**“Ingegneria del Software”**

**2024-2025**

**Docente: Prof. Angelo Furfaro**

**Gestore libreria personale**

|  |  |
| --- | --- |
| **Data** | <gg/mm/aaaa> |
| **Documento** | Documento Finale – D3 |

|  |  |  |
| --- | --- | --- |
| **Team Members** | | |
| **Nome e Cognome** | **Matricola** | **E-mail address** |
| **Leonardo Napoli** | **234364** | **npllrd02s30d086d@studenti.unical.it** |

Sommario

**Non è stata trovata alcuna voce d'indice.**

## List of Challenging/Risky Requirements or Tasks

|  |  |  |  |
| --- | --- | --- | --- |
| **Challenging Task** | **Date the task is identified** | **Date the challenge is resolved** | **Explanation on how the challenge has been managed** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

A. Stato dell’Arte

Tra le applicazioni maggiormente utilizzate per la gestione di librerie ho potuto testare almeno 2 soluzioni utili in merito. Si tratta di Obsidian e Calibre.

## Obsidian

Obsidian è un’applicazione più generalmente utilizzata per la creazione di note, mappe mentali eappunti personali, ma grazie alla grande possibilità che essa offre in termini di personalizzazione, si può adattare a qualsiasi utilizzo. Uno dei tanti utilizzi da me provati è stato proprio quello di mantenere la mia collezione di libri, con tanto di immagine di copertina, valutazione e commenti.

### Vantaggi

* Personalizzazione praticamente illimitata grazie ai plugin open-source;
* Possibilità di creare collegamenti tra libri e/o commenti;
* Informazioni memorizzate in locale tramite file .md;

### Svantaggi

* La gestione di una libreria devia dallo scopo generale dell’applicazione;
* Complessa inizializzazione da parte di un utente non tecnico;

## Calibre

Calibre è un’applicazione open-source orientata alla gestione di Ebook, offre strumenti molto utili come la conversione di formati, gestione di metadati e sincronizzazione tra dispositivi

### Vantaggi

* Software maturo ed open-source;
* Estendibile con plugin;
* Totalmente in locale;

### Svantaggi

* L’applicazione risulta inutile per la gestione di libri cartacei;
* Eccessive funzionalità offerte rendono l’interfaccia molto complessa.

## Considerazioni

Osservando punti di forza e di debolezza delle applicazioni menzionate si è potuto derivare alcune funzionalità utili che saranno implementate nel sistema:

* Salvataggio dei dati in locale ed esportabile;
* Gestione semplice tramite interfaccia grafica;
* Funzioni limitate per un approccio minimale;
* Possibilità di aggiunta di funzionalità aggiuntive.

B. Raffinamento dei Requisiti

*A partire dai servizi minimali richiesti, raffinate la descrizione dei servizi offerti dal vostro applicativo. Descrivete anche I requisiti non funzionali.*

***A.1 Servizi (con prioritizzazione)***

*Descrivete in* ***dettaglio*** *i servizi offerti dal vostro Sistema, insieme a quelli che ritenete siano le soluzioni concettuali necessarie. In questa fase, non fate riferimento ad alcuna tecnologia specifica. Se volete, intervistate stakeholder e collezionate dati dal web o da altre sorgenti. Dovete acquisire una conoscenza avanzata dei problemi associate ai vostri servizi. Assegnate un ID a ciascun servizio. Prioritizzate inoltre I servizi in base a due scale: importanza alta, media, bassa. Complessità alta, media, bassa.*

***A.2 Requisiti non Funzionali***

*Elencare i requisiti non funzionali più’ importanti per il vostro Sistema*

***A.3 Scenari d’uso dettagliati***

*Descrivere gli scenari più comuni, più interessanti, o più complicati d’uso dei vostri servizi.*

***A.4 Excluded Requirements***

*Descrivere i servizi eventualmente i esclusi, e spiegare il perchè*

***A.5 Assunzioni***

*<Briefly document, in this section, the most relevant requirement assumptions/decisions you had to made during your project>*

***A.6 Use Case Diagrams***

C. Architettura Software   
*<IF RELEVANT, Report here both the static and the dynamic view of your system design, in terms of a Component Diagram, and their related Sequence Diagrams >*

***C.1 The static view of the system: Component Diagram***

***C.2 The dynamic view of the software architecture: Sequence Diagram***

D. Dati e loro modellazione (se il sistema si interfaccia con un DBMS)

*Definite le sorgenti di dati a voi necessarie per realizzare I servizi di cui sopra. Modellate tali dati tramite un ER o similari. Specificate se e quali di tali dati sono gia’ forniti da applicativi esistenti.*

E. Scelte Progettuali (Design Decisions)   
<Document here the **5** most important design decisions you had to take. You can use both a textual or a diagrammatic specification.>

F. Progettazione di Basso Livello

G. Spiegare come il progetto soddisfa i requisiti funzionali (FRs) e quelli non funzionali (NFRs) *<Report in this section how the architectural and low level design you produced satisfies the FRs and the NFRs>*

Appendix. Prototype   
*<Provide a brief report on your prototype, and especially: information on what you have implemented, how the implementation covers the FR and NFR, how the prototypes demonstrates your project correctness with respect to the FR and NFR. You may add some screenshots to describe what required above. Be ready to show your prototype during the oral examination>*