,298 s)

Failure Trace

```

java.sql.SQLException: You have an error in your SQL syntax; check the manual t
at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:120)
at com.mysql.cj.jdbc.exceptions.SQLException.createSQLException(SQLException.java:97)
at com.mysql.cj.jdbc.exceptions.SQLExceptionsMapping.translateException(SQLExceptions
at com.mysql.cj.jdbc.StatementImpl.executeUpdateInternal(StatementImpl.java:1335)
at com.mysql.cj.jdbc.StatementImpl.executeLargeUpdate(StatementImpl.java:2108)
at com.mysql.cj.jdbc.StatementImpl.executeUpdate(StatementImpl.java:1245)
at DAO.AssistenzaDAO.update(AssistenzaDAO.java:173)

```

























Il test di unità ha riscontrato un fault nella query update della tabella Assistenza del DB, abbiamo quindi provveduto a modificare la query assicurando il corretto funzionamento del sistema .



3.2.2 Test completato senza fault

The screenshot displays the JUnit test runner interface within an IDE. The top toolbar includes icons for Markers, Properties, Servers, Data Source Explorer, Console, JUnit, and Coverage. Below the toolbar, a status bar indicates "Finished after 14,748 seconds". A progress bar shows "Runs: 37/37", "Errors: 0", and "Failures: 0". The main test results pane lists several test classes, each with a list of individual test methods and their execution times. All tests passed successfully, indicated by green checkmarks.

- AssistenzaDaoTest [Runner: JUnit 5] (4,120 s)**
 - testGetById() (1,544 s)
 - testremove() (0,422 s)
 - testupdate() (0,465 s)
 - testsetRisoltotruo() (0,300 s)
 - testadd() (0,302 s)
 - testgetByEmail() (0,185 s)
 - testgetRisolto() (0,288 s)
 - testGetAll() (0,274 s)
 - testgetAssistenzaEmail() (0,337 s)
- CarrelloDaoTest [Runner: JUnit 5] (3,394 s)**
 - testGetById() (0,708 s)
 - testgetChitarraByEmail() (0,255 s)
 - testgetChitarraNonComprato() (0,267 s)
 - testremove() (0,655 s)
 - testupdate() (0,396 s)
 - testadd() (0,226 s)
 - testgetByEmail() (0,466 s)
 - testGetAll() (0,421 s)
- ChitarraAdminDaoTest [Runner: JUnit 5] (1,636 s)**
 - testGetById() (0,390 s)
 - testremove() (0,493 s)
 - testupdate() (0,361 s)
 - testadd() (0,188 s)
 - testGetAll() (0,204 s)
- ChitarraDaoTest [Runner: JUnit 5] (2,509 s)**
 - testGetById() (0,559 s)
 - testremove() (0,285 s)
 - testupdate() (0,415 s)
 - testgetByEmailNonComprato() (0,458 s)
 - testadd() (0,233 s)
 - testgetByEmail() (0,331 s)
 - testGetAll() (0,228 s)
- UtenteDaoTest [Runner: JUnit 5] (2,857 s)**
 - testGetById() (0,277 s)
 - testpromoAdmin() (0,295 s)
 - testremove() (0,427 s)
 - testupdate() (0,500 s)
 - testadd() (0,302 s)
 - testCheckLogin() (0,465 s)
 - testpromoEsperto() (0,291 s)
 - testGetAll() (0,298 s)

New_configuration (1) (17 feb 2020 18:05:54)				
Element	Coverage	Covered Instructio...	Missed Instructions	Total Instructions
▼ tsw-master	 79,0 %	5.525	1.472	6.997
▼ src/java	 68,2 %	3.164	1.472	4.636
> Servlet	 0,0 %	0	809	809
▼ Bean	 47,4 %	410	455	865
> ListaCarrelli.java	 6,0 %	8	125	133
> Utente.java	 46,6 %	108	124	232
> Chitarra.java	 51,6 %	113	106	219
> Carrello.java	 49,0 %	73	76	149
> Assistenza.java	 68,4 %	52	24	76
> ChitarraAdmin.java	 100,0 %	56	0	56
> database	 26,3 %	74	207	281
▼ DAO	 100,0 %	2.680	1	2.681
> UtenteDAO.java	 99,8 %	580	1	581
> AssistenzaDAO.java	 100,0 %	573	0	573
> CarrelloDAO.java	 100,0 %	556	0	556
> ChitarraAdminDAO.java	 100,0 %	327	0	327
> ChitarraDAO.java	 100,0 %	644	0	644
▼ test	 100,0 %	2.361	0	2.361
▼ (default package)	 100,0 %	2.361	0	2.361
> AssistenzaDaoTest.java	 100,0 %	517	0	517
> CarrelloDaoTest.java	 100,0 %	679	0	679
> ChitarraAdminDaoTest.java	 100,0 %	273	0	273
> ChitarraDaoTest.java	 100,0 %	473	0	473
> UtenteDaoTest.java	 100,0 %	419	0	419

L'Integration Testing e il System Testing sono test che mirano a correggere fault a livello di componenti e di sistema. Nel caso in cui fossero presenti dei fault nel codice (e quindi uno o più metodi della classe fossero stati realizzati in maniera scorretta), questi sarebbero difficilmente individuabili durante l'IT, poiché non vengono testati i singoli metodi.

4 Conclusioni e raccomandazioni

I risultati ottenuti mostrano che non ci sono errori. Si può passare alla fase di testing successiva.

5 Glossario

Amministratore	Il termine indica una particolare tipologia di utente che si occupa dell'amministrazione del sistema e degli altri utenti.
Assistenza	Il termine identifica una richiesta di assistenza inviata nel sistema.
Chitarra	Il termine indica una chitarra pubblicata nel sistema.
DAO	Il Data Access Object è un pattern architetturale per la gestione dei dati persistenti.
Esperto	Il termine indica una particolare tipologia di utente che si occupa della gestione dell'assistenza.
Nashira Custom Guitars	Nome del sistema che verrà sviluppato
ODD	Il documento di Object Design chiude il gap tra oggetti di applicazione e componenti off-the-shelf identificando oggetti di soluzione e raffinando gli oggetti esistenti.
Personalizzazione	Il termine indica una personalizzazione allegata ad una delle chitarre
RAD	Il documento dell'analisi dei requisiti è un'attività preliminare allo sviluppo di un sistema software. Lo scopo principale di tale documento è di definire le funzionalità del sistema.
SDD	Il documento del System Design è un'attività preliminare allo sviluppo di un sistema software. Gli scopi del documento sono quelli di definire gli obiettivi di progettazione del sistema e di decomporre il sistema in sottosistemi più piccoli.
UNISA	Università degli Studi di Salerno.
Utente	Il termine indica l'utente generico che usufruisce del sistema