



UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica

Corso di Laurea Magistrale in Informatica

TEST DESIGN DOCUMENT

INGEGNERIA, GESTIONE ED EVOLUZIONE DEL SOFTWARE

DOCENTI

Prof. Andrea De Lucia

TUTOR

Dott. Stefano Lambiase

Università degli Studi di Salerno

STUDENTI

Francesco Maria Torino

(0522501879)

Francesco Alessandro

Pinto (0522501981)

Stefano Guida

(0522502054)

Indice

Elenco delle Figure	ii
Elenco delle Tabelle	iii
1 Introduzione	1
2 Category Partitioning	3
2.1 RF01 - Info	3
2.2 RF02 - Get Smells	4
2.3 RF03 - Get Smells By Date	5
2.4 RF04 - Get Cultural Dispersion	6
2.5 RF05 - Get Community Patterns	7
3 Test Case	8
3.1 RF01 - Info	8
3.2 RF02 - Get Smells	9
3.3 RF03 - Get Smells By Date	10
3.4 RF04 - Get Cultural Dispersion	11
3.5 RF05 - Get Community Patterns	12

Elenco delle figure

Elenco delle tabelle

CAPITOLO 1

Introduzione

Il presente documento ha l'obiettivo di illustrare il processo di progettazione dei test di sistema relativi alle funzionalità offerte dal sistema GUIDO, adottando la tecnica del *Category Partitioning*. Per ciascun Requisito Funzionale (RF), saranno individuati i parametri significativi della richiesta, le categorie ad essi associate e le rispettive scelte possibili. A partire da tali elementi, verranno poi definiti i *test frame*, ovvero le combinazioni significative di valori che costituiranno la base per la generazione dei test case automatizzati.

I test di sistema preesistenti, anch'essi sviluppati mediante questa metodologia, presentavano tuttavia alcune criticità, evidenziate all'interno del *Test Plan*. In particolare, sono state riscontrate semplificazioni nella definizione delle categorie, incompletezze nella specifica dei vincoli e delle scelte nei test frame, nonché problematiche legate all'effettiva implementazione dei test automatici tramite Selenium. Alla luce di tali osservazioni, si è proceduto a una revisione e a un perfezionamento delle specifiche originarie, al fine di migliorarne la coerenza, la copertura e l'affidabilità.

Per ciascun requisito funzionale verranno quindi presentati:

- il **category partitioning aggiornato**, comprendente parametri, categorie, scelte significative e relativi vincoli;

- i **test frame** derivanti dalle combinazioni selezionate, utilizzati come base per la progettazione dei test case.

Le informazioni fornite in questa sede costituiranno il riferimento per la successiva realizzazione dei test automatici all'interno della suite di sistema, contribuendo a garantire una copertura funzionale ampia e metodologicamente solida.

CAPITOLO 2

Category Partitioning

2.1 RF01 - Info

Descrizione	
Il sistema (Chatbot) deve interpretare correttamente una richiesta Info	
Parametro: Intent	
Nome Categoria	Scelte
Intent [IR]	1. Tipo Intent \neq Info [ERROR]
	2. Tipo Intent = Info

2.2 RF02 - Get Smells

Descrizione	
Il sistema (Chatbot) deve interpretare correttamente una richiesta di tipo get_smells	
Parametro: Richiesta	
Nome Categoria	Scelte
Intent [IR]	1. Intent \neq get_smells [ERROR]
	2. Intent = get_smells [PROPERTY IR_OK]
Parametro: Link	
Formato: (? :https:)? (? www\.) ?github.com/ ([a-zA-Z0-9_]+) (/ [a-zA-Z0-9_]+)	
Nome Categoria	Scelte
Formato [FL]	1. Formato link = false [IF IR_OK][ERROR]
	2. Formato link = true [IF IR_OK]

2.3 RF03 - Get Smells By Date

Descrizione	
Il sistema (Chatbot) deve interpretare correttamente una richiesta di tipo get_smells_by_date	
Parametro: Richiesta	
Nome Categoria	Scelte
Intent [IR]	1. Intent \neq get_smells [ERROR]
	2. Intent = get_smells [PROPERTY IR_OK]
Parametro: Link	
Formato: (?:https:)?(?:www\.)?github.com/([a-zA-Z0-9_]+)(/[a-zA-Z0-9_]+)	
Nome Categoria	Scelte
Formato [FL]	1. Formato link = false [IF IR_OK][ERROR]
	2. Formato link = true [IF IR_OK]
Parametro: Data	
Formato: DD/MM/YYYY	
Nome Categoria	Scelte
Formato [FD]	1. Formato data = false [IF IR_OK][ERROR]
	2. Formato data = true [IF IR_OK][PROPERTY FD_OK]
Validità [VD]	1. Data > Today [IF FD_OK][ERROR]
	2. Data <= Today [IF FD_OK]

2.4 RF04 - Get Cultural Dispersion

Descrizione	
Il sistema (Culture Inspector) deve calcolare correttamente la dispersione culturale del team	
Parametro: Nazionalità	
Nome Categoria	Scelte
Numero Nazioni [NN]	1. Numero Nazioni ≤ 0 [ERROR]
	2. Numero Nazioni = 1 [PROPERTY NN_OK]
	3. Numero Nazioni > 1 [PROPERTY NN_OK]
Numero Partecipanti [NP]	1. Numero Partecipanti ≤ 0 [IF NN_OK][ERROR]
	2. Numero Partecipanti = 1 [IF NN_OK]
	3. Numero Partecipanti > 1 [IF NN_OK]
Nazionalità [NA]	1. Nazionalità Inesistenti [IF NN_OK] [ERROR]
	2. Nazionalità Esistenti [IF NN_OK]
Parametro: Geo Dispersion	
Nome Categoria	Scelte
Valore [VA]	1. Geodispersione < 0 OR > 100 [ERROR]
	2. Geodispersione ≥ 0 AND ≤ 100

2.5 RF05 - Get Community Patterns

Descrizione	
Il sistema deve interpretare correttamente una richiesta get_community_patterns	
Parametro: Link	
Formato: (?:(https:)?(?www\.))?github.com/([a-zA-Z0-9_]+)(/[a-zA-Z0-9_]+)	
Nome Categoria	Scelte
Formato [FL]	1. Formato link = false [ERROR]
	2. Formato link = true [Property FL_OK]
Parametro: Data Fine	
Formato: DD/MM/YYYY	
Nome Categoria	Scelte
Formato [FD]	1. Formato data = false [ERROR]
	2. Formato data = true [Property FD_OK]
Validità [VD]	1. Data Fine > Today [IF FD_OK][ERROR]
	2. Data Fine <= Today [IF FD_OK] [PROPERTY VD_OK]
Parametro: Repository	
Nome Categoria	Scelte
Commit [CM]	1. Commit < 100 [IF FL_OK and VD_OK][ERROR]
	2. Commit >= 100 [IF FL_OK and VD_OK]
Milestones [ML]	1. Milestones < 1 [IF FL_OK and VD_OK][ERROR]
	2. Milestones >= 1 [IF FL_OK and VD_OK]
Membri [ME]	1. Membri < 2 [IF FL_OK and VD_OK][ERROR]
	2. Membri >= 2 [IF FL_OK and VD_OK]

CAPITOLO 3

Test Case

3.1 RF01 - Info

Test Case ID	Test Frame	Risultato
TC_IN_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_IN_2	IR2	Corretto: il chatbot restituisce le info

3.2 RF02 - Get Smells

Test Case ID	Test Frame	Risultato
TC_GS_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_GS_2	IR2, FL1	Errore: Link Repository non valido
TC_GS_3	IR2, FL2	Corretto: il chatbot mostra i community smells rilevati nella repository

3.3 RF03 - Get Smells By Date

Test Case ID	Test Frame	Risultato
TC_GSD_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_GSD_2	IR2, FL1, FD2, VD2	Errore: Link Repository non valido
TC_GSD_3	IR2, FL2, FD1	Errore: formato data non valido
TC_GSD_4	IR2, FL2, FD2, VD1	Errore: data non valida
TC_GSD_5	IR2, FL2, FD2, VD2	Corretto: il chatbot mostra i community smells rilevati nella repository a partire dalla data specificata

3.4 RF04 - Get Cultural Dispersion

Test Case ID	Test Frame	Risultato
TC_GCD_1	NN1, VA2	Errore: Composizione Team Non Valida
TC_GCD_2	NN2, NP1, NA2, VA2	Errore: Numero Partecipanti non Valido
TC_GCD_3	NN3, NP2, NA1, VA2	Errore: Nazione Inesistente
TC_GCD_4	NN2, NP3, NA2, VA1	Errore: Geodispersione non valida
TC_GCD_5	NN3, NP3, NA2, VA2	Corretto: il Culture Inspector mostra le metriche calcolate

3.5 RF05 - Get Community Patterns

Test Case ID	Test Frame	Risultato
TC_GCP_1	FL1, FD2, VD2	Errore: Url Repository Non Valido
TC_GCP_2	FL2, FD1	Errore: Formato data non valido
TC_GCP_3	FL2, FD2, VD1	Errore: Data Non Valida
TC_GCP_4	FL2, FD2, VD2, CM1, ML2, ME2	Errore: Commit Insufficienti
TC_GCP_5	FL2, FD2, VD2, CM2, ML1, ME2	Errore: Milestone Insufficienti
TC_GCP_6	FL2, FD2, VD2, CM2, ML2, ME1	Errore: Membri Insufficienti
TC_GCP_7	FL2, FD2, VD2, CM2, ML2, ME2	Corretto: il Community Inspector mostra i risultati