



UNIVERSITÀ DEGLI STUDI DI NAPOLI
FEDERICO II

DOCUMENTAZIONE PER PROGETTAZIONE
BASE DI DATI
Progetto in Carico: Hackathon

CdL Triennale in Informatica
CORSO DI BASI DI DATI I
GIOELE MANZONI
N86004562
LUCA LUCCI
N86005180
JULY 11, 2025
ANNO ACCADEMICO: 2024/2025

Contents

1	Introduzione	3
1.1	Traccia del Progetto	3
2	Progettazione Concettuale	5
2.1	Analisi delle Entità e degli Attributi	5
2.1.1	Hackathon	5
2.1.2	Utente	5
2.1.3	Organizzatore	5
2.1.4	Giudice	5
2.1.5	Team	6
2.1.6	Documento	6
2.2	Analisi delle Relazioni	6
2.3	Ristrutturazione del Modello Concettuale	7
2.3.1	Analisi delle Ridondanze	7
2.3.2	Analisi delle Generalizzazioni/Gerarchie di Specializzazione	7
2.3.3	Analisi degli Attributi Multivalore	7
2.3.4	Analisi degli Attributi Strutturati	7
2.3.5	Partizionamento/Accorpamento di Entità e Relazioni	7
2.3.6	Analisi delle Chiavi	7
2.4	Class Diagram Ristrutturato	7
2.5	Dizionario delle Classi	8
2.6	Dizionario delle Associazioni	8
2.7	Dizionario dei Vincoli	8
3	Progettazione Logica	8
3.1	Schema Logico	8
4	Progettazione Fisica	8

1 Introduzione

Questa documentazione descriverà il processo di progettazione e sviluppo di un Database relazionale che gestirà il flusso di dati di un applicativo dedicato all'organizzazione di Hackathon. Questo è un progetto a cura degli studenti Gioele Manzoni e Luca Lucci del CdL di Informatica presso l'Università degli Studi di Napoli "Federico II".

1.1 Traccia del Progetto

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.

Caratteristiche dell'Hackathon

Ogni Hackathon ha le seguenti caratteristiche:

- Un **titolo identificativo**;
- Una **sede** in cui si svolge;
- Un **intervallo di tempo**, solitamente di due giorni;
- Un **organizzatore specifico**, registrato sulla piattaforma.

Giudici e Registrazioni

- L'organizzatore seleziona un gruppo di **giudici**, invitandoli tra gli utenti registrati sulla piattaforma.
- L'organizzatore apre le **registrazioni**, che si chiudono due giorni prima dell'inizio dell'evento.
- Ogni evento prevede un **numero massimo di iscritti** e una **dimensione massima del team**.
- Durante il periodo di registrazione, gli utenti possono registrarsi all'Hackathon di loro scelta, previa registrazione sulla piattaforma se non ancora effettuata.

Formazione dei Team

- Una volta iscritti, gli utenti possono **formare team**.
- I team diventano **definitivi alla chiusura delle iscrizioni**.

Svolgimento dell'Hackathon

- All'inizio dell'evento, i giudici **pubblicano una descrizione del problema** da affrontare.
- Durante l'Hackathon, i team lavorano separatamente per risolvere il problema.
- I team devono **caricare periodicamente aggiornamenti sui progressi** tramite documenti sulla piattaforma.
- I documenti possono essere **esaminati e commentati dai giudici**.

Valutazione e Classifica

- Alla fine dell'Hackathon, ogni giudice assegna un **voto da 0 a 10** a ciascun team.
- La piattaforma, dopo aver acquisito tutti i voti, **pubblica le classifiche dei team**.

2 Progettazione Concettuale

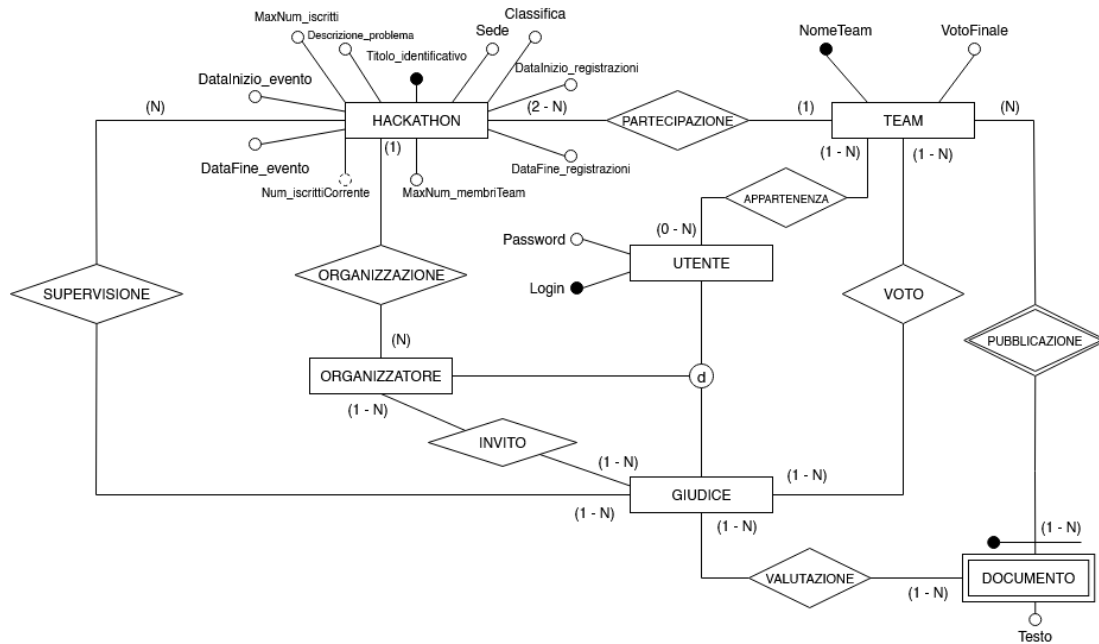


Figure 1: Grafico UML Concettuale

2.1 Analisi delle Entità e degli Attributi

Seguendo la traccia, nella fase di progettazione concettuale sono state trovate le suddette entità:

2.1.1 Hackathon

Entità dedicata a tutte le maratone di Hacking organizzate.

- **Hackathon** (Titolo_identificativo, Descrizione_problema, Sede, Classifica, DataInizio_registrazioni, DataFine_registrazioni, DataInizio_Evento, DataFine_Evento, Num_iscrittiCorrente, MaxNum_membriTeam, MaxNum_iscritti)

2.1.2 Utente

Generalizzazione dedicata a tutti i tipi di utenti che è possibile avere all'interno della piattaforma. La generalizzazione è considerata come **DISGIUNTA PARZIALE**, poiché è possibile avere utenti della piattaforma che non sono né organizzatori né giudici.

- **Utente** (Login, Password)

2.1.3 Organizzatore

Specializzazione dell'entità **Utente**, rappresentante gli organizzatori di maratone di Hacking. Non possiede alcun attributo specifico a sé stesso, la sua specializzazione definisce soltanto gli utenti con le giuste credenziali per poter gestire la piattaforma.

2.1.4 Giudice

Specializzazione dell'entità **Utente**, rappresentante i giudici che daranno le loro valutazioni ai documenti e supervisioneranno le Hackathon.

2.1.5 Team

Entità che definisce una squadra organizzata da un Utente e composta da N Utenti.

- **Conferenza** (*NomeTeam*, VotoFinale)

2.1.6 Documento

Entità debole che definisce un documento scritto da un team.

- **Documento** (*NomeTeam*, Testo)

2.2 Analisi delle Relazioni

Qui verranno descritte tutte le relazioni e le specializzazioni presenti all'interno della struttura concettuale non ancora ristrutturata.

- *Organizzazione* (**Hackathon - Organizzatore**: 1 - N):
Un Hackathon può essere organizzata da un solo organizzatore. Un organizzatore può organizzare più Hackathon.
- *Partecipazione* (**Team - Hackathon**: 1 - 2..N):
Un Team può partecipare ad una sola Hackathon. Un Hackathon, per essere valida, deve avere un minimo di 2 Team fino ad un massimo di N.
- *Supervisione* (**Giudice - Hackathon**: 1..N - N):
Un Giudice può supervisionare N Hackathon. Un Hackathon deve essere monitorata da almeno un giudice.
- *Appartenenza* (**Utente - Team**: 0..N - 1..N):
Un Utente può partecipare ad uno o più team, o può non parteciparci affatto. Un Team deve essere composto da un minimo di un Utente fino ad un massimo di N (il limite di utenti appartenenti ad un Team è deciso dall'Hackathon alla quale si partecipa).
- *Invito* (**Giudice - Hackathon**: 1..N - 1..N):
Un giudice deve invitare un minimo di un utente fino ad un massimo di N utenti per essere giudici. Un giudice, per ricevere un invito, deve riceverlo da almeno un organizzatore.
- *Voto* (**Giudice - Team**: 1..N - 1..N):
Un Giudice può esprimere una votazione ad un minimo di un Team. Un team può ricevere voti da almeno un giudice.
- *Valutazione* (**Giudice - Documento**: 1..N - 1..N):
Un Giudice deve esprimere una valutazione per almeno un documento. Un documento deve ottenere una valutazione da almeno un giudice.
- *Pubblicazione* (**Documento - Team**: 1..N - N):
Relazione identificante per l'entità debole Documento, in quanto generato direttamente da un Team e non può esistere senza un associazione ad un Team.

Dopo aver analizzato i requisiti, le entità e le relazioni ed aver prodotto uno schema concettuale passeremo alla sua Ristrutturazione, seguendo i passaggi necessari elencati nelle prossime sottosezioni, ordinate in:

- 2.3.1** **Analisi delle Ridondanze**
- 2.3.2** **Analisi delle Generalizzazioni/Gerarchie di Specializzazione**
- 2.3.3** **Analisi degli Attributi Multivalore**
- 2.3.4** **Analisi degli Attributi Strutturati**
- 2.3.5** **Partizionamento/Accorpamento di Entità e Relazioni**
- 2.3.6** **Analisi delle Chiavi**

```

classDiagram
    class HACKATHON {
        +Titolo_identificativo: VARCHAR[30]
        +Sede: VARCHAR[30]
        +Classifica: VARCHAR[1000]
        +DataInizio_registrazione: DATE
        +DataFine_registrazione: DATE
        +Descrizione_problema: VARCHAR[1000]
        +DataInizio_evento: DATE
        +DataFine_evento: DATE
        +NumIscritti_corrente: INTEGER
        +MaxNum_iscritti: INTEGER
    }
    class ORGANIZZATORE {
        +Username_organizzatore: VARCHAR[30]
        +Password: VARCHAR[30]
    }
    class MEMBERSHIP {
        +ID_adesione: INTEGER
        +Data_adesione: DATE
    }
    class UTENTE {
        +Username: VARCHAR[30]
        +Password: VARCHAR[30]
    }
    class TEAM {
        +Nome_team: VARCHAR[30]
        +Punteggio_finale: INTEGER
    }
    class VOTO {
        +Punteggio: INTEGER
    }
    class DOCUMENTO {
        +ID_documento: INTEGER
        +Titolo: VARCHAR[30]
        +Contenuto: VARCHAR[500]
        +Data_stesura: DATE
    }
    class VALUTAZIONE {
        +Valutazione_giudice: VARCHAR[500]
    }
    class GIUDICE {
        +ID_giudice: INTEGER
    }

    HACKATHON "0..*" -- "1" TEAM : + partecipazione
    HACKATHON "1..*" -- "1" ORGANIZZATORE : + organizzazione
    HACKATHON "1" -- "0..*" ORGANIZZATORE : + invito
    HACKATHON "1" -- "1" MEMBERSHIP : + aggregazione
    HACKATHON "1" -- "0..*" UTENTE : + promozione
    HACKATHON "1" -- "0..*" GIUDICE : + supervisione
    TEAM "1" -- "1..*" MEMBERSHIP : + assegnazione voto
    TEAM "1" -- "1" VOTO : + immissione voto
    TEAM "1" -- "1" DOCUMENTO : + stesura documento
    TEAM "1" -- "0..*" GIUDICE : + immissione valutazione
    MEMBERSHIP "1" -- "1" UTENTE : + adesione
    VOTO "1" -- "1" DOCUMENTO : + registrazione valutazione
    VOTO "1" -- "1" VALUTAZIONE : + immissione valutazione
    DOCUMENTO "1" -- "1" VALUTAZIONE : + valutazione
    GIUDICE "1" -- "1" VALUTAZIONE : + valutazione
  
```

2.5 Dizionario delle Classi

2.6 Dizionario delle Associazioni

2.7 Dizionario dei Vincoli

3 Progettazione Logica

3.1 Schema Logico

- Hackathon(Titolo_identificativo, Sede, Classifica, DataInizio_registrazione, DataFine_registrazione, Descrizione_problema, DataInizio_evento, DataFine_evento, NumIscritti_corrente, MaxNum_iscritti)
- Utente(Username, Password)
- Organizzatore(Username_organizzatore, Password)
- Giudice(ID_giudice, Username_utente, Titolo_hackathon)
Username_utente \Rightarrow Utente.Username
Titolo_hackathon \Rightarrow Hackathon.Titolo_identificativo
- Team(Nome_team, Titolo_hackathon, Punteggio_finale)
Titolo_hackathon \Rightarrow Hackathon.Titolo_identificativo
- Documento(ID_documento, Nome_team, Titolo_hackathon, Titolo, Contenuto, Data_stesura)
Nome_team \Rightarrow Team.Nome_team
Titolo_hackathon \Rightarrow Hackathon.Titolo_identificativo
- Membership(ID_adesione, Username_utente, Nome_team, Titolo_hackathon, Data_adesione)
Username_utente \Rightarrow Utente.Username
Nome_team \Rightarrow Team.Nome_team
Titolo_hackathon \Rightarrow Hackathon.Titolo_identificativo
- Voto(ID_giudice, Username_giudice, Titolo_hackathon, Nome_team, Punteggio)
ID_giudice \Rightarrow Giudice.ID_giudice
Username_giudice \Rightarrow Utente.Username
Titolo_hackathon \Rightarrow Hackathon.Titolo_identificativo
Nome_team \Rightarrow Team.Nome_team
- Valutazione(ID_giudice, Username_giudice, Titolo_hackathon, Nome_team, ID_documento, Valutazione_giudice)
ID_giudice \Rightarrow Giudice.ID_giudice
Username_giudice \Rightarrow Utente.Username
Titolo_hackathon \Rightarrow Hackathon.Titolo_identificativo
Nome_team \Rightarrow Team.Nome_team
ID_documento \Rightarrow Documento.ID_documento

4 Progettazione Fisica