
CALENDAR

20/03/2023

STRUTTURA

Il database sarà composto dalle seguenti tabelle: tabella STUDENTI, tabella MATERIE, tabella EVENTI.

La tabella STUDENTI sarà composta da **login id** che sarà la Primary Key Auto_increment, **username** che è una `varchar(45)`, **password** che è una `varchar(45)`, **email** è una `varchar(45)`.

[[esempio studenti](#)]

La tabella MATERIE sarà composta da **materie id** che è la Primary Key Auto_increment, **subject** che è una `varchar(45)`, **teacher** che è una `varchar(45)`.

[[esempio materie](#)]

La tabella EVENTI sarà composta da **eventi id** che è la Primary Key Auto_increment, **date** che è di tipo `data(YYYY-MM-DD)`, **description** è una `varchar(45)`.

[[esempio eventi](#)]

La tabella STUDENTI avrà una relazione (1 - MOLTI) con la tabella EVENTI perché uno studente può avere più eventi.

La tabella STUDENTI avrà una relazione (1 - MOLTI) con la tabella MATERIE perché uno studente può avere più materie.

La relazione tra la tabella EVENTI e la tabella MATERIE è (1 - 1) perché ogni evento può avere solo una materia.

OPERAZIONI

L'utente dovrà registrarsi e dare un username, una password e l'email. [vedere [img](#)]

L'utente avrà la possibilità di inviare un **POST** (`quindi Create`) per aggiungere un evento. L'evento dovrà essere composto dall' username (dell'utente), subject, teacher, data e una descrizione.

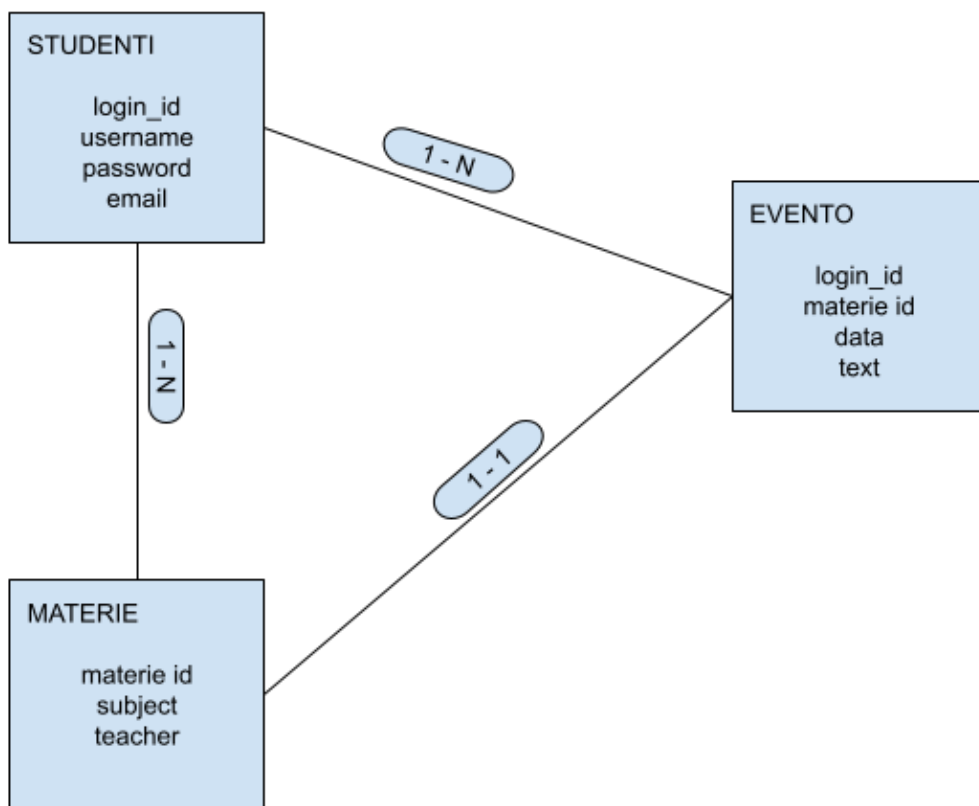
L'utente avrà la possibilità di modificare l'evento con **PUT** (`quindi Upload`), per far sì che l'upload vada a buon fine, bisognerà cambiare la data e/o la descrizione.

L'utente avrà la possibilità di eliminare l'evento già creato con **DELETE**. [vedere [img](#)]

Le operazioni (**CRUD**) da aggiungere in questo progetto sono **Create**, per creare e/o aggiungere un nuovo evento nel calendario. **Read**, per leggere i dati dell'evento. **Update** per aggiornare i dati sull'evento. **Delete** per cancellare o eliminare uno o più dati. Una volta fatto provare se funziona con l'utilizzo di POSTMAN.

Dopo aver fatto ciò e fatto i test, bisognerà criptare le password del utente (**STUDENTE**) per far ciò bisognerà aggiungere SHA2(?, 512) nel post per il sign in.

GRAFICA



ESEMPIO MATERIE

```
mysql> SELECT * FROM materie;
+-----+-----+-----+
| materie_id | subject      | teacher |
+-----+-----+-----+
|          1 | Matematica   | Fardin  |
|          2 | Italiano     | Francesca |
|          3 | Manutenzione | Marco   |
|          4 | Storia       | Christian |
|          5 | Scienze      | Filippo |
+-----+-----+-----+
5 rows in set (0.01 sec)
```

ESEMPIO STUDENTI

```
mysql> SELECT * FROM studenti;
+-----+-----+-----+-----+
| login_id | username | password | email |
+-----+-----+-----+-----+
|          1 | Stephen  | 123      | stephen@gmail.com |
|          2 | Joseph   | 321      | joseph@gmail.com  |
|          3 | Eva      | 654      | eva@gmail.com      |
|          4 | Diana    | 456      | Diana@gmail.com    |
|          5 | Atena    | 789      | Atena@gmail.com    |
|          7 | Leo      | 987      | Leo@gmail.com      |
|          8 | Ettore   | 0         | Ettore@gmail.com   |
|          9 | Blues    | 111      | Blues@gmail.com    |
|         10 | Hope     | 222      | Hope@gmail.com     |
+-----+-----+-----+-----+
9 rows in set (0.00 sec)
```

ESEMPIO EVENTI

```
mysql> SELECT * FROM eventi;
+-----+-----+-----+
| eventi_id | date       | description |
+-----+-----+-----+
|          1 | 2023-03-20 | riunione   |
|          2 | 2023-03-25 | partita    |
+-----+-----+-----+
2 rows in set (0.01 sec)
```

FRONT - END

Come dovrebbe essere [sign-in.html](#).

A wireframe diagram of a sign-in/register form. It features a light blue circular logo at the top with the text "LOGO". Below the logo is a gray rectangular box containing the word "REGISTER" in bold. Underneath "REGISTER" are three stacked input fields labeled "username", "password", and "email". At the bottom of the gray box is a button labeled "SIGN IN". The entire form is enclosed in a black border.

Come dovrebbe essere [index.html](#).

A wireframe diagram of an index page. At the top left is a blue button with a "+" sign. Below it are two identical gray rectangular boxes, each representing a list item. Each box has a header section labeled "DAY" and a footer section containing two buttons: "DELETE" and "CHANGE". The entire page layout is enclosed in a black border.

Come dovrebbe essere **index.html**.

Quando l'utente vuole aggiungere un evento, deve compilare questo.

<input type="text" value=" TITOLO"/>				
<input type="text" value=" testo"/>				
<table><tr><td>MATERIE</td></tr><tr><td>italia</td></tr><tr><td>storia</td></tr></table>	MATERIE	italia	storia	<input type="text" value=" data"/>
MATERIE				
italia				
storia				
<input type="button" value="INVIA"/>				