



UNIVERSITÀ DEGLI STUDI DI NAPOLI  
**FEDERICO II**

DOCUMENTAZIONE PER PROGETTAZIONE  
OBJECT ORIENTATION  
Progetto in Carico: Hackathon

**CdL Triennale in Informatica**

CORSO DI OBJECT ORIENTATION

GIOELE MANZONI

N86004562

LUCA LUCCI

N86005180

APRIL 19, 2025

ANNO ACCADEMICO: 2024/2025

# Contents

<b>1</b>	<b>Traccia Progetto: Hackathon</b>	<b>3</b>
<b>2</b>	<b>Analisi Dominio del Problema</b>	<b>3</b>
2.1	Ricerca delle Classi . . . . .	3
2.2	Ricerca Attributi e Associazioni . . . . .	4
2.2.1	Attributi . . . . .	4
2.2.2	Associazioni . . . . .	5
2.3	Ricerca delle Generalizzazioni . . . . .	5
2.3.1	Utente . . . . .	5
2.4	Ricerca delle Responsabilità . . . . .	5
2.4.1	Hackathon . . . . .	5
2.4.2	Utente . . . . .	5
2.4.3	Team . . . . .	6
2.4.4	Organizzatore . . . . .	6
2.4.5	Giudice . . . . .	6
2.5	Grafico UML del Dominio del Problema . . . . .	6

# 1 Traccia Progetto: Hackathon

Un Hackathon, ovvero una "maratona di hacking", è un evento durante il quale team di partecipanti si sfidano per progettare e implementare nuove soluzioni basate su una certa tecnologia o mirate a un certo ambito applicativo. Ogni Hackathon ha un titolo identificativo, si svolge in una certa sede e in un certo intervallo di tempo (solitamente 2 giorni) e ha un organizzatore specifico (registrato alla piattaforma). L'organizzatore seleziona un gruppo di giudici (selezionati tra gli utenti della piattaforma, invitandoli). Infine, l'organizzatore apre le registrazioni, che si chiuderanno 2 giorni prima dell'evento. Ogni evento avrà un numero massimo di iscritti e una dimensione massima del team. Durante il periodo di registrazione, gli utenti possono registrarsi per l'Hackathon di loro scelta (eventualmente registrandosi sulla piattaforma se non lo hanno già fatto). Una volta iscritti, gli utenti possono formare team. I team diventano definitivi quando si chiudono le iscrizioni. All'inizio dell'hackathon, i giudici pubblicano una descrizione del problema da affrontare. Durante l'hackathon, i team lavorano separatamente per risolvere il problema e devono caricare periodicamente gli aggiornamenti sui "progressi" sulla piattaforma come documento, che può essere esaminato e commentato dai giudici. Alla fine dell'hackathon, ogni giudice assegna un voto (da 0 a 10) a ciascun team e la piattaforma, dopo aver acquisito tutti i voti, pubblica le classifiche dei team.

## 2 Analisi Dominio del Problema

### 2.1 Ricerca delle Classi

Le classi trovate sono le seguenti:

- **Hackathon:** Classe dedicata agli eventi organizzati. Ciascuna sua istanza conterrà le informazioni dell'evento che rappresenterà e sarà di vitale importanza per il funzionamento della struttura.
- **Utente:** Classe per gli Utenti registrati alla piattaforma di organizzazione per gli eventi. Gli Utenti registrati avranno la possibilità di accedere alle varie funzionalità della piattaforma, a seconda dei poteri che gli verranno assegnati.
- **Organizzatore:** Classe rappresentante gli Utenti in grado di organizzare eventi.
- **Giudice:** Classe rappresentante gli Utenti invitati a fare da Giudici per i progetti proposti dai Team delle Hackathon.
- **Team:** Classe che aggrega i partecipanti di un Hackathon. Questa classe farà da portale per tutte le azioni relate alle attività dell'Hackathon alla quale il Team è stato iscritto.
- **Documento:** Classe per le documentazioni rilasciate dai Team durante lo sviluppo del progetto.

## 2.2 Ricerca Attributi e Associazioni

### 2.2.1 Attributi

Hackathon		
Attributo	Tipo Attributo	Descrizione
idNum	String	Numero identificativo per l'istanza di Hackathon
sede	String	Sede di appartenenza dell'evento
dataInizio	String	Data di inizio dell'evento
dataFine	String	Data della fine dell'evento
dataInizioRegistrazioni	String	Data in cui iniziano le registrazioni all'evento
dataFineRegistrazioni	String	Data in cui finiscono le registrazioni all'evento
titolo	String	Titolo dell'evento descrivente il tema sulla quale i team lavoreranno
maxMembriTeam	int	Numero massimo di membri per team
maxNumIscritti	int	Numero massimo di iscritti per evento
descrizioneProblema	String	Descrizione del problema da risolvere
giudiciEvento	ArrayList<Giudice>	Lista dei giudici che saranno invitati per suddetto ruolo all'Hackathon scelta

Utente		
Attributo	Tipo Attributo	Descrizione
login	final String	Nome utente per il login
password	String	Password dell'utente

Organizzatore		
Attributo	Tipo Attributo	Descrizione
hackathonOrganizzate	private ArrayList<Hackathon>	Lista delle Hackathon organizzate

Giudice		
Attributo	Tipo Attributo	Descrizione
userGiudice	Utente	Utente che ha ricevuto l'invito per diventare Giudice

Team		
Attributo	Tipo Attributo	Descrizione
membro	Utente[]	Lista di Utenti appartenenti al team; Poiché sapremo in anticipo quanto saranno grandi i team, verranno gestiti con l'ausilio di Array statici
nomeTeam	String	Nome del team
eventoPartecipazione	Hackathon	Hackathon per la quale il team sarà iscritto
documentazione	ArrayList<Documento>	Lista di tutti i documenti rilasciati dal team durante lo sviluppo del progetto
votoFinale	int	Somma di tutti i voti attribuiti al team alla fine dell'evento
index	int	Indice di controllo per la grandezza dell'array di membri del team

Documento		
Attributo	Tipo Attributo	Descrizione
text	String	Testo della documentazione
source	Team	Collegamento al team che ha scritto il documento

### 2.2.2 Associazioni

Nome Associazione	Classi Coinvolte	Molteplicità Associazione	Descrizione	Attributi (se classe associativa)
Partecipazione	Hackathon - Team	(1 - 2..*)	Un Hackathon deve avere un minimo di Team fino ad un massimo di N. Un Team può partecipare soltanto ad un Hackathon.	
Organizzazione	Hackathon - Organizzatore	(* - 1)	Un Hackathon può avere un singolo Organizzatore. Un Organizzatore può organizzare N Hackathon.	
Invito	Organizzatore - Giudice	(1..* - *)	Un Organizzatore può invitare più Giudici. Un Giudice può essere invitato da 1 a N Giudici.	
Supervisione	Hackathon - Giudice	(1..* - *)	Un Hackathon può essere supervisionato da 1 a N Giudici. Un Giudice può supervisionare da 1 a N Hackathon.	
Pubblicazione	Team - Documento	(1 - 1..*)	Un Documento può essere pubblicato solo da un Team. Un Team può pubblicare fino ad N Documenti.	
Team (Aggregazione di Utenti in Team)	Team - Utente	(0..1 - 1..*)	Un Team deve avere da 1 a N Utenti a far parte di sé stesso, dando per scontato che l'Utente che creerà un Team ne faccia automaticamente parte. Un Utente può pure non far parte di un Team, ma può far parte soltanto di un Team per Hackathon.	
Valutazione	Giudice - Documento	(1..* - 1..*)	Un Giudice deve valutare da 1 a N Documento. Un Documento deve essere valutato da 1 a N Giudici.	giudizio: stringa di testo contenente il parere espresso dal Giudice in relazione al Documento valutato.
Voto	Giudice - Team	(1..* - 1..*)	Un Giudice deve dare un voto ad 1 fino a N Team. Un Team deve ricevere un voto da 1 a N Giudici.	voto: somma di tutti i voti dei Giudici ricevuti dal Team.

## 2.3 Ricerca delle Generalizzazioni

### 2.3.1 Utente

La classe padre **Utente** rappresenta tutte le persone registrate alla piattaforma. Un Utente può unirsi ad un Hackathon come tale, creando anche un Team. Nel caso di ruoli che abbiano una qualche sorta di amministrazione, l'Utente può essere specializzato in due classi figlie:

- **Organizzatore:** la piattaforma può decidere di dare ad un Utente il ruolo di Organizzatore per gestire le Hackathon. Un Organizzatore può essere tale alla nascita stessa della piattaforma, ma rimarrà sempre registrato nella stessa come Utente, in quanto classe principale predisposta alla gestione dei login.
- **Giudice:** Un Utente può ricevere un invito da un Organizzatore a diventare Giudice di un Hackathon. Questa classe avrà il solo scopo di supervisionare l'andamento di un evento e di offrire i propri giudizi sui progetti dei Team.

## 2.4 Ricerca delle Responsabilità

### 2.4.1 Hackathon

- `aggiungiGiudice(giudice):` Aggiunge tra i giudici di quell'evento un nuovo giudice invitato da un organizzatore;
- `pubblicaClassifica():` Pubblica la classifica con i risultati finali.

### 2.4.2 Utente

- `registrazioneHackathon(Hackathon):` l'utente si registra ad un Hackathon;

- `getLogin(password)`: controlla l'accesso a tutte le azioni che richiedono la password dell'utente;
- `creaTeam(nomeTeam)`: crea un nuovo team.

### 2.4.3 Team

- `aggiungiMembro(Utente)`: aggiungi un nuovo membro al team;
- `uploadDocumento(testo)`: carica una nuova documentzione nella piattaforma.

### 2.4.4 Organizzatore

- `creazioneHackathon(Hackathon)`: crea un nuovo evento;
- `invitoGiudice(Utente)`: invita un utente a diventare giudice.

### 2.4.5 Giudice

- `pubblicazioneProblema()`: pubblica il problema da risolvere dell'Hackathon.

## 2.5 Grafico UML del Dominio del Problema

