



LAUREA MAGISTRALE IN SOFTWARE ENGINEERING AND IT MANAGEMENT
UNIVERSITÀ DI SALERNO
CORSO DI INGEGNERIA GESTIONE EVOLUZIONE DEL SOFTWARE

PROF. ANDREA DE LUCIA
DOTT. MANUEL DE STEFANO,
EMANUELE IANNONE

Test Plan

Repository GitHub:

<https://github.com/rebeccadimatteo/csDetector/tree/Explainability/explainability>

<https://github.com/rebeccadimatteo/CADOCS>

2023

Rebecca Di Matteo

Leonardo Monaco

1	Introduzione	2
2	Testing Approach	4
2.0.1	Regression Testing	4
2.0.2	System Testing	4
2.0.3	Unit Testing	5
3	Features to Test	6
4	Pass and Fail Criteria	8
5	Category Partition	9
5.1	CR_1 Estensione di CSDETECTOR con modulo di ML basato sull'Explainability	9
5.2	CR_2 Estensione di CADOCS con un interfaccia grafica per rappresentare l'Explainability	10
5.3	CR_3 Refactoring del codice	17

Per fornire agli utenti nuove conoscenze, si vuole raffinare il tool CADOCS in modo da dare ai praticanti uno strumento migliore in grado di esprimere anche l'Explainability dei modelli che ricavano, mediante le metriche socio-tecniche, la presenza dei Community Smells. Explainable Artificial Intelligence è un insieme di metodi e processi che consentono agli utenti di comprendere e considerare attendibili i risultati e l'output creati dagli algoritmi di Machine Learning. L'intelligenza artificiale spiegabile viene utilizzata per descrivere un modello di intelligenza artificiale, il relativo impatto previsto ed i potenziali errori.

Le funzionalità del tool verranno estese mediante l'analisi della spiegabilità dei modelli di predizione dei Community Smells per mezzo di due librerie: **Python LIME e SHAP**, al fine di comprendere quali metriche socio-tecniche siano più influenti nella predizione dei suddetti. L'Explainability sarà inserita all'interno di CADOCS mediante un'interfaccia grafica che rappresenterà l'output di una delle due librerie. La scelta dell'Explainability da rappresentare sarà affettuta sulla base di uno studio relativo ai risultati di un questionario che, sarà somministrato agli studenti del Dipartimento di Informatica dell'Università degli Studi di Salerno. Nel questionario verranno rappresentati gli output delle due librerie con delle domande a riguardo. I risultati ottenuti ci permetteranno di rispondere alle domande di ricerca che ci siamo posti per comprendere quale fra i due framework ha un'Explainability più chiara e una predizione più veritiera nel determinare le metriche socio-tecniche per la ricerca dei Community Smells, così da rappresentarla graficamente su CADOCS. Nello stato attuale CADOCS non contiene un approfondimento relativo all'Explainability. L'inserimento di tale

approfondimento può portare l'utente ad avere una maggiore fiducia sui risultati del processo decisionale che porta all'identificazione dei Community Smells. Inoltre, basandosi su questi risultati l'utente può comprendere quale metrica socio-tecnica è più influente rispetto ad un'altra nel determinare il Community Smells e dunque attuare delle strategie per risolverlo.

Dato che stiamo per evolvere e mantenere uno strumento già esistente, considereremo più di un approccio per i test. In particolare, durante la manutenzione, dobbiamo assicurarci che le nuove porzioni di codice introdotte non abbiano compromesso la regolare esecuzione dello strumento. Per questo motivo, abbiamo in programma di eseguire **Regression Testing** per essere sicuri che CADOCs continui a funzionare come previsto anche dopo le nostre modifiche. A tal fine, eseguiremo anche **System Testing** per verificare se la nuova versione dello strumento funziona come la precedente. Si prevede inoltre di eseguire dei **Unit Testing** per le funzioni aggiunte al sistema. Per definire i casi di test utilizzeremo la tecnica del **Category Partition**.

2.0.1 Regression Testing

Per attuare il Regression Testing prenderemo in considerazione, i test già precedenti alle modifiche effettuate al sistema e li rieseguiremo, per accertarci che le nostre modifiche non abbiano generato fault al sistema, che causerebbero delle failure.

2.0.2 System Testing

È il testing applicato sul sistema software completo ed integrato. L'obiettivo è quello di valutare l'adesione del sistema ai requisiti specificati. Dunque andremo a verificare se la nuova versione dello strumento funziona come la precedente, nello specifico se rispetta tutti i requisiti definiti nella fase di analisi.

2.0.3 Unit Testing

Eseguiamo l' Unit Testing sulle nuove funzionalità che creeremo nel contesto della CR.

Features to Test

In questa sezione spiegheremo ad alto livello le caratteristiche che testeremo. Nell'ambito di ogni richiesta di modifica, identificheremo i nuovi requisiti funzionali introdotti.

CR_1 Estensione di CSDETECTOR con modulo di ML basato sull'Explainability
Test Item
L'obiettivo della modifica è quello di estendere CSDETECTOR aggiungendo un modulo di ML basato sull'Explainability, essendo un modulo costruito esternamente non andrà a compromettere il sistema nè i requisiti.
Tested Features
NA

Tabella 3.1: CR_1 Feature to test

CR_2 Estensione di CADOCS con un interfaccia grafica per rappresentare l'Explainability	
Test Item	
Test Item	
<p>L'obiettivo della modifica è quello di estendere CADOCS aggiungendo l'Explainability, tale modifica porterà ad eseguire il sistema in modo diverso, dunque verificheremo se la nuova esecuzione è equivalente a quella precedente.</p> <p>Nel fare ciò il processo di test coprirà ogni nuova funzione introdotta</p>	
Tested Features	
CR_2 : Il Sistema deve essere in grado di rappresentare l'Explainability sulla piattaforma Slack	

Tabella 3.2: CR_2 Feature to test

CR_3 Refactoring del codice	
Test Item	
<p>L'obiettivo della modifica è riorganizzare, ristrutturare e rendere più chiaro il codice già esistente di CADOCS, tale modifica porterà ad eseguire il sistema in modo diverso, dunque verificheremo se la nuova esecuzione è equivalente a quella precedente, nel fare ciò il processo di test eseguirà ogni nuova funzione introdotta.</p>	
Tested Features	
<p>CR_3 : Il Sistema deve essere in grado di eseguire tutte le funzionalità che eseguiva precedentemente al nostro refactoring.</p> <p>RF_1 Rilevare Community Smells; RF_2 Rilevare Community Smells per data; RF_3 Mostrare un report dell'ultima esecuzione; RF_4 Rappresentazione Community Smells; RF_5 Suggerire strategie di Refactoring;</p>	

Tabella 3.3: CR_3 Feature to test

Pass and Fail Criteria

I test che stiamo per eseguire hanno l'obiettivo di rilevare se ci sono guasti all'interno del sistema per poi correggerli. Ogni test avrà un **oracolo** che rappresenta l'uscita attesa di un ingresso specificato. Un test si considera superato se, dato un certo input dell'oracolo, l'output fornito dal sistema è conforme all'output descritto dall'oracolo. In caso contrario, se con un dato ingresso dell'oracolo l'output del sistema non rispetta l'oracolo, il test sarà considerato fallito.

In questa sezione, per ogni CR, verranno definiti i test frames per i test item identificati nelle sezioni precedenti.

5.1 CR_1 Estensione di CSDETECTOR con modulo di ML basato sull'Explainability

Per quanto riguarda la **CR_1** non saranno definiti test frames, poichè i modelli sono stati creati esternamente al sistema, ed è stato testato il grado di verità dell'Explainability di tale modelli mediante un questionario che confrontava il grado di Predizione-Realtà. Il questionario è stato sottoposto ad un campione di studenti dell'Università degli Studi di Salerno, nello specifico è stato inviato su diversi gruppi Whatsapp e Telegram con all'interno all'incirca 60 studenti frequentati il corso di studi di Informatica, dei quali solo 18 hanno partecipato al questionario. I risultati sono visualizzabili al seguente link: <https://github.com/rebeccadimatteo/csDetector/tree/Explainability/explainability>.

5.2 CR_2 Estensione di CADOCS con un interfaccia grafica per rappresentare l'Explainability

Descrizione
Il Sistema deve essere in grado di rappresentare l'Explainability sulla piattaforma Slack.
Parametri
message_example_OSE_explainability message_example_BCE_explainability message_example_PDE_explainability message_example_SV_explainability message_example_OS_explainability message_example_SD_explainability message_example_RS_explainability message_example_TF_explainability message_example_UI_explainability message_example_TC_explainability

Categorie	
Category Name	Category Value
message_example_OSE_explainability	<p>SMS_1 : message_example_OSE_explainability contiene un messaggio valido contenente l'explainability del Community Smells OSE</p> <p>SMS_2 : message_example_OSE_explainability non è valido (passato come tipo None)</p>
message_example_BCE_explainability	<p>SMS_3 : message_example_BCE_explainability contiene un messaggio valido contenente l'explainability del Community Smells BCE</p> <p>SMS_4 : message_example_BCE_explainability non è valido (passato come tipo None)</p>
message_example_PDE_explainability	<p>SMS_5 : message_example_PDE_explainability contiene un messaggio valido contenente l'explainability del Community Smells PDE</p> <p>SMS_6 : message_example_PDE_explainability non è valido (passato come tipo None)</p>
message_example_SV_explainability	<p>SMS_7 : message_example_SV_explainability contiene un messaggio valido contenente l'explainability del Community Smells SV</p> <p>SMS_8 : message_example_SV_explainability non è valido (passato come tipo None)</p>
message_example_OS_explainability	<p>SMS_9 : message_example_OS_explainability contiene un messaggio valido contenente l'explainability del Community Smells OS</p> <p>SMS_10 : message_example_OS_explainability non è valido (passato come tipo None)</p>
message_example_SD_explainability	<p>SMS_11 : message_example_SD_explainability contiene un messaggio valido contenente l'explainability del Community Smells SD</p> <p>SMS_12 : message_example_SD_explainability non è valido (passato come tipo None)</p>

message_example_RS_explainability	<p>SMS_13 : message_example_RS_explainability contiene un messaggio valido contenente l'explainability del Community Smells RS</p> <p>SMS_14 : message_example_RS_explainability non è valido (passato come tipo None)</p>
message_example_TF_explainability	<p>SMS_15 : message_example_TF_explainability contiene un messaggio valido contenente l'explainability del Community Smells TF</p> <p>SMS_16 : message_example_TF_explainability non è valido (passato come tipo None)</p>
message_example_UI_explainability	<p>SMS_17 : message_example_UI_explainability contiene un messaggio valido contenente l'explainability del Community Smells UI</p> <p>SMS_18 : message_example_UI_explainability non è valido (passato come tipo None)</p>
message_example_TC_explainability	<p>SMS_19 : message_example_TC_explainability contiene un messaggio valido contenente l'explainability del Community Smells TC</p> <p>SMS_20 : message_example_TC_explainability non è valido (passato come tipo None)</p>
Vincoli	
Attributi	Vincoli
message_example_OSE_explainability	<p>SMS_1 : message_example_OSE_explainability contiene un messaggio valido contenente l'explainability del Community Smells OSE [Property correct]</p> <p>SMS_2 : message_example_OSE_explainability non è valido (passato come tipo None) [errore]</p>
message_example_BCE_explainability	<p>SMS_3 : message_example_BCE_explainability contiene un messaggio valido contenente l'explainability del Community Smells BCE [Property correct]</p> <p>SMS_4 : message_example_BCE_explainability non è valido (passato come tipo None) [errore]</p>

message_example_PDE_explainability	<p>SMS_5 : message_example_PDE_explainability contiene un messaggio valido contenente l'explainability del Community Smells PDE [Property correct]</p> <p>SMS_6 : message_example_PDE_explainability non è valido (passato come tipo None) [errore]</p>
message_example_SV_explainability	<p>SMS_7 : message_example_SV_explainability contiene un messaggio valido contenente l'explainability del Community Smells SV [Property correct]</p> <p>SMS_8 : message_example_SV_explainability non è valido (passato come tipo None) [errore]</p>
message_example_OS_explainability	<p>SMS_9 : message_example_OS_explainability contiene un messaggio valido contenente l'explainability del Community Smells OS [Property correct]</p> <p>SMS_10 : message_example_OS_explainability non è valido (passato come tipo None) [errore]</p>
message_example_SD_explainability	<p>SMS_11 : message_example_SD_explainability contiene un messaggio valido contenente l'explainability del Community Smells SD [Property correct]</p> <p>SMS_12 : message_example_SD_explainability non è valido (passato come tipo None) [errore]</p>
message_example_RS_explainability	<p>SMS_13 : message_example_RS_explainability contiene un messaggio valido contenente l'explainability del Community Smells RS [Property correct]</p> <p>SMS_14 : message_example_RS_explainability non è valido (passato come tipo None) [errore]</p>
message_example_TF_explainability	<p>SMS_15 : message_example_TF_explainability contiene un messaggio valido contenente l'explainability del Community Smells TF [Property correct]</p> <p>SMS_16 : message_example_TF_explainability non è valido (passato come tipo None) [errore]</p>

message_example_UI_explainability	<p>SMS_17 : message_example_UI_explainability contiene un messaggio valido contenente l'explainability del Community Smells UI [Property correct]</p> <p>SMS_18 : message_example_UI_explainability non è valido (passato come tipo None) [errore]</p>
message_example_TC_explainability	<p>SMS_19 : message_example_TC_explainability contiene un messaggio valido contenente l'explainability del Community Smells TC [Property correct]</p> <p>SMS_20 : message_example_TC_explainability non è valido (passato come tipo None) [errore]</p>

Tabella 5.1: Category Partition CR_2

Test Case ID	Test Frame	Result
TC_1_CR_2_ose	SMS_1	Success: il community smells è stato rappresentato correttamente
TC_1_1_CR_2_ose	SMS_2	Error : il community smells non è stato rappresentato correttamente
TC_2_CR_2_bce	SMS_3	Success: il community smells è stato rappresentato correttamente
TC_2_1_CR_2_bce	SMS_4	Error : il community smells non è stato rappresentato correttamente
TC_3_CR_2_pde	SMS_5	Success: il community smells è stato rappresentato correttamente
TC_3_1_CR_2_pde	SMS_6	Error : il community smells non è stato rappresentato correttamente
TC_4_CR_2_sv	SMS_7	Success: il community smells è stato rappresentato correttamente
TC_4_1_CR_2_sv	SMS_8	Error : il community smells non è stato rappresentato correttamente
TC_5_CR_2_os	SMS_9	Success: il community smells è stato rappresentato correttamente
TC_5_1_CR_2_os	SMS_10	Error : il community smells non è stato rappresentato correttamente
TC_6_CR_2_sd	SMS_11	Success: il community smells è stato rappresentato correttamente
TC_6_1_CR_2_sd	SMS_12	Error : il community smells non è stato rappresentato correttamente
TC_7_CR_2_rs	SMS_13	Success: il community smells è stato rappresentato correttamente
TC_7_1_CR_2_rs	SMS_14	Error : il community smells non è stato rappresentato correttamente
TC_8_CR_2_tf	SMS_15	Success: il community smells è stato rappresentato correttamente
TC_8_1_CR_2_tf	SMS_16	Error : il community smells non è stato rappresentato correttamente

TC_9_CR_2_ui	SMS_17	Success: il community smells è stato rappresentato correttamente
TC_9_1_CR_2_ui	SMS_18	Error : il community smells non è stato rappresentato correttamente
TC_10_CR_2_tc	SMS_19	Success: il community smells è stato rappresentato correttamente
TC_10_1_CR_tc	SMS_20	Error : il community smells non è stato rappresentato correttamente

Tabella 5.2: Test Frame CR_2

5.3 CR_3 Refactoring del codice

Nella **CR_3** abbiamo attuato un refactoring del codice rinominando alcuni metodi e variabili, rendendo il codice più leggibile e chiaro. Sulla base di queste modifiche dovremmo andare a testare le funzionalità di CADOCS per accertarci che le nostre modifiche non abbiano generato fault al sistema, che causerebbero delle failur. Testeremo il sistema sulla base dei test frames definiti nelle documentazioni precedenti da altri sviluppatori, ed rappresentati nel documento **Pre-Maintenance e Test Case Specification**, con l'aggiunta di nuovi test frames sviluppati sui requisiti funzionali di CADOCS: **RF_3**, **RF_4**, **RF_5**.

Sottoliniamo che essendo CADOCS un agente di conversazione basato sul linguaggio naturale, l'esecuzione delle sue funzionalità ha input basati su frasi. Al fine di specificare le categorie per questa CR, rilasceremo i concetti di Category Partition.

Descrizione	
Il sistema deve essere in grado di mostrare un report dell'ultima esecuzione.	
Parametri	
repo, pat, user, report	
Categorie	
Category Name	Category Value
report	R1: il Sistema contiene il report dell'ultima esecuzione R2: il Sistema non contiene il report dell'ultima esecuzione
Vincoli	
Attributi	Vincoli
report	R1: il Sistema contiene il report dell'ultima esecuzione R2: il Sistema non contiene il report dell'ultima esecuzione [errore]

Tabella 5.3: Category Partition RF_3

Test Case ID	Test Frame	Result
TC_1_RF_3	R1	Success: l'agente conversazionale è in grado di rappresentare il report dell'ultima esecuzione
TC_1_1_RF_3	R2	Error : l'agente conversazionale non è in grado di rappresentare il report dell'ultima esecuzione

Tabella 5.4: Test frame RF_3

Descrizione	
Il sistema deve essere in grado di rappresentare i Community Smells su Slack.	
Parametri	
repo, pat, user, message	
Categorie	
Category Name	Category Value
message	R1: il messaggio contiene i Community Smells R2: il messaggio non contiene i Community Smells
Vincoli	
Attributi	Vincoli
message	R1: il messaggio contiene dei Community Smells R2: il messaggio non contiene i Community Smells [errore]

Tabella 5.5: Category Partition RF_4

Test Case ID	Test Frame	Result
TC_1_RF_4	R1	Success: l'agente conversazionale è in grado di rappresentare i Community Smells come output
TC_1_1_RF_4	R2	Error : l'agente conversazionale non è in grado di rappresentare i Community Smells come output

Tabella 5.6: Test frame RF_4

Descrizione	
Il sistema deve essere in grado di suggerire strategie di Refactoring	
Parametri	
repo, pat, user, message	
Categorie	
Category Name	Category Value
message	R1: il messaggio contiene dei suggerimenti su strategie di Refactoring R2: il messaggio non contiene suggerimenti su strategie di Refactoring
Vincoli	
Attributi	Vincoli
message	R1: il messaggio contiene suggerimenti su strategie di Refactoring R2: il messaggio non contiene suggerimenti su strategie di Refactoring [errore]

Tabella 5.7: Category Partition RF_5

Test Case ID	Test Frame	Result
TC_1_RF_4	R1	Success: l'agente conversazionale è in grado rappresentare dei suggerimenti sulle strategie di Refactoring
TC_1_1_RF_4	R2	Error : l'agente conversazionale non è in grado di rappresentare suggerimenti sulle strategie di Refactoring

Tabella 5.8: Test frame RF_5