

WBS

Recommendation Engine Subsystem

- [#1] Setup del repository
- [#2] Definizione delle regole per la qualità del codice
- [#3] Setup dei framework necessari all'implementazione del microservizio
- [#4] Setup della build automation e delle pipeline di CI/CD
- [#5] Creazione documentazione relativa al codice sviluppato

Requisiti Funzionali:

Search Functionality

1.1.1 Il sistema deve rispondere a delle interrogazioni basate su parole chiave, restituendo un elenco di documenti che meglio rispondono all'interrogazione, tenendo conto anche delle preferenze e dell'esperienze dell'utente.

1.1.2 L'interrogazione deve poter essere raffinata per mezzo di filtri

- [#6] Design e documentazione della chiamata REST
- [#7] Implementazione della funzione per la ricerca nel sottosistema di storage
- [#8] Generazione di un dataset per il training di un modello di machine learning
- [#9] Individuazione della tecnica opportuna di ML per il raffinamento e la personalizzazione dei risultati di ricerca
- [#10] Training del modello
- [#11] Testing & Validation

Recommendation Engine

1.2.1 Sulla base delle note presenti nel sistema, combinate con le informazioni circa le preferenze espresse dall'utente, i feedback e le valutazioni espresse dagli utenti del sistema, il sistema deve fornire un elenco di documenti che potrebbero essere di utilità e interesse all'utente.

- [#12] Design e documentazione della chiamata REST
- [#13] Individuazione dei dati utili alla funzione di raccomandazione
- [#14] Implementazione delle funzioni di integrazione con i vari microservizi
- [#15] Generazione di un dataset per il training di un modello di machine learning
- [#16] Individuazione dell'algoritmo best-fit per la funzione di raccomandazione
- [#17] Training del modello
- [#18] Testing & Validation

Requisiti Non Funzionali:

Performance

2.1.1 Il sistema deve restituire i risultati delle ricerche entro 1 secondo.

2.1.2 Il sistema aggiorna le raccomandazioni all'utente in tempo reale.

- [#19] Le attività per soddisfare i requisiti di performance, verranno identificate durante la fase di sviluppo

User Authentication and Profile Management Subsystem

- [#20] Setup del repository
- [#21] Definizione delle regole per la qualità del codice
- [#22] Setup dei framework necessari all'implementazione del microservizio
- [#23] Setup della build automation e delle pipeline di CI/CD
- [#24] Creazione documentazione relativa al codice sviluppato
- [#25] Progettazione del database

Requisiti Funzionali:

User Registration

1.1.1 Il sistema deve permettere agli utenti di registrarsi e creare un nuovo account.

- [#26] Design e documentazione dell'API
- [#27] Implementazione delle funzioni di salvataggio
- [#28] Testing

Login and Authentication

1.2.1 Il sistema deve consentire un accesso sicuro mediante autenticazione multi-fattore (MFA).

- [#29] Implementazione delle funzioni necessarie all'invio di sms
- [#30] Implementazione dell'autenticazione a multi fattore

1.2.2 Il sistema deve consentire l'accesso attraverso delle credenziali universitarie.

- [#31] Individuazione delle API del sistema di autenticazione dell'ateneo
- [#32] Implementazione delle funzioni per l'autenticazione attraverso le API esterne

Profile Management

1.3.1 Il sistema deve permettere la modifica delle informazioni del profilo da parte degli utenti, permettendo loro di aggiungere, oltre a Nome e Cognome e data di nascita, anche il corso di studi frequentato, i corsi di maggiore interesse e i voti degli esami conseguiti.

- [#33] Design e documentazione dell'API
- [#34] Implementazione delle funzioni di modifica e salvataggio
- [#35] Testing

Requisiti Non Funzionali:

Security

2.1.1 Il sistema deve rispettare le norme in materia di protezione dei dati personali in conformità al GDPR.

- [#36] Le attività necessarie al soddisfacimento del requisito verranno individuate durante la fase di sviluppo

Frontend Subsystem WBS

1. Requisiti Funzionali

1.1. Dashboard

- 1.1.1. Il sistema deve permettere di visualizzare le note personali di un utente.
 - [#37] Design dell'interfaccia grafica
 - [#38] Implementazione dell'interfaccia grafica
 - [#39] Scrittura del codice di interfacciamento con il backend
 - [#40] Testing
- 1.1.2. Il sistema deve permettere la visualizzazione delle raccomandazioni personalizzate.
 - [#41] Design dell'interfaccia grafica
 - [#42] Implementazione dell'interfaccia grafica
 - [#43] Scrittura del codice di interfacciamento con il backend
 - [#44] Testing

1.2. Notes Repository Interface

- 1.2.1. Il sistema deve poter permettere di creare, leggere, modificare e cancellare documenti.
 - [#49] Design dell'interfaccia grafica
 - [#50] Implementazione dell'interfaccia grafica
 - [#51] Scrittura del codice di interfacciamento con il backend
 - [#52] Testing
- 1.2.2. Il sistema deve permettere l'utilizzo di filtri e strumenti di ricerca avanzata.
 - [#53] Design dell'interfaccia grafica
 - [#54] Implementazione dell'interfaccia grafica
 - [#55] Scrittura del codice di interfacciamento con il backend
 - [#56] Testing

1.3. Rating System & Feedback Interface

- 1.3.1. Il sistema deve permettere di valutare i documenti (1-5 stelle)
 - [#57] Design dell'interfaccia grafica
 - [#58] Implementazione dell'interfaccia grafica
 - [#59] Scrittura del codice di interfacciamento con il backend
 - [#60] Testing
- 1.3.2. Il sistema deve permettere l'inserimento di commenti e recensioni alle note pubblicate dagli utenti.
 - [#61] Design dell'interfaccia grafica
 - [#62] Implementazione dell'interfaccia grafica
 - [#63] Scrittura del codice di interfacciamento con il backend
 - [#64] Testing
- 1.3.3. Il sistema deve consentire la moderazione dei commenti
 - [#65] Design dell'interfaccia grafica
 - [#66] Implementazione dell'interfaccia grafica
 - [#67] Scrittura del codice di interfacciamento con il backend
 - [#68] Testing

1.4. User Authentication and Profile Management Interface

- 1.4.1. Il sistema deve permettere di registrarsi e accedere al sistema mediante opportune form.
 - [#69] Design dell'interfaccia grafica
 - [#70] Implementazione dell'interfaccia grafica
 - [#71] Scrittura del codice di interfacciamento con il backend
 - [#72] Testing
- 1.4.2. Il sistema deve permettere l'utilizzo di sistemi di Autenticazione multi-fattore (MFA).
 - [#73] Design dell'interfaccia grafica
 - [#74] Implementazione dell'interfaccia grafica
 - [#75] Scrittura del codice di interfacciamento con i sistemi MFA
 - [#76] Testing
- 1.4.3. Il sistema deve consentire la modifica delle informazioni del profilo
 - [#77] Design dell'interfaccia grafica

- [#78] Implementazione dell'interfaccia grafica
 - [#79] Scrittura del codice di interfacciamento con il backend
 - [#80] Testing
- 1.4.4. Il sistema deve permettere di gestire le impostazioni di privacy e sicurezza
- [#81] Design dell'interfaccia grafica
 - [#82] Implementazione dell'interfaccia grafica
 - [#83] Scrittura del codice di interfacciamento con il backend
 - [#84] Testing

2. Requisiti Non Funzionali

2.1. Usability

- 2.1.1. Esperienza utente deve essere uniforme su tutti i dispositivi
- [#85] Organizzazione di sessioni di Focus Group
 - [#86] Organizzazione di sessioni di Experience-prototyping
 - [#87] Stesura e somministrazione di questionari sulla UX
- 2.1.2. Il Design dell'applicazione deve essere *responsive* e adattarsi alle dimensioni dei vari dispositivi.
- 2.1.3. Il sistema deve essere compatibile con i principali browser (Edge, Google Chrome, Firefox)
- Le attività necessarie al soddisfacimento del requisito verranno individuate durante la fase di sviluppo.

2.2. Performance

- 2.2.1. I tempi di risposta alla ricerca devono essere inferiori a 1 secondo nel 99% dei casi
- 2.2.2. Il tempo per la visualizzazione dei documenti deve essere inferiore a 2 secondi nel 99% dei casi.
- Le attività necessarie al soddisfacimento dei requisiti verranno individuate durante la fase di sviluppo.

3. Requisito di integrazione

3.1. API Integration

- 3.1.1. Il sistema deve integrarsi con sottosistema di storage (Notes Repository Subsystem).
- 3.1.2. Il sistema deve integrarsi con il sistema di autenticazione e gestione dei profili (User Authentication and Profile Management Subsystem).
- 3.1.3. Il sistema deve Integrarsi con il motore di raccomandazione (Recommendation Engine Subsystem).
- Le attività necessarie al soddisfacimento dei requisiti **sono coperte** dalle attività previste per i requisiti funzionali.