

DOIT

Progetto di Ingegneria del Software

Gruppo FVF

Autori: Flavio Fabbri – Carlo Verdini – Martina Ferri

Indice

- 1. Vision**
- 2. Attori**
 - 2.1 Proponente
 - 2.2 Progettista
 - 2.3 Esperto
 - 2.4 Utente
- 3. Casi d'uso**
 - 3.1 Modifica un progetto
 - 3.2 Elimina un progetto
 - 3.3 Valuta una candidatura:
 - 3.4 Valuta un utente
 - 3.5 Chiede una valutazione
 - 3.6 Chiude il progetto
 - 3.7 Invia una candidatura
 - 3.8 Visualizza una candidatura
 - 3.9 Modifica una candidatura
 - 3.10 Elimina una candidatura
 - 3.11 Crea curriculum
 - 3.12 Modifica curriculum
 - 3.13 Elimina curriculum
 - 3.14 Autenticazione
 - 3.15 Logout
 - 3.16 Crea un progetto
 - 3.17 Valuta una richiesta ricevuta
 - 3.18 Visualizza utenti
 - 3.19 Visualizza elenco progetti
 - 3.20 Visualizza progetto
 - 3.21 Visualizza curriculum
 - 3.22 Invia una richiesta
 - 3.23 registrazione
- 4. Iterazioni**
 - 4.1 Prima Iterazione
 - 4.2 Seconda Iterazione
 - 4.3 Terza Iterazione
 - 4.4 Quarta Iterazione
- 5. Progetto**

1. Vision

Progettazione collaborativa e matching di competenze

La piattaforma vuole promuovere lo svolgimento collaborativo di progetti innovativi e facilitare l'inserimento delle persone all'interno degli stessi progetti sulla base delle loro competenze. Allo stesso tempo vuole realizzare una vetrina dei progetti svolti in cui un qualsiasi cittadino sia dunque capace di ricercare e visualizzare le informazioni sui progetti e chi li ha svolti. In questo caso le funzionalità che si immaginano saranno fornite dalla piattaforma e si vogliono soddisfare le esigenze dei seguenti attori.

Attraverso questa piattaforma determinati utenti hanno la possibilità di creare dei progetti: in questo modo gli utenti, dopo aver creato un progetto diventano proponenti di quest'ultimo.

Le figure competenti che hanno i requisiti richiesti per lo svolgimento del progetto possono effettuare una candidatura; se quest'ultima sarà accettata diventeranno progettisti di quel progetto.

Gli utenti possono creare un proprio curriculum contenente le proprie informazioni personali e lavorative; verrà aggiunto nel curriculum anche ogni progetto che è stato portato a termine.

Gli utenti hanno la possibilità di mandare richieste ad altri utenti per partecipare ai progetti.

I progettisti, prima di iniziare un nuovo progetto possono chiedere un consulto ad un'altra figura presente, ossia, gli esperti: questi hanno il compito di valutare il progetto.

Inoltre, anche gli esperti, possono valutare gli utenti dopo lo svolgimento dei progetti.

I proponenti dei progetti hanno il compito di valutare le candidature dei progettisti, accettarle o meno e valutare i progettisti dopo aver svolto il lavoro.

2. Attori

→ Proponente:

Sono utenti o organizzazioni che vogliono inserire proposte di progetti da far svolgere a persone competenti in un determinato settore. Durante la creazione del progetto, il proponente inserirà tutte le informazioni principali al fine di identificare le persone migliori per attuare gli obiettivi preposti.

Il proponente riceve le richieste di candidatura dai progettisti che ritengono di essere in grado e valuta se accettare o meno le candidature.

Inoltre, i proponenti possono inviare richieste di candidatura ad utenti ritenuti idonei.

I proponenti possono affidarsi a figure esterne esperte per la valutazione del proprio progetto.

→ Progettisti:

I progettisti sono soggetti aventi determinati requisiti richiesti, specifiche competenze e possono candidarsi ai progetti inseriti nella piattaforma.

Hanno interesse dunque ad avere un ampio curriculum dove inseriscono le proprie generalità e competenze al fine di poter essere selezionati o comunque al fine di supportare le proprie competenze.

La piattaforma mantiene traccia dei progetti svolti che contribuiscono al curriculum del progettista.

Il progettista inoltre, può essere valutato dal proponente o da un esperto dopo aver svolto un progetto.

→ Esperti:

(Valutatori) sono persone fisiche che possono dare un giudizio sulla proposta di un progetto o su eventuali progettisti, i quali hanno svolto un progetto.

→ Utente: persona fisica, ente o organizzazione che utilizza la piattaforma: ha la possibilità di creare un proprio curriculum, visualizzare tutti i progetti presenti nella piattaforma e può inviare richieste.

3. Casi d'uso

CASI D'USO PROPONENTE

1. modifica un progetto: il proponente di un progetto può modificare un progetto precedentemente creato, cambiando i campi inseriti

2. elimina un progetto: il proponente di un progetto può eliminare un progetto da lui inserito nella piattaforma e quindi non renderlo più disponibile ai progettisti

3. valuta una candidatura: il proponente del progetto valuta se il progettista che ha inviato la candidatura ha i requisiti necessari allo svolgimento del progetto

4. valuta un utente: un utente dopo aver svolto un progetto con un altro utente, può esprimere una valutazione nei suoi riguardi

5. chiede una valutazione: il proponente di un progetto può voler sottoporre soggetti esterni alla valutazione del proprio progetto in fase iniziale, ossia capire se un determinato progetto è fattibile o meno

6. chiude il progetto: il proponente ha la facoltà di chiudere il progetto dopo averlo terminato oppure se ha intenzione di sospendere lo svolgimento. terminato il progetto quest'ultimo verrà inserito automaticamente nel curriculum del progettista e del proponente

CASI D'USO PROGETTISTA

1. invia una candidatura: se un progettista trova un progetto a lui interessante e di sua competenza invia la propria candidatura al proponente che in secondo tempo la valuterà

2. visualizza una candidatura: il progettista ha la possibilità di visualizzare una candidatura che ha inviato

3. modifica una candidatura: il progettista ha la possibilità di modificare una candidatura che ha inviato

4. elimina una candidatura: il progettista ha la possibilità di eliminare una candidatura

CASI D'USO ESPERTO

1. invia una valutazione: gli esperti vengono contattati dai proponenti per valutare una proposta di progetto; gli esperti indicano la loro disponibilità e successivamente inviano una recensione riguardo la proposta valutata
2. visualizza una valutazione: gli esperti possono visualizzare le valutazioni che hanno inviato
3. elimina una valutazione: gli esperti possono eliminare le valutazioni che hanno inviato
4. modifica una valutazione: gli esperti possono modificare una valutazione
5. valuta un utente: gli esperti possono valutare degli utenti dopo che aver svolto il progetto

CASI D'USO UTENTE

1. crea curriculum: un utente inserisce le proprie informazioni personali e professionali richieste e vengono salvate nel suo curriculum
2. modifica curriculum: un utente può modificare le informazioni inserite nel proprio curriculum
3. elimina curriculum: un utente può eliminare il proprio curriculum (con eventuale cancellazione dalla piattaforma)
4. autenticazione: un utente può accedere con le proprie credenziali
5. logout: l'utente può decidere di scollegarsi dalla piattaforma.
6. crea un progetto: un utente che desidera svolgere un progetto, inserisce la propria proposta nella piattaforma; durante l'inserimento della proposta indica le *caratteristiche principali* del progetto che verrà svolto
 nome del progetto, città dove verrà svolto il progetto, scadenza del progetto (ossia quanto tempo è necessario per lo svolgimento), posti disponibili (quante persone occorrono per lo svolgimento), scopo (se di lucro o meno), categoria e una descrizione contenente gli obiettivi del progetto
 quindi l'utente diventerà un "proponente", il quale, salva la proposta e questa viene inserita nella piattaforma

7.valuta una richiesta ricevuta: l'utente potrebbe ricevere da un proponente una richiesta per partecipare allo svolgimento di un progetto, lui può accettare o rifiutare; accettando diventa un "progettista" di quel progetto

8.visualizza utenti: l'utente decide di visualizzare l'elenco degli utenti registrati sulla piattaforma scegliendo precedentemente le informazioni di proprio interesse riguardo l'utente da cercare.

9.visualizza elenco progetti: l'utente decide di visualizzare la lista di tutti i progetti che sono all'interno della piattaforma sulla base di alcune informazioni specifiche da lui precedentemente inserite

10.visualizza progetto: l'utente decide di visualizzare la specifica del progetto selezionato e tutte le informazioni contenenti

11.visualizza curriculum: l'utente decide di visualizzare il curriculum contenente le informazioni personali quali: nome del progetto, città dove verrà svolto il progetto, scadenza del progetto (ossia quanto tempo è necessario per lo svolgimento), posti disponibili (quante persone occorrono per lo svolgimento), scopo (se di lucro o meno), categoria e una descrizione contenente gli obiettivi del progetto

12.invia una richiesta: un utente invita un altro utente alla partecipazione del progetto

13. registrazione: un utente inserisce nome cognome utente e password ed invia la richiesta di registrazione

4. ITERAZIONI

1.PRIMA ITERAZIONE (18 novembre - 15 dicembre)

Primo incontro: Discussione del progetto e come dev'essere implementato

Secondo incontro: Analisi dei requisiti: vengono stabiliti gli attori principali e UC ad essi associati

Terzo incontro: Descrizione breve dei casi d'uso

Quarto incontro: Divisione del progetto in iterazioni a loro volta divise in -modelli di dominio- e -modelli di progettazione-

Quinto incontro: analisi del diagramma delle classi

Sesto incontro: creazione del diagramma delle classi di progetto

➔ Nessun UC implementato nella prima iterazione

2.SECONDA ITERAZIONE (16 dicembre - 15 gennaio)

Primo incontro: controllo generale dei casi d'uso ed eventuali modifiche

Secondo incontro: implementazione dei casi d'uso scelti elaborando il flusso di sequenza

Terzo incontro: creazione dei diagrammi di sequenza degli UC scelti

Quarto incontro: implementazione dei casi d'uso in base ai modelli attraverso le tecnologie scelte

Quinto incontro: discussione sull'implementazione dei casi d'uso e eventuali modifiche

➔ Nella Seconda iterazione vengono implementati i seguenti casi d'uso:

1. -crea curriculum
2. -modifica curriculum
3. -elimina curriculum
4. -visualizza curriculum
5. -visualizza elenco progetti
6. -visualizza utenti
7. -visualizza progetti
8. -crea un progetto
9. -modifica un progetto
10. -elimina un progetto

3.TERZA ITERAZIONE (16 gennaio - 15 febbraio)

Primo incontro: ampliamento dei casi d'uso

Secondo incontro: descrizione elaborata dei UC

Terzo incontro: creazione dei diagrammi di sequenza

Quarto incontro: implementazione dei casi d'uso scelti

Quinto incontro: modifiche riguardo l'implementazione del codice

Sesto incontro: revisione dell'implementazione e risoluzione di eventuali errori

➔ Nella terza iterazione vengono implementati i seguenti casi d'uso

1. -invia una candidatura
2. -visualizza una candidatura
3. -modifica una candidatura
4. -elimina una candidatura
5. -invia una richiesta
6. -chiede una valutazione
7. -invia una valutazione
8. -modifica una valutazione
9. -elimina una valutazione
10. -visualizza una valutazione

4.QUARTA ITERAZIONE (15 febbraio - 24 febbraio)

Primo incontro: descrizione degli ultimi uc e flusso di sequenza

Secondo incontro: creazione di diagrammi di sequenza e inizio dell'implementazione

Terzo incontro: implementazione e revisione finale del progetto con eventuali modifiche

Quarto incontro: modifiche implementative

Quinto incontro: chiusura del progetto e completamento della relazione

➔ I casi d'uso implementati nella quarta iterazione:

1. -chiude il progetto
2. -valuta un utente
3. -valuta una candidatura
4. -valuta una richiesta
5. -autenticazione
6. -logout
7. -registrazione

5. PROGETTO

Il progetto DOIT è stato sviluppato seguendo il pattern MVC su architettura REST API.

Il progetto si basa su tecnologia Java Spring Boot e la persistenza dei dati avviene su DBMS Mysql attraverso connettore mysql-connector-java ed accesso ai dati tramite JPA spring-boot-starter-jpa.

Attraverso JPA è l'applicazione stessa in modalità hibernate ddl update che permette di adeguare la modellazione del DB in coerenza con la classe model e di gestire l'integrità referenziale tra le entità.

L'applicazione prevede un endpoint pubblico public/login da cui effettuare l'autenticazione e recuperare il token di sessione, tutto il resto delle chiamate sono in modalità protected e richiedono il token di sessione. Le regole sono specificate nella classe **WebSecurityConfig**.

Per ogni entità legata ai casi d'uso è stata creata una rappresentazione model repository controller:

1. Il **model** contiene la struttura dell'entità con le relative proprietà e metodi di gestione (getter, setter)
2. Il **repository** estende il JpaRepository in modo da ereditare le CRUD e contiene anche dei metodi specifici per il recupero di liste o oggetti in base a criteri specifici funzionali ai vari casi d'uso
3. Nel **controller** sono dichiarati i metodi pubblici e la logica di controllo che comprende controlli consistenza e gestione dei vincoli (solo determinati utenti può effettuare determinate operazioni)

Tutte le operazioni di creazione modifica relative a curriculum, candidatura, valutazione, nomina esperto prevedono un'auto referenziazione dell'utente loggato in modo da garantire la paternità dell'entità ad esempio nel metodo di creazione candidatura di seguito riportato si nota che viene preso come riferimento della candidatura l'utente di sessione.

```
@PostMapping("/addCandidatura")
public String addCandidatura(@RequestBody Candidatura candidatura){
    //associo utente a candidatura
    UserDetails ud = (UserDetails)
        SecurityContextHolder.getContext().getAuthentication().getPrincipal();
    Utente user = uRep.findByUsername(ud.getUsername());
    candidatura.setCandidato(user);
    //procedo con la candidatura e la imposto come attiva
    candidatura.setStato(Candidatura. ACTVD);
    cRep.save(candidatura);
    return "candidatura creata correttamente";
}
```

Sempre dallo stesso esempio si deduce l'assegnazione implicita dello stato della candidatura che, nel caso specifico viene automaticamente impostata come attiva ("ACTVD").

Di seguito la tabella degli stati della candidatura e gli attori che possono gestirli:

Attore	Codice Stato	Descrizione
qualsiasi utente	RQSTD	Proposta di Candidatura, un utente qualsiasi può suggerire ad un altro utente di candidarsi, compreso il proponente o l'esperto
candidato	ACTVD	Attivazione candidatura
	CNCLD	Ritiro candidatura
Valutatore (proponente, esperto nominato)	CFRMD	Conferma candidatura (ruolo progetto affidato)
Valutatore (proponente, esperto nominato)	RJCTD	Scarto candidatura (ruolo progetto non affidato)

Per la gestione degli stati sono state implementate delle proprietà `public static final String` in modo da definire il dominio degli stati all'interno della classe.

Nel progetto è stata aggiunta l'entità **ruolo progetto** su cui si basano tutte le dinamiche di candidatura e valutazione. Il motivo è dato dal fatto che il progetto può essere di diversa complessità quindi il proponente nel progetto specifica dei ruoli per i quali vengono aperte delle posizioni.

Il candidato che visiona il progetto si candida ad uno o più ruoli del progetto. Anche l'eventuale esperto, nominato dal proponente, viene ingaggiato per valutare candidature relative ad uno o più ruoli a lui assegnati in quanto, considerando un progetto complesso, potrebbero essere necessari più esperti per valutare specifici ruoli.

Il ruolo progetto, ha 3 semplici stati:

DRAFT in bozza, il proponente deve completare/rivedere la presentazione
 OPENED aperto, quindi accetta candidature
 CLOSED chiuso, non accetta candidature

AMBIENTE DI SVILUPPO

Per realizzare l'applicazione abbiamo utilizzato:

Back-end: IntelliJ IDEA

Front-end: React

DBMS: MAMP / XAMP

Frontend: Visual Studio Code

Tunneling: ngrok