

Per gestire il progetto ri9chiesto, abbiamo deciso di affidarci all’uso di entità e attributi. Di seguito ci sarà un elenco delle entità utilizzate e una spiegazione del loro impiego nel progetto con i rispettivi attributi.

* **Studente**: entità per gestire gli studenti;
  + ID: impostato come chiave primaria;
  + Nome: nome dello studente
  + Cognome: cognome dello studente
  + Username: username dello studente
  + Password: password studente
  + Classe: classe frequentata dallo studente
* **Professore**: entità per gestire i professori;
  + ID: impostato come chiave primaria;
  + Nome: nome del professore;
  + Cognome: cognome del professore;
  + Username: username del professore;
  + Password: password professore;
* **Evento**: entità per gestire gli eventi del calendario (gite, verifiche, interrogazioni…)
  + ID: chiave primaria per identificare ogni evento;
  + Data: data prevista;
  + Ora: ora prevista;
  + Note: eventuali note riguardo l’evento;
* **Classe**: entità classe;
  + ID: chiave primaria per identificare ogni classe;
  + Classe: è il numero dell’anno della classe;
  + Sezione: a, b, c…;
  + Indirizzo: indirizzo della classe (es: informatica, meccanica…).
* **Materie**: entità per gestire ogni materia;
  + ID: chiave primaria per identificare ogni materia;
  + Nome: matematica, italiano…

Analizziamo ora le associazioni tra le entità, evidenziando la molteplicità che esiste.

1. **Tengono**: Professore – Classe (M: M), l’associazione è di tipo molti a molti in quanto più professori tengono più classi. Viceversa, più classe sono tenute da più professori.
2. **Insegnano**: Professore – Materie (1: M) l’associazione di tipo 1 a molti prevede infatti che un professore insegni una materia, e che una materia sia insegnata da più professori.
3. **Hannoprof**: Professore – Evento (M: M) l’associazione è di tipo molti a molti perché un professore programma molti eventi, e molti eventi sono stati programmati da molti professori.
4. **Hannostud**: Studente – Evento (M: M) quest’associazione molti a molti è così composta: uno studente ha in programma uno o più eventi e un evento coinvolge uno o più studenti.

Infine creiamo lo schema logico di ogni associazione.

1. Mapping associazione ISA:

- **Studente**: (ID; Nome; Cognome; Username, Password, Classe);

- **Professore**: (ID; Nome; Cognome: Username, Password).

2. Mapping associazione Hannoprof tra Professore ed evento:

- **Professore**: (ID; Nome; Cognome: Username, Password);

- **Evento**: (ID; Data; Ora; Note).

3. Mapping associazione Hannostud tra Studente ed evento:

- **Professore**: (ID; Nome; Cognome: Username, Password);

- **Evento**: (ID; Data; Ora; Note).

4. Mapping associazione Insegnano tra Professore e materie:

- **Professore**: (ID; Nome; Cognome: Username, Password);

- **Materie**: (ID; Nome).

5. Mapping associazione Tengono tra Professore e classe:

- **Professore**: (ID; Nome; Cognome: Username, Password);

- **Classe**: (ID; Classe, Sezione, Indirizzo).