

Documentazione Progetto SWA

Nome Progetto: Gestione Aule

Autori:

Nome	Cognome	Matricola	Ruolo
Stefano	Bavota	258821	Sviluppo front-end e back-end
Lorenzo	Bosica	267240	Database, supporto front-end, back -end e documentazione

Dipendenze software:

Tecnologie/Librerie/Framework Client:

-HTML

HTML (HyperText Markup Language) è il linguaggio di markup standard utilizzato per strutturare il contenuto delle pagine web. Nella nostra applicazione, utilizziamo HTML per definire la struttura dei componenti e dei template dell'interfaccia utente.

-CSS

CSS (Cascading Style Sheets) è il linguaggio utilizzato per definire lo stile e la presentazione delle pagine web. Nella nostra applicazione, utilizziamo CSS per gestire l'aspetto grafico dell'interfaccia utente.

-JavaScript

JavaScript è un linguaggio di programmazione client-side ampiamente utilizzato per aggiungere interattività e funzionalità dinamiche alle pagine web.

-Angular

Angular è un framework di sviluppo front-end sviluppato da Google che facilita la creazione di applicazioni web complesse e dinamiche. Includiamo Angular nel nostro progetto per sfruttare le sue funzionalità avanzate, come il data binding bidirezionale, la gestione delle route, l'iniezione delle dipendenze. Angular ci consente di organizzare il nostro codice in modo modulare e mantenibile, facilitando la realizzazione di applicazioni scalabili.

-Full-calendar

Full-calendar è una libreria JavaScript per la gestione delle date che estende le funzionalità del calendario. Utilizziamo Full-calendar per visualizzare, selezionare e gestire gli eventi e le date all'interno della nostra applicazione. La libreria offre una vasta gamma di funzioni, come la visualizzazione di diversi tipi di calendari, il supporto per gli eventi ricorrenti e altro ancora.

Tecnologie/Librerie/Framework Server:

-Java

Java è un linguaggio di programmazione versatile, affidabile e portabile che si basa su un modello orientato agli oggetti. È noto per la sua sicurezza, la gestione automatica della memoria e la capacità di eseguire il codice su diverse piattaforme senza modifiche significative. Java è ampiamente utilizzato per lo sviluppo di applicazioni desktop, server, mobile e web, ed è supportato da una vasta libreria standard (API) che semplifica lo sviluppo. La sua sintassi è chiara e leggibile, il che lo rende adatto sia per i principianti che per gli sviluppatori esperti. Inoltre, Java è noto per la sua affidabilità e scalabilità, il che lo rende una scelta popolare per applicazioni critiche e di grandi dimensioni.

-SpringBoot

Spring Boot è una piattaforma di sviluppo che offre numerosi vantaggi:

Semplicità di configurazione: Spring Boot elimina la necessità di configurazioni complesse, consentendo una rapida messa a punto del progetto.

Incorporazione di server: Include server incorporati come Tomcat, Jetty o Undertow, semplificando la distribuzione dell'applicazione.

Gestione delle dipendenze: Spring Boot gestisce automaticamente le dipendenze delle librerie, riducendo al minimo il lavoro manuale.

Struttura organizzata: Favorisce la struttura modulare del progetto, con controller, DTO, entity, repository, service e services ben definiti.

Struttura del Progetto:

Nel nostro progetto, abbiamo organizzato il Back-End come segue:

-Controller: Abbiamo utilizzato i controller per gestire le richieste HTTP in arrivo dai client. Ciascun controller è specializzato in un'area funzionale specifica dell'applicazione e indirizza le richieste ai servizi appropriati.

-DTO (Data Transfer Object): Abbiamo definito DTO per trasferire dati tra il Back-End e il Front-End in modo efficiente, includendo solo i campi di dati necessari.

-Model (Entity): Le entity rappresentano gli oggetti di dominio dell'applicazione e sono mappate alle tabelle del database, garantendo una rappresentazione accurata dei dati.

-Repository: Abbiamo creato repository per gestire l'accesso e la persistenza dei dati nel database, sfruttando le funzionalità di Spring Data JPA.

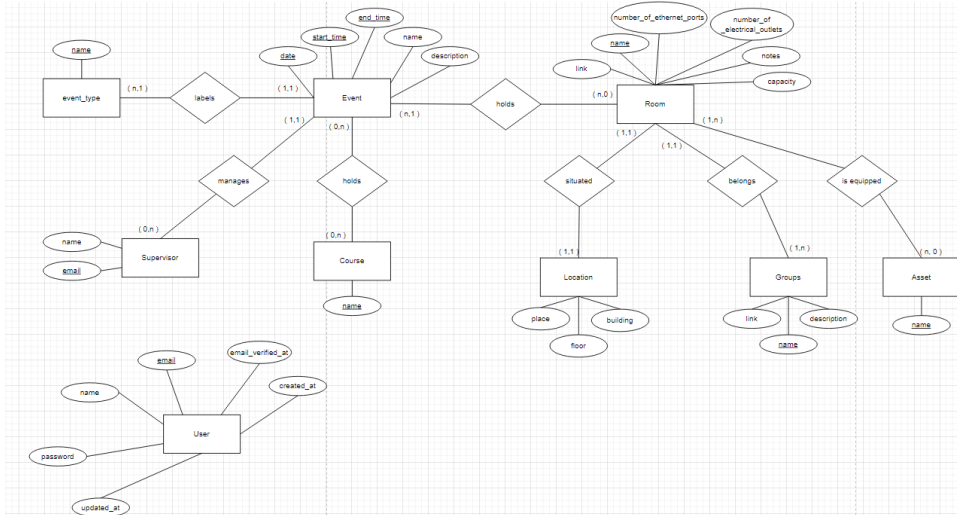
-Service: I service contengono la logica di business dell'applicazione, gestendo l'elaborazione dei dati e chiamando i repository per l'accesso ai dati.

-Services: Abbiamo utilizzato i services per offrire funzionalità ausiliarie, come la gestione delle transazioni e altre operazioni comuni.

-Mysql

MySQL è un sistema di gestione di database relazionali (RDBMS) ampiamente utilizzato per la memorizzazione e la gestione dei dati. Nel nostro progetto, utilizziamo MySQL per creare e gestire il database dell'applicazione. Utilizziamo l'ORM eloquent di Laravel per interagire con il database in modo semplice e sicuro.

-Modello E/R DB



Descrizione Database

Tabelle:

Event(date, start_time, end_time, name, description)

Room(name, number_of_ethernet_ports,
number_of_electrical_outlets, notes, capacity, link)

Supervisor(email, name)

Course(name)

Location (place, floor, building)

Group(name, link, description)

Asset(name)

Event_type(name)

User(email, email_verified_at, created_at, name, password,
updated_at)

Dopo aver strutturato il modello del database abbiamo scelto di partire prima con una implementazione MySQL, per testare l'interazione delle relazioni fra le varie tabelle e i funzionamenti delle query, una volta verificato il tutto abbiamo iniziato lo sviluppo vero e proprio in Laravel (in allegato insieme alla documentazione il file MySQL, l'implementazione in Laravel differisce leggermente dalla prima implementazione in MySQL).

Funzionalità Realizzate:

1. Login/logout con username e password (per gli amministratori).
2. Esportazione e importazione CSV configurazione aule non funziona solo dal front-end.
3. Inserimento di una nuova aula.
4. Assegnazione di un'aula a un gruppo.
5. Lettura delle informazioni di base relative a un'aula.
6. Lista delle attrezzature presenti in un'aula.
7. Inserimento di un nuovo evento.
8. Modifica di un evento.
9. Lettura delle informazioni su un evento.
10. Lista degli eventi associati a una specifica aula in una determinata settimana.
11. Lista degli eventi attuali e quelli delle prossime tre ore.
12. Creazione, visualizzazione, modifica eventi
13. Creazione, visualizzazione, modifica gruppi
14. Creazione, visualizzazione, modifica classi
15. La cancellazione implementate lato back-end ma non lato front-end
16. L'export degli eventi la query è stata implementata e funziona ma non funziona quando richiamata dal front end

Funzionalità Non Realizzate:

1. La formattazione degli orari con scarti di 15 minuti
2. Importazione CSV configurazione aule
3. La pianificazione di eventi giornalieri, settimanali, ecc...

Specifica delle API:

Per una visione migliore andare su <https://editor.swagger.io/> ed incollare il codice del file.yaml lasciato in allegato al progetto.

Esempi JSON:

Esempio JSON Entità gruppo

```
[
  {
    "id": 0,
    "name": "string",
    "description": "string",
    "link": "string"
  }
]
```

Esempio JSON Entità Classe

```
[
  {
    "id": 0,
    "name": "string",
    "capacity": 0,
    "description": "string",
    "link": "string",
    "electricalOutlets": 0,
    "ethernetPorts": 0,
    "supervisor_id": 0,
    "group_id": 0,
    "location_id": 0
  }
]
```

Esempio Entità Evento

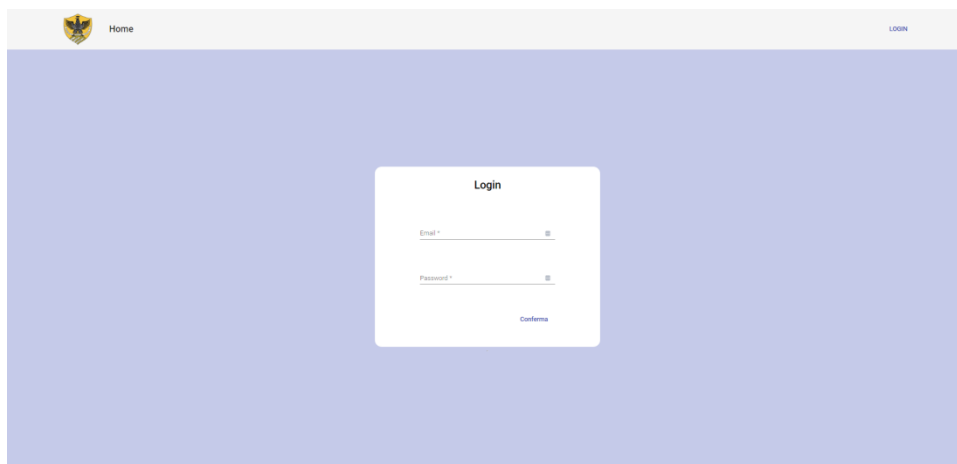
```
[
  {
    "id": 0,
    "name": "string",
    "description": "string",
    "date": "string",
    "start_date": "string",
    "end_date": "string",
    "typology_id": 0,
    "supervisor_id": 0,
    "room_id": 0,
    "room": {
      "id": 0,
      "name": "string",
      "capacity": 0,
      "description": "string",
      "link": "string",
      "electricalOutlets": 0,
      "ethernetPorts": 0,
      "supervisor_id": 0,
      "group_id": 0,
      "location_id": 0
    },
  },
]
```

Lista Browser Compatibili:

- Google Chrome
- Microsoft Edge
- Opera
- Mozilla Firefox
- Brave
- Vivaldi

Struttura del sito:

Pagina login

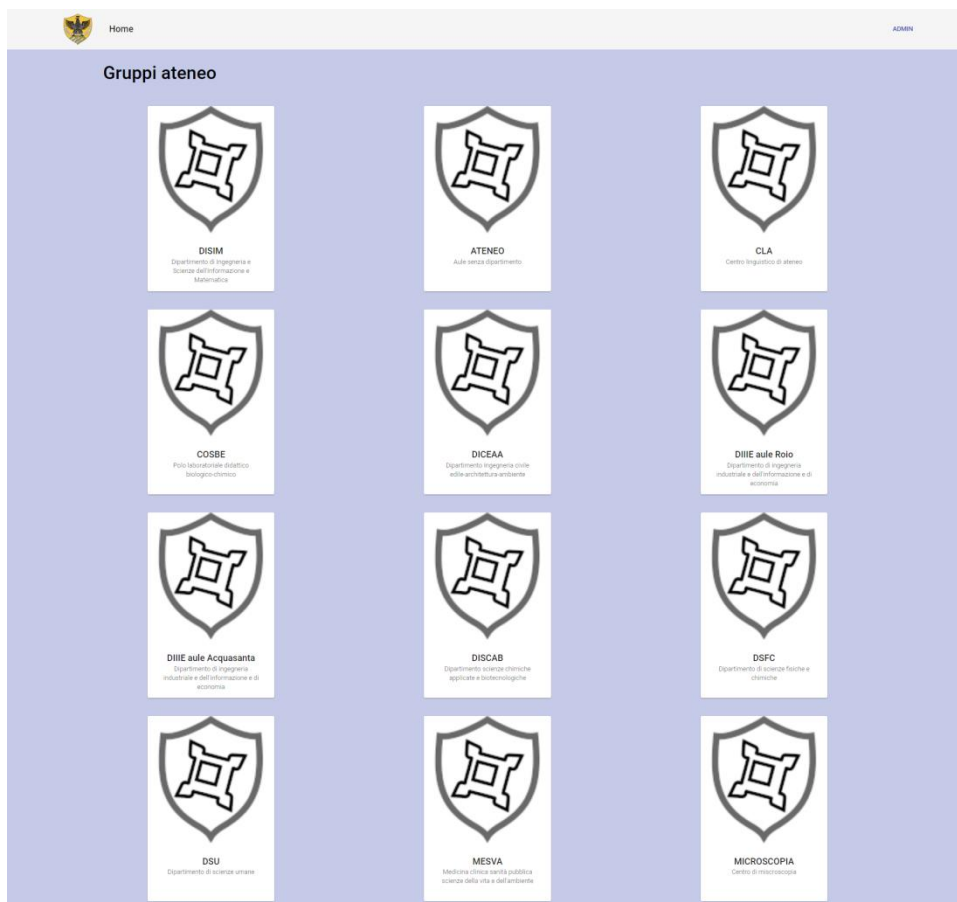


Nella pagina del login gli amministratori possono effettuare l'accesso tramite l'inserimento di e-mail e password.

Questa pagina sarà accessibile da tutto il sito se non si è autenticati tramite il tasto in alto a destra "LOGIN" se si è già autenticati il tasto verrà sostituito con il tasto "ADMIN".

In alto a sinistra invece c'è il tasto home che porta sulla pagina principale del sito.

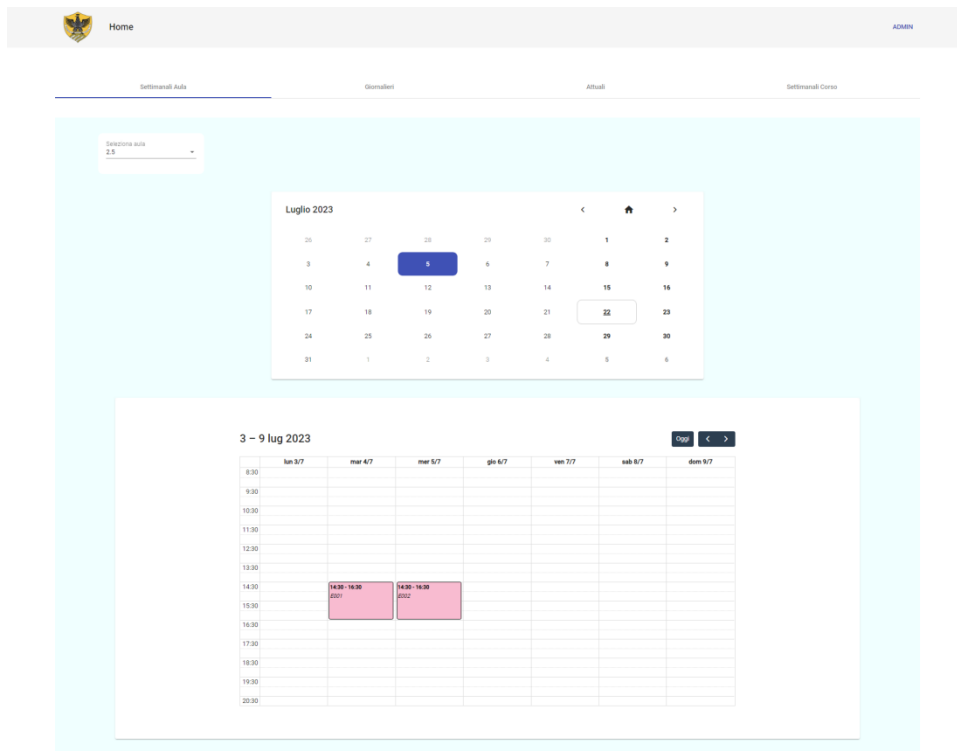
Pagina Scelta Gruppi:



In questa pagina si vedono tutti i gruppi disponibili, interagendo con uno di essi si passa alla pagina dove si potranno visualizzare le informazioni di tutte le classi con tutti gli eventi a loro associati di quel determinato gruppo.

passa
con

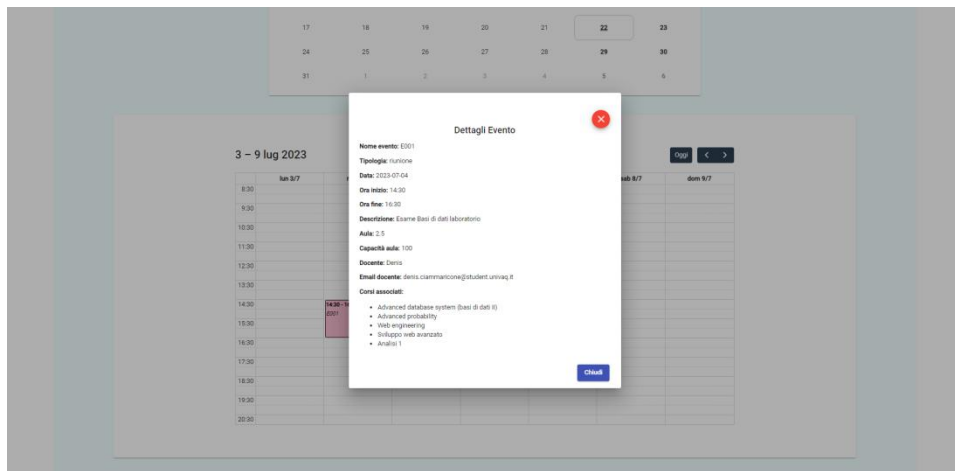
Pagina Classi ed Eventi:



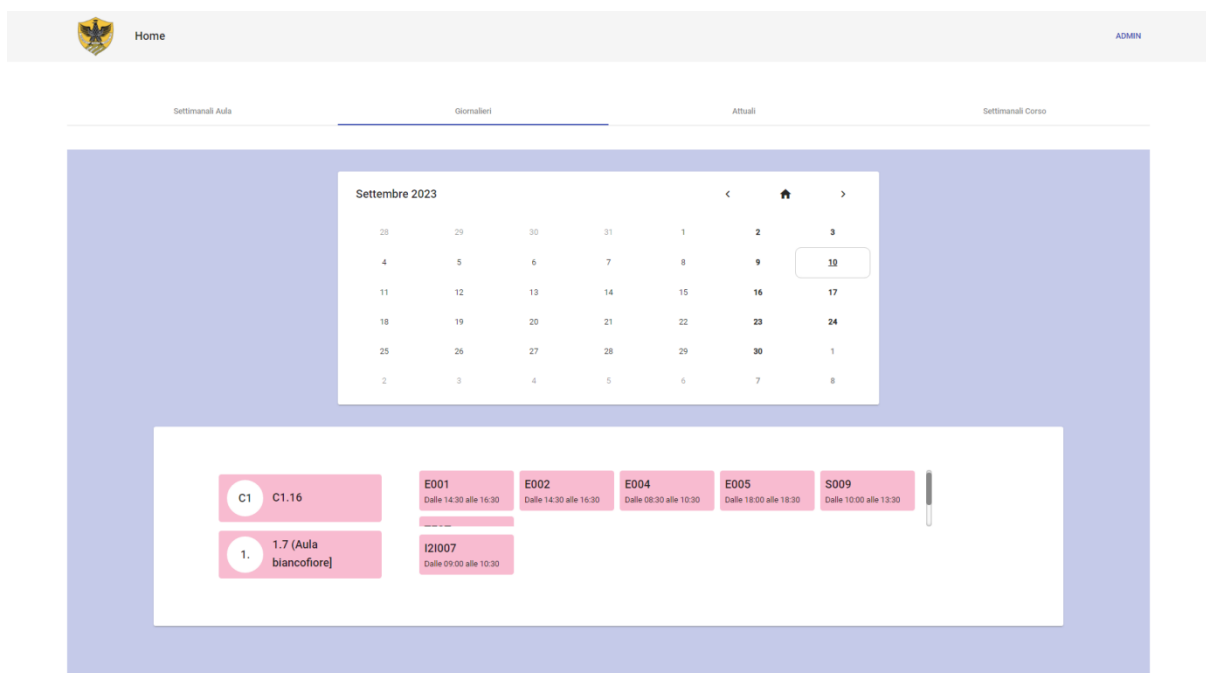
In questa pagina abbiamo 4 tasti “Settimanali Aula”, “Giornalieri”, “Attuali”, “Settimanali Corso”.

Scegliendo il primo “Settimanali Aula” otteniamo come risultato un menù a tendina in alto a sinistra “Seleziona aula” per scegliere la classe di cui ci interessa sapere gli eventi ad essa associati. Al centro dello schermo abbiamo un calendario da cui selezionare un data che, come risultato, ci genera il settimanale degli eventi associati alla nostra aula precedentemente selezionata.

Andando ad interagire con gli eventi possiamo avere maggiori informazioni su quest’ultimo.

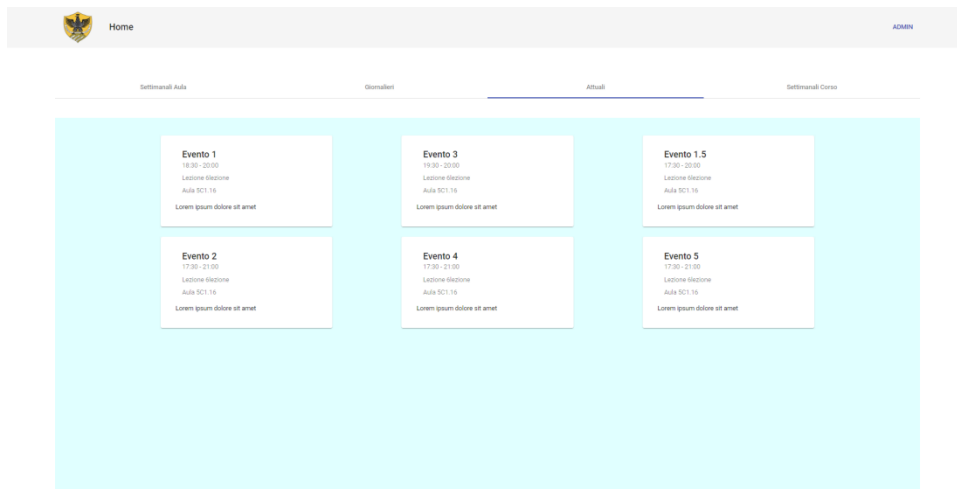


Scegliendo il secondo “Giornalieri” otteniamo



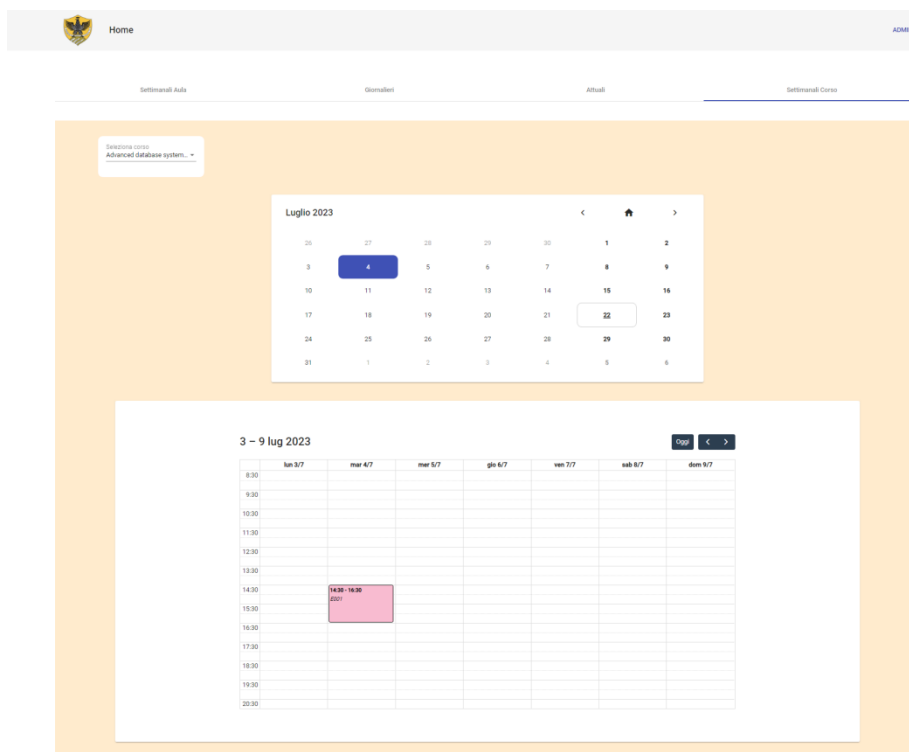
Qui otteniamo la lista degli eventi nel giorno odierno. Anche qui interagendo con gli eventi otterremo una vista con maggiori informazioni come nel caso precedente.

Scegliendo il terzo “Attuali” otteniamo



Qui otteniamo la lista degli eventi nelle prossime tre ore.

Scegliendo il quarto “Settimanali Corso” otteniamo



Qui come nel primo caso abbiamo un menù a tendina dove possiamo scegliere un Corso, il calendario dove scegliere una data. Il risultato della nostra ricerca verrà generato nel settimanale in fondo alla pagina in cui vedremo gli eventi associati al corso scelto nella data scelta.

Una volta autenticati il tasto “LOGIN” viene sostituito con il tasto “ADMIN” se vi interagiamo avremo 2 opzioni quella di logout in cui effettueremo la disconnessione e verremo riportati sulla pagina di login oppure quella di andare nella dashboard in cui poter inserire e

modificare nuovi elementi all'interno del sito. Pagina Dashboard:

The screenshot shows a web application dashboard with a light blue header. On the left, there's a sidebar with a logo and the word 'Home'. On the right, there's a user profile icon labeled 'ADMIN'. The main content area has a light blue background. At the top, there are three tabs: 'Gruppi' (with a group icon), 'Classi' (with a book icon), and 'Eventi' (with a calendar icon). The 'Eventi' tab is active. Below the tabs, there's a form titled 'Creazione / modifica Eventi'. The form has two radio buttons: 'Crea' (selected) and 'Modifica'. The form fields include: 'Nome evento' (text input), 'Data' (date picker showing 'gg/mm/aaaa'), 'Orario inizio' (time picker), 'Orario fine' (time picker), 'Descrizione' (text area), 'Seleziona un tipo' (dropdown menu), 'Seleziona un supervisore' (dropdown menu), 'Seleziona un aula' (dropdown menu), and 'Seleziona i corsi associati' (dropdown menu). At the bottom right of the form, there are two buttons: 'Pulisci' (red) and 'Conferma' (blue). Below the form, there's a blue button labeled 'Esporta in CSV'.

In questa pagina abbiamo tre scelte “Creazione/Modifica gruppi”, “Creazione/Modifica classi”, “Creazione/Modifica eventi”.

Indipendentemente da quale sceglieremo avremo due possibilità, scegliere di creare un nuovo elemento del tipo precedentemente scelto oppure la modifica di uno già esistente.

Nel caso della creazione avremo un form da riempire con tutte le informazioni dell’entità scelta una volta riempiti tutti basterà premere il tasto “Conferma” per creare il nuovo oggetto.

Nel caso della modifica avremo un menù a tendina in cui andare a scegliere tra gli oggetti già esistenti quello che vogliamo modificare. Fatto questa anche qui bisogna compilare il form con i nuovi dati ed infine premere “Conferma” per confermare la modifica dell’oggetto.

Il tasto “Pulisci” ci permette di pulire i form in maniera rapida.

Contributo partecipanti:

Stefano si è occupato principalmente dello sviluppo del front-end e ha dato supporto per lo sviluppo del back-end.

Lorenzo si è occupato principalmente dello sviluppo del database e della documentazione e ha dato un supporto allo sviluppo back-end e al front-end.