
Statement of Work

Progetto:

CleanDesk



CleanDesk

Powered by AI

Riferimento:	
Versione:	0.9
Data:	10/02/2024
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
10/10/2023	0.1	Prima stesura	A. Camoia, G. Ambrosio
10/02/2024	0.9	Rimozione Criteri di Premialità	A. Camoia, G. Ambrosio



Indice

1	Piano Strategico / Strategic Plan	3
2	Obiettivi di Business / Business Needs	3
3	Ambito del Prodotto / Product Scope	3
4	Data di Inizio e di Fine	4
5	Deliverables	4
6	Vincoli/Constraints	4
6.1	Vincoli collaborativi e comunicativi	4
6.2	Vincoli Tecnici	4
6.2.1	Analisi e Specifica dei Requisiti	4
6.2.2	System Design	5
6.2.3	Object Design	5
6.2.4	Testing	5
7	Criteri di Accettazione / Acceptance Criteria	5

Statement of Work (SOW) del Progetto CleanDesk

1 Piano Strategico / Strategic Plan

Il Team di CleanDesk intende intervenire nel come le persone utilizzano i personal computer; in particolare il focus è posto nella maniera in cui quest'ultime gestiscono i propri spazi di lavoro, a partire dall'ambiente desktop fino ad arrivare alle cartelle personali, di lavoro, studio o qualsiasi altro spazio.

2 Obiettivi di Business / Business Needs

La volontà alla base del progetto è quella di sopperire alla frequente mancanza di organizzazione che l'utente medio ha quando utilizza il proprio personal computer, poiché, quest'ultima molto spesso si traduce in difficoltà di utilizzo e poca produttività soprattutto per gli utenti meno avvezzi all'utilizzo di dispositivi informatici come il computer.

3 Ambito del Prodotto / Product Scope

Intendiamo fornire ai nostri utenti un servizio che permetta di gestire e accedere ai file all'interno del computer in maniera semplice e veloce. Questo verrà fatto creando strutture organizzate (cartelle) in base ad una classificazione per tipologia e contenuto dei file, effettuata tramite sistemi di Intelligenza Artificiale che, a partire da questi fattori, riescano ad individuare la categoria e l'ambito di utilizzo del file, al fine di posizionarlo in una cartella adeguata.

Il sistema inoltre fornirà strumenti utili alla gestione efficiente dei propri file, implementando funzionalità che permetteranno di:

- Individuare file simili o duplicati.
- Analizzare lo spazio di memoria occupato dalle varie categorie di file.

4 Data di Inizio e di Fine

Inizio: Ottobre 2023

Fine: Gennaio-Febbraio 2024. E' possibile concordare la data di consegna che potrà essere una delle seguenti:

- Fine Gennaio 2024, secondo appello.
- Metà Febbraio 2024, terzo appello.

5 Deliverables

RAD, SDD, ODD, Matrice di Tracciabilità, Test Plan, Test Case Specification, Test incident Report, Test Summary Report, Manuale D'Uso, Manuale Installazione e ogni altro documento richiesto per lo sviluppo del sistema.

6 Vincoli/Constraints

6.1 Vincoli collaborativi e comunicativi

- Rispetto delle scadenze intermedie e di fine progetto definite nello statement of work.
- Budget effort non superiore a 50 ore per ogni membro del team.
- Uso di GitHub come sistema di versioning.
- Uso di tool per la gestione di task e attività (Trello).
- Uso di un tool di comunicazione tracciabile (Slack).

6.2 Vincoli Tecnici

6.2.1 Analisi e Specifica dei Requisiti

- Specifica di minimo 2 e massimo 4 scenari per ogni membro del team.
- Specifica di minimo 2 e massimo 4 requisiti funzionali e non funzionali per ogni membro del team.
- Esattamente uno use case per ogni membro del team.
- Esattamente un sequence diagram ogni due membri del team.

- Esattamente un diagramma a scelta tra statechart e activity diagram ogni due membri del team.
- Specifica di un class diagram per team.

6.2.2 System Design

- Specifica di minimo 2 e massimo 4 design goal per ogni membro del team.
- Definizione di un diagramma di decomposizione dei sottosistemi per team, con annessa descrizione e motivazione all'uso.
- Definizione di un deployment diagram per team, con annessa descrizione e motivazione all'uso.

6.2.3 Object Design

- Uso di minimo uno e massimo due design pattern per team.
- Uso di UML.

6.2.4 Testing

- Ogni studente dovrà effettuare il testing di unità, tramite category partition, di esattamente un metodo di una classe sviluppata.
- Ogni studente dovrà effettuare il testing di sistema, tramite category partition, di esattamente una funzionalità del sistema sviluppato.

7 Criteri di Accettazione / Acceptance Criteria

Criteri che, se non rispettati, portano al fallimento del progetto.

- Utilizzo appropriato di GitHub, che preveda il rispetto delle linee guida definite nel contesto della prima lezione di laboratorio.
- Adeguato utilizzo del pull-based development, che preveda il rispetto delle linee guida definite nel contesto della prima lezione di laboratorio.
- Adeguato utilizzo di Slack, che preveda il rispetto delle linee guida definite nel contesto della seconda lezione di laboratorio.



- Adeguato utilizzo di Trello, che preveda il rispetto delle linee guida definite nel contesto della seconda lezione di laboratorio.
- Fornire una documentazione adeguata. Verranno usati tool di plagiarism detection per identificare casi in cui gli studenti hanno copiato da progetti di anni precedenti e/o da altre fonti.
- Appropriato test di unità di un metodo sviluppato, che prevede il rispetto dei vincoli.
- Appropriato test di sistema di una funzionalità del sistema sviluppato, che preveda il rispetto dei vincoli.