

substitution-scheduler

Autore: Laila Azza

Istituto di Istruzione Superiore "Archimede" - Treviglio

Abstract

"substitution-scheduler" è un progetto nato dall'idea di gestire digitalmente le sostituzioni all'interno dell'istituto.

Si tratta di un progetto basato su React, in cui il frontend è implementato utilizzando questa libreria insieme a JavaScript per l'interazione con l'API e i componenti dinamici all'interno delle pagine. Il backend è sviluppato in Python e include lo scraping della pagina dell'orario del sito dell'istituto, la configurazione del database e i vari endpoint dell'API.

L'assegnazione delle sostituzioni non è automatizzata, ma il programma si occupa di estrarre e classificare informazioni (ottenute tramite scraping o da documenti gestionali) relative al docente da sostituire, fornendo una serie di opzioni compatibili.

Queste opzioni sono selezionate in base a criteri quali: favorire la scelta di un docente appartenente al consiglio di classe, appartenente alla stessa area disciplinare, ecc.

Questa soluzione mira a semplificare notevolmente il lavoro del personale scolastico, consentendo una distribuzione rapida e accurata delle risorse umane in caso di imprevisti.

Introduzione

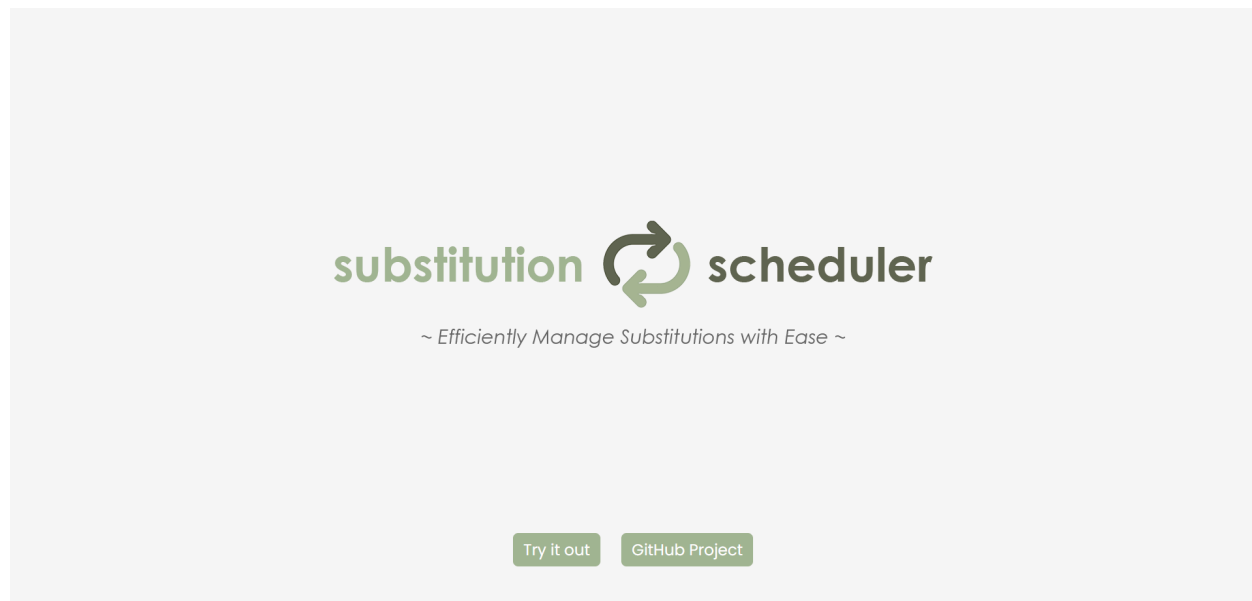
"substitution-scheduler" è un progetto sviluppato per digitalizzare la gestione delle sostituzioni dei docenti all'interno degli istituti scolastici. La necessità di un sistema efficiente e intuitivo per gestire queste sostituzioni è cruciale, data l'importanza di rispondere prontamente e con precisione alle assenze improvvise del personale docente.

Questo progetto si propone di ridurre significativamente il carico di lavoro del personale scolastico, offrendo una soluzione che consente una rapida e accurata distribuzione delle risorse umane in caso di imprevisti, migliorando così l'efficienza operativa e la qualità dell'ambiente educativo.

Materiale e metodi

Front-end

La pagina iniziale del sito mostra il logo del progetto e due pulsanti principali: uno che permette di accedere alla repository del progetto su GitHub, dove è possibile consultare il codice sorgente e la documentazione, e l'altro che permette di passare alla pagina "Try it out", che rappresenta la pagina principale del progetto.



pagina principale

Piccolo Gianluca		LUN	MAR	MER	GIO	VEN	SAB
8.00			4BT-I Lab. Informatica Bovenzi Graziano Lab. Informatica Aula 39	4FT-I Lab. Informatica Taranto Angelo Lab. Informatica Aula 39	5BT-I Lab. Informatica Bovenzi Graziano Caleca Mariella Lab. Informatica Aula 39	5FT-I Lab. Informatica Taranto Angelo Lab. Informatica Aula 39	5FT-I Lab. Informatica Taranto Angelo Lab. Informatica Aula 39
9.00			4BT-I Lab. Informatica Bovenzi Graziano Lab. Informatica Aula 39	4FT-I Lab. Informatica Taranto Angelo Lab. Informatica Aula 39	5BT-I Lab. Informatica Bovenzi Graziano A.E. Maffei Sara Lab. Informatica Aula 39	5FT-I Lab. Informatica Taranto Angelo Lab. Informatica Aula 39	5FT-I Lab. Informatica Taranto Angelo Lab. Informatica Aula 39
10.00			R.P. Vinciguerra Nicola Riggi Daniela Aula 230	3LT-I Lab. Informatica Santandrea Davide Furlano Sofia Lab. Informatica Aula 39		4BT-I Lab. Informatica Bovenzi Graziano Lab. Informatica Aula 39	3FT-I Lab. Informatica Santandrea Davide Lab. Informatica Aula 39
11.00				3LT-I Lab. Informatica Santandrea Davide Lab. Informatica Aula 39		4FT-I Lab. Informatica Taranto Angelo Lab. Informatica Aula 39	3FT-I Lab. Informatica Santandrea Davide Lab. Informatica Aula 39
12.00				5BT-I	3HT-I	3BT-I	

pagina Try it Out dopo aver selezionato un docente dal dropdown

La pagina "Try it out" è progettata per essere user-friendly e comprende tre componenti principali:

- **dropdown:** contiene la lista completa dei docenti dell'istituto, permettendo di selezionare il docente di interesse.
- **timetable del docente selezionato:** mostra l'orario delle lezioni e le eventuali disponibilità per le sostituzioni.
- **sidebar (ancora in fase di implementazione):** visualizzerà, sotto forma di componenti "tag-like", i nomi dei docenti più adatti a effettuare la sostituzione. Questi docenti vengono selezionati in base a criteri specifici, come l'appartenenza al consiglio di classe o alla stessa area disciplinare del docente assente.

Back-end

Il progetto si basa fortemente su APIs, poiché la maggior parte dei dati utilizzati dall'interfaccia utente e dalla logica di sostituzione vengono richiesti direttamente tramite esse.

Il backend è sviluppato utilizzando Python e integra le seguenti funzionalità chiave:

- **scraping**: viene eseguito dello scraping della pagina dell'orario del sito dell'istituto per raccogliere i dati necessari sugli orari dei docenti. Questa operazione è fondamentale per mantenere aggiornate le informazioni e garantire che le sostituzioni siano effettuate in modo accurato.

ORARIO DOCENTE Piccolo Gianluca

	LUN	MAR	MER	GIO	VEN	SAB
8.00		4BT-I Bovenzi Graziano Lab. Informatica Lab. Informatica Aula 39	4FT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39	5BT-I Bovenzi Graziano .Caleca Mariella Lab. Informatica Lab. Informatica Aula 39	5FT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39	5FT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39
9.00		4BT-I Bovenzi Graziano Lab. Informatica Lab. Informatica Aula 39	4FT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39	5BT-I Bovenzi Graziano .A.E Maffei Sara Lab. Informatica Lab. Informatica Aula 39	5FT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39	5FT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39
10.00		Riggi Daniela Vinciguerra Nicola .R.R. Aula 2.10	3LT-I Santandrea Davide .Furfaro Sofia Lab. Informatica Lab. Informatica Aula 39		4BT-I Bovenzi Graziano Lab. Informatica Lab. Informatica Aula 39	3FT-I Santandrea Davide Lab. Informatica Lab. Informatica Aula 39
11.00			3LT-I Santandrea Davide Lab. Informatica Lab. Informatica Aula 39		4FT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39	3FT-I Santandrea Davide Lab. Informatica Lab. Informatica Aula 39
12.00			5BT-I Bovenzi Graziano Lab. Informatica Lab. Informatica Aula 39	3HT-I Taranto Angelo .Esposito Luigi Lab. Informatica Lab. Informatica Aula 39	3BT-I Bovenzi Graziano .A.E Maffei Sara Lab. Informatica Lab. Informatica Aula 39	
13.00			5BT-I Bovenzi Graziano Lab. Informatica Lab. Informatica Aula 39	3HT-I Taranto Angelo Lab. Informatica Lab. Informatica Aula 39	3BT-I Bovenzi Graziano .A.E Maffei Sara Lab. Informatica Lab. Informatica Aula 39	
14.00						

esempio dell'orario di un docente

```
def fetch_timetable(teacher):
    teachers_dict = fetch_teachers()

    teacher_html = requests.get(f'{default_url}/{teachers_dict[teacher]}').text
    teacher_table = BeautifulSoup(teacher_html, 'lxml').find('table')

    rows = iter(teacher_table.find_all('tr'))
    days = [day.strip() for day in next(rows).text.split() if day.strip()]
    hours = []

    timetable = [[] for _ in days]

    for time_index, row in enumerate(rows):
        columns = iter(row.find_all('td'))
        lesson_time = next(iter(next(columns).find_all('p'))).text.strip()
        hours.append(lesson_time)
        day_index = 0
        for cell in columns:
            while day_index < len(days) and len(timetable[day_index]) > time_index:
```

```

        day_index += 1
        lesson_duration = int(cell['rowspan'])
        for _ in range(lesson_duration):
            timetable[day_index].append(extract_data(cell))

timetable_data = {}

for day, lessons in zip(days, timetable):
    timetable_data[day] = {}
    for lesson_time, lesson in zip(hours, lessons):
        timetable_data[day][lesson_time] = lesson

return timetable_data

```

code snippet con la main function di scraping dell'orario di un docente

- **database:** il sistema include la configurazione di un database che memorizza tutte le informazioni rilevanti sui docenti, sugli orari e sulle sostituzioni. Il database è strutturato per consentire un accesso rapido e efficiente ai dati.
- **API end-points:** sono stati sviluppati diversi endpoint API per gestire le richieste dal frontend. Alcune delle funzionalità che questi endpoint permettono sono: recuperare la lista dei docenti e ottenere l'orario del docente selezionato.

```

@app.route('/<path:teacher_name>', methods=['GET'])
def timetable(teacher_name):
    decoded_teacher_name = unquote(teacher_name)
    try:
        timetable = fetch_timetable(decoded_teacher_name)
        response_JSON = json.dumps(timetable, ensure_ascii=False, indent=2)
        response = Response(response_JSON, content_type='application/json; charset=utf-8')

        return response, 200
    except KeyError:
        return jsonify({'error': 'Teacher not found'}), 404

@app.route('/<path:teacher_name>_subjects', methods=['GET'])
def subjects(teacher_name):
    decoded_teacher_name = unquote(teacher_name)
    try:
        subjects = fetch_subjects(decoded_teacher_name)
        response_JSON = json.dumps(subjects, ensure_ascii=False, indent=2)
        response = Response(response_JSON, content_type='application/json; charset=utf-8')

        return response, 200
    except KeyError:
        return jsonify({'error': 'No results found.'}), 404

@app.route('/<path:teacher_name>_classes', methods=['GET'])
def teacher_classes(teacher_name):
    decoded_teacher_name = unquote(teacher_name)
    try:

```

```
        classes = fetch_teacher_classes(decoded_teacher_name)
        response_JSON = json.dumps(classes, ensure_ascii=False, indent=2)
        response = Response(response_JSON, content_type='application/json; charset=utf-8')

        return response, 200
    except KeyError:
        return jsonify({'error': 'No results found.'}), 404

@app.route('/classes', methods=['GET'])
def classes_list():
    classes_dict = fetch_classes()
    classes_list = list(classes_dict.keys())
    return jsonify(classes_list), 200
```

code snippet con alcuni end-points della Flask API

Oltre agli endpoint creati per lo scraping dell'orario scolastico, il sistema utilizzerà le Google Spreadsheets API per la lettura e l'eventuale aggiornamento di una "dashboard".

Questa dashboard consiste in una spreadsheet che contiene una tabella con una lista di docenti, le ore di supplenza da assegnare, le ore già assegnate e il saldo rimanente (l'unica informazione veramente necessaria per lo scopo del progetto).

Funzionamento

Il progetto non mira a automatizzare completamente l'assegnazione delle supplenze, poiché ciò comporterebbe la necessità di analizzare una vasta gamma di situazioni, molte delle quali dipendenti da variabili legate alla burocrazia scolastica.

Invece, "substitution-scheduler" è concepito come uno strumento di supporto che semplifica l'aggiornamento e la distribuzione delle informazioni tra i diversi uffici scolastici, come la vicepresidenza e la segreteria scolastica.

Questo strumento è stato progettato per essere utilizzato da collaboratori della presidenza, come la figura del vicepresidente, che necessitano di un metodo efficiente per gestire le sostituzioni dei docenti in modo tempestivo e accurato.

La spiegazione seguente si basa sulla visione del progetto una volta completato.

All'interno delle celle della tabella oraria di ciascun docente sarà possibile interagire direttamente. Al clic, comparirà una sidebar che permetterà la visualizzazione dei docenti più idonei ad effettuare una sostituzione.

I **criteri di "selezione"** per la creazione della lista dei docenti "più adatti" sono basati su diversi fattori:

1. **appartenenza al corpo docenti della classe:** si favorisce un docente che insegna nella stessa classe in cui verrà svolta la supplenza, garantendo una maggiore coerenza con l'ambiente e gli studenti;
2. **area di insegnamento:** si privilegia la supplenza di un docente appartenente alla stessa area disciplinare (informatica, italiano/storia, chimica, ecc.). Ad esempio, in caso di assenza di un docente di informatica, verrà favorito come sostituto un docente dell'area,

che potrebbe insegnare informatica, sistemi e reti, TPSI, GPOI, sia in forma teorica che pratica.

3. **idoneità all'insegnamento di una stessa materia:** simile alla casistica precedente, si darà priorità alla supplenza di un docente che insegna la stessa materia, garantendo una maggiore coerenza didattica e competenza specifica.

Altri principali aspetti da considerare sono:

- le compresenze, che coinvolgono un docente teorico e un docente ITP. Generalmente, un docente ITP coinvolto in una compresenza viene spesso assegnato a effettuare supplenze in caso di necessità;
- docenti con ore di R.P. (colloqui) una volta terminati i periodi in cui possono essere prenotati/fatti colloqui genitori-docenti;
- docenti con FS (funzione strumentale): figure idonee a svolgere le funzioni strumentali per la realizzazione e la gestione del POF (Piano dell'Offerta Formativa);

Questa lista non è esaustiva e non copre tutte le possibili casistiche della complessa struttura gerarchica della scuola. Tuttavia, evidenzia ulteriormente la scelta di non automatizzare il delicato processo dell'assegnazione delle sostituzioni.

Risultati e discussione

Nonostante l'implementazione delle funzionalità principali del progetto, il sistema non è stato completato a causa dell'accesso limitato ai documenti amministrativi necessari. Questa mancanza ha impedito l'acquisizione di dati completi e accurati per la configurazione del database e il perfezionamento degli algoritmi di selezione delle sostituzioni. I test preliminari hanno comunque dimostrato il potenziale del sistema nel semplificare il processo di gestione delle sostituzioni.

La capacità del backend di effettuare lo scraping e di interfacciarsi con il front-end tramite API è stata verificata con successo.

Tuttavia, la mancanza di dati completi ha limitato la possibilità di eseguire test più approfonditi e di valutare l'efficacia del sistema in situazioni reali.

Conclusione

Il progetto ha dimostrato di poter offrire una soluzione innovativa per la gestione delle sostituzioni dei docenti, riducendo il carico di lavoro del personale scolastico e migliorando l'efficienza operativa dell'istituto.

Sebbene non sia stato completato a causa dell'accesso limitato ai documenti amministrativi, il sistema ha mostrato risultati promettenti nei test preliminari. Per il futuro, l'accesso completo ai dati amministrativi sarà essenziale per perfezionare e completare il progetto.

Con le necessarie informazioni, il sistema potrà essere ulteriormente sviluppato e testato, garantendo una gestione delle sostituzioni ancora più efficiente e accurata.

Inoltre, l'implementazione della sidebar e il miglioramento degli algoritmi di selezione dei docenti permetteranno di sfruttare appieno il potenziale del progetto.