
Test Incident Report

Progetto:

CleanDesk



CleanDesk

Powered by AI

Riferimento:	
Versione:	0.8
Data:	09/03/2023
Destinatario:	Esame di Ingegneria del Software 2023/24
Presentato da:	Ambrosio Gennaro, Camoia Andrea
Approvato da:	



UNIVERSITÀ DEGLI STUDI
DI SALERNO



Revision Hystory

Data	Versione	Descrizione	Autori
15/02/2024	0.1	Stesura iniziale del documento	Ambrosio Gennaro Camoia Andrea
05/03/2024	0.5	Aggiunta Test Incident Report	Ambrosio Gennaro Camoia Andrea
09/03/2024	0.8	Revisione Finale	Ambrosio Gennaro Camoia Andrea



Team Member

Nome	Ruolo nel progetto	Acronimo	Informazioni di contatto
Andrea Camoia	Team Member	AC	a.camoia@studenti.unisa.it
Gennaro Ambrosio	Team Member	AG	g.ambrosio35@studenti.unisa.it



Indice

1	Introduzione	4
1.1	Scopo del Documento	4
1.2	Riferimenti	4
2	Ambiente di esecuzione	5
3	Test Incident Report	6
3.1	Unit Testing	6
3.2	Integration Testing	6
3.3	System Testing	6
3.3.1	Errori Riscontrati	7

1 Introduzione

CleanDesk è un software progettato per migliorare l'esperienza informatica e aumentare la produttività degli utenti. Sfruttando l'intelligenza artificiale, organizza i file dello spazio digitale in modo intuitivo ed accessibile, offrendo un ambiente di utilizzo semplice ed efficace.

Inoltre, il sistema consente di gestire in modo consapevole lo spazio utilizzato da una cartella: analizzandone dettagliatamente l'utilizzo fisico e individuando file duplicati che occupano spazio inutilmente, contribuendo così a ridurre la confusione e ottimizzare la memoria disponibile.

1.1 Scopo del Documento

Il seguente documento mostra l'**esito di tutti i test** effettuati, **descrivendo eventuali** incidenti e i **fallimenti scoperti** durante l'attività, che devono essere analizzati. Una volta completata l'analisi, saranno pianificate le modifiche da apportare al sistema per risolverli.

Come già specificato, saranno testate le funzionalità che richiedono l'interazione con l'utente.

1.2 Riferimenti

- CleanDesk Statement of Work (SOW)
- CleanDesk Requirements Analysis Document (RAD)
- CleanDesk System Design Document (SDD)
- CleanDesk Test Plan (TP)
- CleanDesk Test Case Specification (TCS)
- CleanDesk Object Design Document (ODD)
- Dispense del prof. Carmine Gravino, fornite mediante la pagina del corso "Ingegneria del Software - Resto 2 - 2023/2024" sulla Piattaforma E-learning del Corso di Laurea in Informatica dell'Università degli Studi di Salerno;
- Libro di testo "Object Oriented Software Engineering Using UML Patterns and Java Prentice Hall 2010 Bernd Bruegge Allen H.Dutoit"
- Libro di testo "C. GHEZZI, D. MANDRIOLI, M. JAZAYERI, INGEGNERIA DEL SOFTWARE – FONDAMENTI E PRINCIPI, PRENTICE HALL, 2004"



2 Ambiente di esecuzione

Per eseguire il test dell'applicazione, è stato utilizzato il framework Mocha di JavaScript. Questa tecnologia è stata impiegata per automatizzare e ottimizzare il processo di testing, consentendo la **suddivisione dei test in categorie** per una migliore organizzazione e gestione.

In particolare, i tester hanno potuto creare una funzione dedicata per ogni caso di test presente nel Test Case Specification, suddividendo quest'ultima in diverse categorie.

3 Test Incident Report

3.1 Unit Testing

Test Case ID	Test Incident ID	Tester	Data	Result
TC_OF.1_1	TI_OF.1_1	AC	19/03/2024	Passed
TC_OF.1_2	TI_OF.1_2	AC	19/03/2024	Passed
TC_OF.1_3	TI_OF.1_3	AC	19/03/2024	Passed
TC_OF.1_4	TI_OF.1_4	AC	19/03/2024	Passed
TC_OF.1_5	TI_OF.1_5	AC	19/03/2024	Passed
TC_OF.2_1	TI_OF.2_1	AG	20/03/2024	Passed
TC_OF.2_2	TI_OF.2_2	AG	20/03/2024	Passed
TC_OF.2_3	TI_OF.2_3	AG	20/03/2024	Passed

Table 1: Test Incident Report

Tutti i test case eseguiti non hanno riportato errori, evidenziando come il software riesca a riconoscere ed analizzare correttamente gli input ricevuti dall'utente, rifiutando quelli non conformi.

3.2 Integration Testing

Il test di integrazione è avvenuto tramite una **strategia bottom-up**. Quindi sono state verificate con successo per prime le componenti di accesso al database, per poi integrare quelle per l'effettiva realizzazione delle funzionalità richieste. Infine, una volta ottenuto l'insieme di queste componenti funzionanti, è stata costruita l'interfaccia grafica per l'utilizzo finale del software.

3.3 System Testing

Una volta ottenuto il software nella sua interezza, con tutte le componenti incluse, è stato effettuato il Testing di Sistema, con l'obiettivo di verificare che il comportamento del software sotto tutti i punti di vista, in base ai requisiti Funzionali e Non Funzionali.

3.3.1 Errori Ricontrati

Questa è stata la fase che ha rilevato più **failure (fallimenti)**.

In particolare, questi riguardano il comportamento del software su sistemi operativi diversi: i tester hanno evidenziato come la **funzionalità di organizzazione** subisce un **inaspettata interruzione** subito dopo il suo avvio, su macchine con sistema operativo macOS.

Questa interruzione, si verifica non appena l'utente preme il tasto "Avvia Organizzazione", non analizzando/spostando nessun file della cartella indicata nella fase precedente fase di inserimento del path e personalizzazione, che quindi funziona correttamente.

Questo ha permesso di individuare l'area in cui agire per risolvere la problematica: ovvero l'avvio del modulo di machine learning.

Dopo un accurato debugging, il **fault** è stato individuato nella funzione adibita a caricare in memoria il classificatore. In particolare si è notato che il **path** del file in cui è contenuto il modello serializzato, ha come sua origine la cartella base di CleanDesk, che però viene vista in posizione diversa in base al sistema operativo; questo path, inizialmente statico per tutte le macchine, risulta impossibile da leggere per le macchine macOS, causando di conseguenza l'errore.

Per **risolvere questa failure**, il path in questione è stato reso dinamico, con il separatore che cambia automaticamente in base al sistema operativo. Ma soprattutto i file del modello e del vectorizer, sono stati spostati dalla cartella madre a quella in cui è contenuto il codice che li utilizza, rendendoli accessibili in maniera diretta.