**DESCRIZIONE GENERALE:**

Il sistema rappresentato ha l’obiettivo di gestire lo svolgimento di un Hackathon, ovvero delle olimpiadi di hacking.

Durante l’evento, i vari team di partecipanti hanno l’obiettivo di progettare e implementare soluzioni innovative in un intervallo di tempo limitato.

Ora procediamo a descrivere le singole classi, focalizzandoci su: attributi, metodi e relazioni di ogni singola classe.

CLASSE: ***Utente***

Rappresenta una persona che può registrarsi sulla piattaforma e partecipare ad un Hackaton

**Attribut**i:

* **IdUtente: int** 🡪 identificativo univoco per l’utente [Visibilità protected per permetterne l’estensione alle classi “figlie” senza che altre classi abbiano accesso a questo attributo]

**Metodi:**

* **registraSuPiattaforma()** 🡪 permette ad un utente di registrarsi

**Relazioni con altre classi:**

* E’ la superclasse delle classi Giudice e Organizzatore (questa scelta è stata motivata dal fatto che i giudici e l’organizzatore sono versioni estese dell’utente con funzionalità in più).
* **Team**: Può far parte di un team
* **Piattaforma**: Si registra sulla piattaforma

CLASSE: ***Giudice*** (Estende Utente)

Utente che ha il compito di commentare e valutare i documenti caricati sulla piattaforma dai team

**Attributi:** Nessun attributo aggiuntivo rispetto alla classe utente (eredita id utente)

**Metodi:**

* **pubblicaDescrizione()**🡪 Pubblica il problema da risolvere
* **esaminaDocumento()**🡪Valuta i documenti caricati dai team.

**Relazioni con altre classi:**

* **Organizzatore**: È selezionato da un organizzatore
* **Documento**: Esamina e commenta i documenti caricati sulla piattaforma dai team
* **Voto**: Assegna i voti che vengono caricati su una piattaforma

CLASSE ASSOCIATIVA : ***Voto***

**Classe associativa** che rappresenta il voto espresso da un **Giudice** a un **Team**, all’interno della **Piattaforma**. Arricchisce la relazione con il valore numerico del voto e altre informazioni identificative.

**Attributi:**

* **valoreVoto: int**🡪 Punteggio assegnato dai giudici
* **idGiudice: int** 🡪 Identificativo del giudice votante
* **nomeSquadra: String**🡪 Nome del team votato

**Relazioni con altre classi:**

* **Giudice**: Assegna il voto
* **Piattaforma:** il voto viene salvato nella piattaforma

CLASSE: ***Organizzatore*** (Estende Utente)

Utente speciale che ha il compito di organizzare l’Hackathon, aprendo le registrazioni e selezionando i giudici

**Attributi: :** Nessun attributo aggiuntivo rispetto alla classe utente (eredita id utente)

**Metodi:**

* **selezionaGiudici()**:🡪 Seleziona i giudici che valuteranno i progetti
* **apriRegistrazioni()**🡪 Avvia il periodo di registrazione per l’Hackaton

**Relazioni con altre classi:**

* **Giudice:** l’organizzatore seleziona i giudici

CLASSE: ***Team***

Rappresenta un gruppo collaborativo formato da utenti registrati, che lavorano a una soluzione da presentare.

**Attributi:**

* dimensioneMaxTeam: int🡪 numero massimo di utenti in un team
* nomeTeam: string [Visibilità pubblica per permetterne il riutilizzo quando i giudici assegnano il voto]

**Metodi:**

* ***progettaSoluzione()***

**Relazioni con le altre classi:**

* **Utente**🡪 Un team è formato da uno o più utenti
* **Documento**🡪 Viene elaborato dal team e caricato sulla piattaforma
* **Piattaforma**🡪 Utilizzata dal team per caricare il documento

CLASSE ASSOCIATIVA: ***Documento***

**Classe associativa** che rappresenta l’associazione tra un **Team** e una **Piattaforma**. Il team carica uno o più documenti come soluzione al problema. Oltre alla relazione, mantiene informazioni proprie (nome, versione, orario).

**Attributi:**

* **nomeDocumento: String** 🡪 Nome del file caricato
* **versioneRilascio: float**🡪 Versione del documento
* **orarioAggiornamento: String**🡪 Orario della consegna del documento

**Metodi:**

* Nessun metodo proprio, ma viene utilizzato dal giudice per la valutazione

**Relazioni con altre classi**:

* **Team**: Carica il documento
* **Piattaforma**: il documento è caricato all’interno della piattaforma
* **Giudice**: esamina il documento per assegnare un voto

CLASSE: ***Piattaforma***

Gestisce tutto il processo dell’hackathon: registrazione, gestione team, documenti e voti.

**Attributi:**

* **numeroMaxIscritti: int**🡪 numero massimo di partecipanti all’evento

**Metodi:**

* **pubblicaClassifica()**🡪 Pubblica la classifica dei team
* **acquisisciVoto()**🡪 Riceve i voti assegnati dai giudici

**Relazioni con altre classi:**

* **Utente**: Si registra sulla piattaforma
* **Team**: I team partecipano tramite la piattaforma
* **Giudice/Documento/Voto**: Tutti gli elementi sono gestiti nel contesto della piattaforma