



UNIVERSITÀ DEGLI STUDI DI SALERNO

Dipartimento di Informatica

Corso di Laurea Magistrale in Informatica

Corso di

INGEGNERIA, GESTIONE ED EVOLUZIONE DEL SOFTWARE

Test Design Document

DOCENTE

Prof. Andrea De Lucia

Università degli Studi di Salerno

REVIEWER

Dott. Stefano Lambiase

Università degli Studi di Salerno

AUTORI

Benedetto Scala

Mat: 0522501794

Leopoldo Todisco

Mat: 0522501795

Carlo Venditto

Mat: 0522501796

Indice

Elenco delle Figure	ii
Elenco delle Tabelle	iii
1 Introduzione	2
1.1 Contesto del Progetto	2
1.2 Funzionalità da testare	3
1.3 Criteri di Adeguatezza	3
1.4 Struttura del documento	4
2 Test Cases	5
2.1 Test Case Get_Smells	5
2.2 Test Case Get_Smells_By_Date	6
2.3 Test Case Info	7
2.4 Test Culture-Inspector	7
2.5 Test Suite	8

Elenco delle figure

Elenco delle tabelle

1.1	Tabella dei requisiti funzionali del sistema Cadocs II	3
-----	--	---

a

CAPITOLO 1

Introduzione

1.1 Contesto del Progetto

Negli studi recenti di Ingegneria del Software, la comunità ha cominciato a preoccuparsi dell'impatto degli aspetti umani nello sviluppo del software. I lavori esistenti hanno analizzato come gli sviluppatori e le sotto-comunità interagiscono, con l'obiettivo di individuare schemi di comunicazione e collaborazione, portando al concetto di "Community Smells". I community smells riflettono schemi organizzativi e socio-tecnici sub-ottimali nella struttura della comunità del software.

Allo stato dell'arte, siamo a conoscenza di un solo tool, CADOCS, che propone il rilevamento di community smells. Tale tool, è presente in due varianti:

- CADOCS che è un Conversational Agent integrato in Slack;
- CADOCS II che è la corrispettiva desktop application standalone;

Inoltre, data l'introduzione recente di questi concetti, molti manager non sono a conoscenza della loro esistenza e dell'importanza della loro individuazione.

1.2 Funzionalità da testare

Come già riportato nel documento di Pre-Maintenance, i requisiti funzionali di Cadocs II sono 4, ma di queste solo 3 sono state implementate del tutto come si può vedere nella tabella 1.1.

Requisito Funzionale	Descrizione
RF1: Get Smells	Tale requisito fa riferimento alla capacità del sistema di poter rilevare Community Smells in una repository Github.
RF2: Get Smells By Date	Tale requisito fa riferimento alla capacità del sistema di poter rilevare Community Smells in una repository Github a partire da una determinata data.
RF3: Info	Tale requisito fa riferimento alla capacità del sistema di spiegare cosa è in grado di fare e di spiegare i community smells agli utenti

Tabella 1.1: Tabella dei requisiti funzionali del sistema Cadocs II

1.3 Criteri di Adeguatezza

Un test viene considerato passato quando il comportamento ottenuto è uguale al comportamento atteso dall'oracolo, altrimenti il test si considera fallito.

Nel contesto del nostro progetto saranno testati tutti i requisiti funzionali, sebbene nel momento di inizio del progetto non vi è alcun test da cui partire, e additionally, si testerà anche tutto ciò che si implementerà per realizzare le Change Requests.

Per quanto riguarda i test di unità e integrazione si fa affidamento a tecniche white-box e il requisito minimo è quello del raggiungimento del 75% di branch coverage.

1.4 Struttura del documento

Tale documento è così strutturato:

- Capitolo 1: Introduzione;
- Capitolo 2: Test Cases;

2.1 Test Case Get_Smells

Descrizione	
Il sistema deve interpretare correttamente una richiesta di tipo get_smells	
Parametro: richiesta	
Nome Categoria	Scelte
Intent [IR]	1. Tipo intent \neq get_smells [ERROR] 2. Tipo 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 \neq false [ERROR] 2. Formato link = true [Property IR_OK]

2.2 Test Case Get_Smells_By_Date

Descrizione	
Il sistema deve interpretare correttamente una richiesta di tipo get_smells_by_date	
Parametro: richiesta	
Nome Categoria	Scelte
Intent [IR]	1. Tipo intent \neq get_smells [ERROR] 2. Tipo 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 [ERROR] 2. Formato link = true [IF IR_OK][Property FL_OK]
Parametro: Data	
Formato:	
DD/MM/YYYY	
Nome Categoria	Scelte
Formato [FD]	1. Formato data = false [ERROR] 2. Formato data = true [IF FL_OK][Property FD_OK]
Validità [VD]	1. Validità data = false [ERROR] 2. Validità data = true [IF FD_OK]

2.3 Test Case Info

Descrizione	
Il sistema deve interpretare correttamente una richiesta di tipo info	
Parametro: richiesta	
Nome Categoria	Scelte
Intent [IR]	1. Tipo intent \neq info [ERROR] 2. Tipo intent = info

2.4 Test Culture-Inspector

Descrizione	
Il sistema deve interpretare correttamente una richiesta di tipo geodispersion	
Parametro: intent	
Nome Categoria	Scelte
Intent [IR]	1. Tipo intent \neq geodispersion [ERROR] 2. Tipo intent = geodispersion
Parametro: Nazionalità	
Nome Categoria	Scelte
Nazionalità [NA]	1. Nazionalità NOT in csv [ERROR] 2. Nazionalità in CSV
Numero Partecipanti [NUM]	1. Numero Partecipanti < 0 [ERROR] 2. Numero Partecipanti > 0
Parametro: GeoDispersionValue	
Nome Categoria	Scelte
Value [VA]	1. Value > 100 OR Value < 0 [ERROR] 2. Value > 0 AND Value < 100

2.5 Test Suite

Test Case ID	Test Frame	Risultato
TC_GS_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_GS_2	IR2, FL1	Errore: il formato del link non è corretto
TC_GS_3	IR2, FL2	Corretto

Test Case ID	Test Frame	Risultato
TC_GSD_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_GSD_2	IR2, FL1	Errore: il formato del link non è corretto
TC_GSD_3	IR2, FL2, FD1	Errore: il formato della data non è corretto
TC_GSD_4	IR2, FL2, FD2, VD1	Errore: la data non è valida
TC_GSD_5	IR2, FL2, FD2, VD2	Corretto

Test Case ID	Test Frame	Risultato
TC_RP_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_RP_2	IR2	Corretto

Test Case ID	Test Frame	Risultato
TC_IN_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_IN_2	IR2	Corretto

Test Case ID	Test Frame	Risultato
TC_CI_1	IR1	Errore: la richiesta non è interpretata correttamente
TC_CI_2	IR2, NA1	Errore: la nazionalità non ha valori di Hofstede
TC_CI_3	IR2, NA2, NUM1	Errore: il numero di partecipanti non è tollerato
TC_CI_4	IR2, NA2, NUM2, VA1	Errore: il valore per la geodispersione non è tollerato
TC_CI_5	IR2, NA2, NUM2, VA2	Corretto