**Titolo progetto**: gestione risorse didattiche

L’applicazione consente agli studenti di cercare tra le risorse disponibili quelle più adatte alle materie e ai corsi che stanno seguendo. Gli studenti e i tutor hanno la possibilità di pubblicare delle risorse digitali per supportare lo studio di altri studenti, una volta finito il percorso di studi le risorse sono ancora pubbliche e disponibili.

Le risorse possono essere:

* digitali (link verso documenti memorizzati in rete);
* fisiche (biblioteca: libri, cd, test vecchi);
* tutor (professori disponibili).

Uno studente durante l’anno segue diversi corsi (obbligatori e facoltativi).

I corsi obbligatori sono abbinati all’indirizzo scelto e alla classe frequentata.

I dati per lo studente gestiti sono i seguenti:

* nome;
* cognome;
* data di nascita;
* matricola;
* anno di corso;
* indirizzo scelto;
* lingua scelta;
* mail istituzionale;
* password.

Il corso (= materia) dello studente contiene i seguenti dati:

* nome;
* anno di corso;
* argomenti trattati, tali argomenti dipendono dalla classe frequentata.

Alcuni corsi sono comuni indipendentemente dall’indirizzo, altri sono specifici in base all’indirizzo scelto.

I dati del tutor sono:

* nome;
* cognome;
* materie di tutoraggio;
* email interna;
* password.

Sono presenti uno o più amministratori di sistema (Admin). Deve essere uno dei tutor.

**Casi d’uso**:

* Attori del sistema: studente, tutor e amministratore.
* Admin
  + Identificazione da parte del sistema (in base a email istituzionale e password privata).
  + Validazione nuove risorse digitali degli studenti (upload).
  + Attiva/disabilita/inserisce tutor.
  + Disabilita gli studenti che non fanno più parte dell’istituto.
  + Inserisce/modifica corsi obbligatori
  + Inserisce/modifica corsi facoltativi
* Tutor
  + Identificazione da parte del sistema (in base a email istituzionale e password privata).
  + Accesso al sistema e visualizzazione degli argomenti di cui fa tutoraggio.
  + Proposizione di una nuova risorsa digitale riguardo agli argomenti di cui ha dato la disponibilità.
  + Inserimento/rimozione degli argomenti di tutoraggio.
  + Aggiornamento delle risorse.
* Studente
  + Identificazione da parte del sistema (in base a email istituzionale e password privata) solo per gli studenti ancora iscritti nell’istituto.
  + Accesso al sistema e visualizzazione delle materie del proprio corso e anno.
  + Possibilità di selezionare un corso e vedere l’elenco delle risorse disponibili suddivise per tipo e argomento.
  + Proposizione di una nuova risorsa digitale (da validare).
  + Ricerca di risorse inerenti a uno specifico argomento.
  + Rimozione di una nuova risorsa non ancora approvata e da lui sottoposta.

**Schema logico**

[schema logico](https://drive.google.com/file/d/1KbOwXlS0HGhEZsn95-we9VDEusvqVZ_b/view?usp=sharing)

**Schema fisico**

[schema fisico](https://drive.google.com/file/d/1gZhHQEHyNqJnPnjVJCFCszRAdRlOBB7_/view?usp=sharing)

**Dominio**

Il Database deve contenere le seguenti entità:

* Elenco risorse disponibili
* Studenti
* Tutor
* Corsi
* Risorse digitali
* Risorse fisiche
* Studenti-Corsi
* Studenti-Corsi-facoltativi
* Argomenti
* Tutor-argomenti

L’entità **Elenco risorse disponibili** deve contenere i seguenti dati:

* ID\_risorsa int (5)
* Tipo varchar (20)
* matricola int (5) FK
* id\_tutor int (5) FK

L’entità **Studenti** deve contenere i seguenti dati:

* matricola int (5)
* nome varchar (30)
* cognome varchar (30)
* data\_nascita date
* anno\_corso date
* lingua\_scelta varchar (20)
* mail\_istituzionale varchar (30)
* password varchar (8)

L’entità **tutor** deve contenere i seguenti dati:

* ID\_tutor int (5)
* nome varchar (30)
* cognome varchar (30)
* email\_interna varchar (30)
* password varchar (8)
* admin bool
* attivo bool
* id\_risorse int (5) FK

L’entità **Corsi** deve contenere i seguenti dati

* ID\_corso int (5)
* nome varchar (30)
* anno\_corso date

L’entità **Risorse digitali** deve contenere i seguenti dati

* ID\_digitali int (5)
* nome varchar (30)
* link varchar (100)
* approvata bool
* id\_risorse int (5) FK

L’entità **Risorse fisiche** deve contenere i seguenti dati.

* ID\_fisiche int (5)
* Nome varchar (50)
* Ubicazione varchar (50)
* Approvata bool
* Tipo varchar (20)
* id\_risorse int (5) FK

L’entità **Studenti-corsi-facoltativi** deve contenere i seguenti dati:

* id\_corso int (5) PK, FK
* matricola int (5) PK, FK

L’entità **Studenti-corsi** deve contenere i seguenti dati:

* id\_corso int (5) PK, FK
* matricola int (5) PK, FK
* classe varchar (5)
* indirizzo varchar (15)

L’entità

**Query di creazione del database**

CREATE DATABASE Gestione\_risorse\_didattiche

CREATE TABLE **Studenti** (

Matricola int (5) PRIMARY KEY,

nome varchar (30) NOT NULL,

cognome varchar (30) NOT NULL,

data\_nascita date,

anno\_corso int (4),

indirizzo\_scelto varchar (20),

lingua\_scelta varchar (20),

mail\_istituzionale varchar (30) NOT NULL,

password varchar (8) NOT NULL

);

CREATE TABLE **Tutor** (

ID\_tutor int (5) PRIMARY KEY,

nome varchar (30) NOT NULL,

cognome varchar (30) NOT NULL,

materie\_tutoraggio varchar (20),

email\_interna varchar (30) NOT NULL,

password varchar (8) NOT

)

CREATE TABLE **Corsi** (

ID\_corso int (5) PRIMARY KEY,

nome varchar (30) NOT NULL,

anno\_corso date NOT NULL,

argomenti\_trattati varchar (100)

);

CREATE TABLE **Risorse** (

ID\_risorsa int (5) PRIMARY KEY,

digitali bool,

fisiche bool,

tutor bool,

id\_tutor int (5),

id\_studenti int (5),

FOREIGN KEY (id\_tutor) REFERENCES tutor (ID\_tutor),

FOREIGN KEY (id\_studenti) REFERENCES studenti (Matricola)

);

CREATE TABLE **Studenti\_corsi** (

matricola int (5),

id\_corso int (5),

CONSTRAINT PK\_studenti\_corsi PRIMARY KEY (matricola, id\_corso),

FOREIGN KEY matricola REFERENCES Studenti (Matricola),

FOREIGN KEY id\_corso REFERENCES Corsi (ID\_corso)

);

CREATE TABLE **Corsi\_tutor** (

id\_tutor int (5),

id\_corso int (5),

CONSTRAINT PK\_corsi\_tutor PRIMARY KEY (id\_tutor, id\_corso),

FOREIGN KEY (id\_tutor) REFERENCES tutor (ID\_tutor),

FOREIGN KEY (id\_corso) REFERENCES corsi (ID\_corso)

);