Università degli Studi di Parma
Facoltà di Scienze MM. FF. NN.
Corso di Laurea in Informatica

Ingegneria del Software

La fase di Test

Giulio Destri

Scopo del modulo

Definire

Proprietà e strutturazione
del processo di Test
entro un progetto informatico

Argomenti

- Definire il processo di Test
- Caratteristiche importanti
- Linee guida
- Criteri di completezza

Definire il processo di Test

- Agendo sul software/sistema realizzato durante la fase di implementazione
- La fase di test o testing deve eliminare il maggior numero possibile di errori
- Per arrivare all'entrata in produzione con un sistema il più possibile privo di errori

Glossario per il testing

- Error o errore
- Fault
- Failure
- Verifica
- Controllo di validità
- Dati di test
- Grafo di copertura
- Classi di equivalenza

Cause e locazione degli errori

Fonti di errore	Fase iniziale dell'errore
Comunicazione mancante	Coordinamento del team
Requisiti cambiati	Analisi dei requisiti
Fretta	Progettazione
Membri del team	Costituzione del team, formazione
Codice mal documentato	Implementazione
Supporti d'impl. Insufficienti	Analisi e progettazione
Strumenti di sviluppo	Progett.(scelta) impl.(uso)
Progettazione tradotta male	Implementazione
Errore del programmatore	Implementazione

Contesto del test (test frame)

- Criteri per la definizione di riuscita e fallimento
- Criteri per inizio e fine di una fase di test
- Criteri per l'interruzione e ripresa dei test
- Prodotti da testare
- Metodi di comunicazione

Contesto del test (test frame) - 2

- Genere del test
- Configurazione della piattaforma dei test
- Problemi al momento dell'esecuzione e responsabilità della loro eliminazione

Strategie di test

- Scopo dei test
- Prorità di qualità
- Obiettivi di test

Oggetto e casi di test

- Unità da testare (definizione della granularità)
- Proprietà da testare
- Proprietà che non vengono testate
- Programmazione temporale dei test

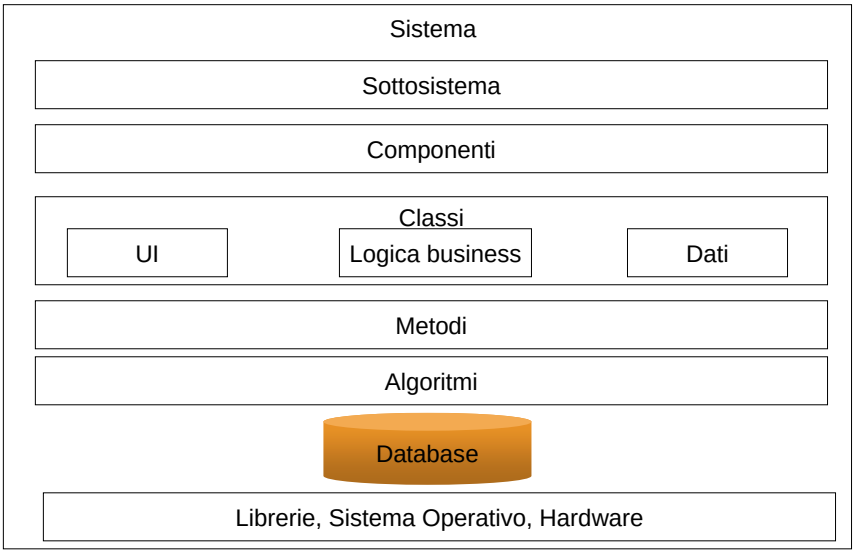
Metodi di test: black box

- Non coinvolgono la struttura interna dell'elemento in esame
- Deve essere nota solo la sua interfaccia
- Evidenziano errori ma non la loro causa
- Possono essere compiuti esaustivamente rispetto a percorsi (es. i basic course degli use case)

Metodi di test: white box

- Coinvolgono la struttura dell'elemento in esame
- Deve essere nota la sua struttura interna
- Esame dei possibili percorsi di esecuzione
- Sono più complessi e dispendiosi

La mappa logica del sistema



Metodi di test nel loro contesto

Metodi di test	Presupposti	Oggetto del test
Classi di equivalenza (black-box)	Casi d'uso, modello di dominio, tipo di var	Sottosistemi
Analisi dei valori limite (black-box)	Informazioni sul tipo di var, limiti dal modello	Sottosistemi
Gradi di copertura (white-box)	Struttura del codice	Metodi
Test esplorativi	Conoscenza del dominio, analisi, prog.	Applicazione, sottosistemi
Test orientati alle classi	Modello di analisi	Classi
Test orientati ai dati	Strutture di dati, intervalli di definizione	Metodi
Ispezione	Codice sorgente, documenti	Codice sorgente, documenti
Soluzione degli errori	Esperienza, fortuna	Applicazione, sottosistemi

Uso dei test black-box

- Ignorano la struttura interna
- Partono dai casi d'uso dell'analisi
- Servono a verificare il corretto funzionamento del sistema:
  - Se tutto bene, allora convalida
  - Se vengono rilevati comportamenti errati, allora vanno individuati gli errori con altri metodi

Suddivisione in classi di equivalenza

- Non si possono esaminare tutti gli input possibili
- Occorre dividere gli input in classi di equivalenza, ossia sottoaree che dovrebbero produrre un comportamento identico nel sistema
- Occorre particolare attenzione ai valori limite, ossia ai limiti degli intervalli di equivalenza o proprio dei valori possibili

White-box e gradi di copertura

- Focus sul codice concreto
- Istruzioni mappate in grafo
- Approccio bottom-up
- Copertura Cn
  - Copertura delle istruzioni
  - Copertura degli archi
  - Copertura dei cammini (teorica)
  - Copertura delle condizioni

Test delle classi

- Istanziare la classe inizializzando gli attributi a valori ragionevoli
- Verificare le operazioni che non cambiano lo stato dell'oggetto
- Verificare tutte le operazioni che cambiano lo stato
- Testare le serie di operazioni secondo almeno la copertura C1

Test delle classi - 2

- Test dimostrativo
- Completezza
- Test dello stato del modello

Test delle basi di dati

- Devono essere evidenziati i dati errati eventualmente presenti nel database
- Verifica di:
  - Tabelle disponibili
  - Campi delle tabelle
  - Collegamenti locali (entro un singolo record)
  - Dipendenza tra le tabelle
  - Connessioni complesse

Test di funzione e struttura combinati

- Limiti dei processi visti:
  - Non è possibile con un solo test di struttura (white-box) accertare se una funzionalità è mancante o errata
  - I test di funzione (black-box) non possono identificare un'implementazione mal funzionante

➢ I due approcci devono essere combinati

Test di funzione e struttura combinati –2

- Test di funzione
  - Da classi di equivalenza e valori limite si eseguono casi
  - Determinare quali parti del flusso esecutivo non sono provate
- Test di struttura
  - Per queste parti progettare test White-box
  - Copertura il più ampia possibile
- Test di regressione
  - Dopo la correzione verificare di nuovo il sistema, o almeno la parti coinvolte

Test di sistema

- Configurazione
  - Eseguito sulla piattaforma di destinazione
  - Si rieseguono i test precedenti
- Installazione
  - I componenti si installano senza problemi?
- Funzioni
  - Le funzioni vengono eseguite regolarmente?
- Prestazioni
  - Il sistema risponde in tempi conformi alle attese?

Test di sistema: le prestazioni

- Risorse
  - Memoria e disco reali
- Tempi di risposta
  - Casi reali d'uso
- Carico
  - Carico risultante, transazioni (tipico OLTP)
  - Simulatore di carico
- Throughput
  - Flusso dei dati (tipico batch)

Test di sistema - 2

- Usabilità
  - Caso d'uso realistico con utente finale
- Sicurezza
  - Verificare i casi critici per la sicurezza
- Interoperabilità
  - Funziona correttamente la comunicazione con altri sistemi?
- Reinizializzazione
  - Il sistema riparte correttamente?

Test esplorativi

- Eseguire una prima panoramica
- Pianificare
- Navigare
- Osservare
- Sperimentare
- Decidere
- Tracciare
- Raccontare

Test esplorativi - 2

- Creazione dei casi di test
- Descrizione dei casi di test

- Generazione dei dati di base
- Verifica dei dati giornalieri
- Verifica delle funzioni

Esecuzione dei test

- Concordare le condizioni di fine del test
- Preparare la piattaforma di test
- Eseguire i test e compilarne resoconto
- Identificare e documentare gli errori riscontrati nel modo più chiaro possibile

Esecuzione dei test: conseguenze

- Comunicazione degli errori
- Classificazione degli errori
- Assegnazione degli errori
- Eliminazione degli errori
- Distribuzione delle correzioni
- Verifica degli errori
- Termine degli errori

Automatizzazione dei test

- Strumentazione del codice
- Strumenti di test
- Simulatori

Errori con i test: ipotesi generali

- Solo i test e i tester sono responsabili della qualità del sistema
- L'unico compito del test è trovare gli errori del programmatore
- I programmatori sono in grado di trovare da soli i loro errori

Errori con i test: pianificazione

- L'attenzione è rivolta alle qualità sbagliate
- I test considerano rischi non realistici
- Il procedimento di test non è stato validato
- Troppa attenzione all'esecuzione del test e poca al suo progetto
- Test non abbastanza diversificati

Errori con i test: pianificazione

- Test di regressione mancanti o eseguiti solo sugli errori iniziali
- Dati di test troppo precisi (casi tralasciati)
- Dati di test troppo imprecisi o poco significativi

Errori con i test: esecuzione

- Test eseguiti sempre in ordine fisso
- Test eseguiti solo sulle interfacce utente (nessun test di struttura)
- Si cercano solo errori poco importanti
- Si trascurano le “piccolezze”
- Non sono previsti messaggi di errore
- Non è previsto test delle routine di installazione

Errori con i test: documentazione

- Non viene creata documentazione concomitante con i test
- La correzione degli errori avviene senza documentarne la riuscita
- Il report di test è insufficiente
- Manca una valutazione dei dati dei test già utilizzati per permettere l'eventuale correzione del processo stesso di test

Errori con i test: Suggerimenti

- Eseguire i test il prima possibile
- Nessun caso di test è "troppo banale"
- Tenere documentazione ordinata sui test
- Eseguire i test durante il processo di integrazione
- Individuare blocchi con errori ricorrenti (cut-and-paste di codice)

Errori con i test: Suggerimenti - 2

- La verifica non è mai sufficiente: trovare un compromesso ragionevole
- Il tasso di ritrovamento degli errori è massimo all'inizio dei test e poi diminuisce continuamente
- La legge di Murphy viene quasi sempre rispettata…

Sommario

- Definire il processo di Test
- Caratteristiche importanti
- Linee guida
- Criteri di completezza