

# Ingegneria, Gestione ed Evoluzione del Software

## Master Test Plan Social Notes



<b>Luigi Allocca</b>	<b>0522501559</b>	<a href="mailto:l.allocca8@studenti.unisa.it">l.allocca8@studenti.unisa.it</a>
<b>Rocco Iuliano</b>	<b>0522501533</b>	<a href="mailto:r.iuliano13@studenti.unisa.it">r.iuliano13@studenti.unisa.it</a>
<b>Simone Della Porta</b>	<b>0522501534</b>	<a href="mailto:s.dellaporta6@studenti.unisa.it">s.dellaporta6@studenti.unisa.it</a>

## 1 Approccio e strategia di testing

Verrà utilizzato l'approccio *Bottom-up* per la fase di testing. Quindi verranno eseguiti prima i test di unità e poi i test di sistema. Per la realizzazione dei test case di sistema e di unità si utilizzerà l'approccio *Black-box*.

Nel **test di unità**, si testerà singolarmente ogni DAO modificato dalle change requests utilizzando la strategia **Category Partition** per suddividere lo spazio degli input in categorie e creare classi di equivalenza chiamate scelte, consentendoci di creare una suite di test che copre tutti i casi di test per ciascun DAO. Inoltre, per il testing di unità, si utilizzerà la tecnica di *Test Driven Development* che consente di sviluppare prima i casi di test e successivamente sviluppare o modificare la componente in modo da evitare di introdurre potenziali failure nel sistema.

Il **test di integrazione** tra i DAO e le Servlet, invece, non verrà effettuato e passeremo direttamente al testing di sistema. Questa decisione è stata presa perché le Servlet e le JSP si trovano nello stesso livello di presentazione e le JSP utilizzano principalmente i servizi forniti dalle Servlet.

Per il **testing di sistema** abbiamo ritenuto che il Category Partition fosse sufficiente per coprire tutti i casi di test per le Servlet. In caso di problemi, riusciremo ad individuare facilmente il difetto poiché le JSP chiamano solo le Servlet e i DAO sono già stati testati. Per il **regression testing** si utilizzerà l'approccio *TEST-ALL* in quanto l'esecuzione di tutti i test esistenti non supera i 10 minuti e ci consente di ritestare tutte le funzionalità del sistema.

## 2 Pass/Fail Criteria

Il test si considera superato se l'output corrisponde all'output atteso presente nell'oracolo, il test si considera fallito se l'output non corrisponde all'output atteso presente nell'oracolo. Nel caso in cui verrà riscontrata una failure, bisognerà verificare a che tipo di fault è legata e successivamente si procederà alla sua correzione. Infine, sarà eseguita di nuovo la fase di testing per verificare che la modifica effettuata non abbia creato ulteriori failure.

## 3 Attività di testing

Il testing prevede le seguenti attività:

- pianificazione dei test case;
- creazione test case;
- esecuzione dei test;
- report dei test;
- gestione delle failure.

## 4 Testing materials

Gli strumenti necessari per svolgere le attività di testing sono:

- JUnit
- Selenium
- DBUnit
- Web server Apache Tomcat9 in locale
- DBMS MySQL
- Web Browser per effettuare le richieste al server

## 5 Relazione con altri documenti

Il master test plan è strettamente relato ai documenti di test esistenti:

- TP\_sistema;
- TP\_unità;
- TCS\_sistema;
- TCS\_unità.