

Documentazione Hackathon – Fase 2

Versione: 2.0

Data: Maggio 2025

Autori: Alessio Somma N86005057, Valerio Malvone N86005228

Sommario

- 1. Introduzione
- 2. Evoluzione rispetto alla Fase 1
- 3. Stato del Controller
- 4. Stato dell'interfaccia grafica (Swing)
- 5. Nuove entità e aggiornamento del Class Diagram
- 6. Considerazioni progettuali
- 7. Prossimi sviluppi

1. Introduzione

Questa seconda fase del progetto Hackathon ha avuto l'obiettivo di iniziare a costruire l'infrastruttura operativa dell'applicazione, seguendo il modello definito nella prima fase. In particolare, sono stati impostati i primi metodi del controller e avviata la costruzione dell'interfaccia grafica in Java Swing.

2. Evoluzione rispetto alla Fase 1

Nella Fase 1 si era realizzata la modellazione concettuale del sistema con un Class Diagram UML del problema e una descrizione testuale delle entità principali (Hackathon, Utente, Giudice, Partecipante, Organizzatore, Team, Documento, Voto), delle responsabilità e delle relazioni introducendo il package Model con le entità appartenenti.

Durante questa seconda fase, si è cominciato a sviluppare la parte relativa al controller e alle prime interfacce grafiche utente.

3. Stato del Controller

Il controller rappresenta l'intermediario tra la GUI ed il Model. Sono stati definiti i metodi principali, ma molti sono ancora in fase iniziale o non completamente implementati, in attesa dell'integrazione con la GUI e successivamente con il database.

Metodi previsti (al momento solo dichiarati o parziali):

- creaHackathon()
- invitaGiudice()
- apriRegistrazioni()
- registrati(), accedi(), iscriviti()
- creaTeam(), invitaPartecipanti()
- inviaRichiestaIngressoTeam()
- accettaORifiutaInvitoTeam()
- pubblicaProblema(), assegnaVoto()
- pubblicaDocumento(), stampaClassifica()

4. Stato dell'interfaccia grafica (Swing)

L'interfaccia grafica è stata parzialmente realizzata con Java Swing, seguendo un'architettura a schermate per i diversi ruoli. In base al ruolo associato all'account che accede, verrà mostrata la home di sua appartenenza.

- Organizzatore: è stata completata solo la Home, mentre le altre schermate sono in fase di sviluppo;
- Partecipante: sono state create le principali funzionalità grafiche, ma manca l'iscrizione all'Hackathon;
- Giudice: non ancora iniziata la realizzazione della GUI;
- Accesso: l'interfaccia grafica ed i collegamenti alle varie home dei singoli ruoli sono stati sviluppati. Al momento, data l'assenza di un Sistema di persistenza dei dati, l'accesso viene sempre eseguito.
- Registrazione: L'interfaccia grafica è stata implementata ma il collegamento al controller è ancora in fase di sviluppo.

5. Nuova entità 'Invito' e aggiornamento del Class Diagram

Durante questa fase è stata aggiunta una nuova entità: 'Invito', non presente nella prima versione del diagramma. Essa rappresenta l'invito tra partecipanti per formare un team, e modella la richiesta come una classe con relazioni multiple:

- Un partecipante può inviare o ricevere inviti da altri partecipanti
- L'invito è associato anche a un Team potenziale

La struttura ereditaria tra Utente, Giudice, Partecipante e Organizzatore è rimasta invariata.

6. Considerazioni progettuali

- La logica nei metodi del controller è per ora semplificata e non prevede accesso a dati persistenti.
- La progettazione del sistema segue una logica orientata agli oggetti, facendo uso intensivo di concetti come incapsulamento, ereditarietà e polimorfismo, al fine di favorire il riutilizzo del codice, ridurre la ridondanza e rendere il sistema facilmente scalabile.
- L'introduzione dell'entità Invito ha reso il sistema più fedele alle dinamiche sociali della progettualità.
- Sul repository è presente un file readme contenente tutte le informazioni utili ai fini di test e visione dell'applicativo.

7. Prossimi sviluppi

I prossimi passi previsti per il completamento del progetto sono:

- Completare tutte le interfacce grafiche per i tre ruoli (Giudice, Partecipante, Organizzatore);
- Integrare la logica nel controller con le GUI;
- Collegare l'applicativo ad un Sistema di persistenza dei dati, così da ultimare gli sviluppi delle funzioni presenti nel package Controller;
- Implementare un sistema di autenticazione reale;
- Migliorare la validazione dei dati e la gestione degli errori.