PizzAdvisor

Federico Ranocchia 310813

Tecnologie utilizzate

REACT : è una libreria JavaScript/Typescript per il front-end utilizzata per la creazione di interfacce utente.

EXPRESS: Un framework per applicazione web/api basato su node.js per la creazione di un back-end.

MariaDB: un database system basato su modello razionale. MariaDB è un fork di MySQL, quindi le funzioni sono molto simili se non uguali a quest'ultimo.

REACT

L’ approccio è basato su componenti riutilizzabili che semplificano lo sviluppo e la manutenzione.

React utilizza JSX per definire componenti UI, che si presenta molto simile al normale HTML.

```
return (
    <div>
        {place_id &&
            <div>
                <PlaceData place_id={place_id} more_details={true}/>
                <ViewReview place_id={place_id} />
                <SendReview place_id={place_id} />
            </div>
        }
    </div>
)
```

```
return (
  <div>
    <h2>Recensioni:</h2>
    {reviews.map((review, index) => (
      <div key={index}>
        <h3>Recensione #{index + 1}</h3>
        <p>Nome: {review.first_name}</p>
        <p>Cognome: {review.last_name}</p>
        <p>Oggetto: {review.review_object}</p>
        <p>Corpo: {review.review_body}</p>
        <p>Data di creazione: {new Date(review.created).toLocaleString()}</p>
        <p>Modificato: {review.modified ? 'Sì' : 'No'}</p>
      </div>
    ))}
  </div>
);
```

Librerie utilizzate di React

React non è un framework come angular, ma soltanto una libreria e quindi necessita di librerie esterne JavaScript/Typescript

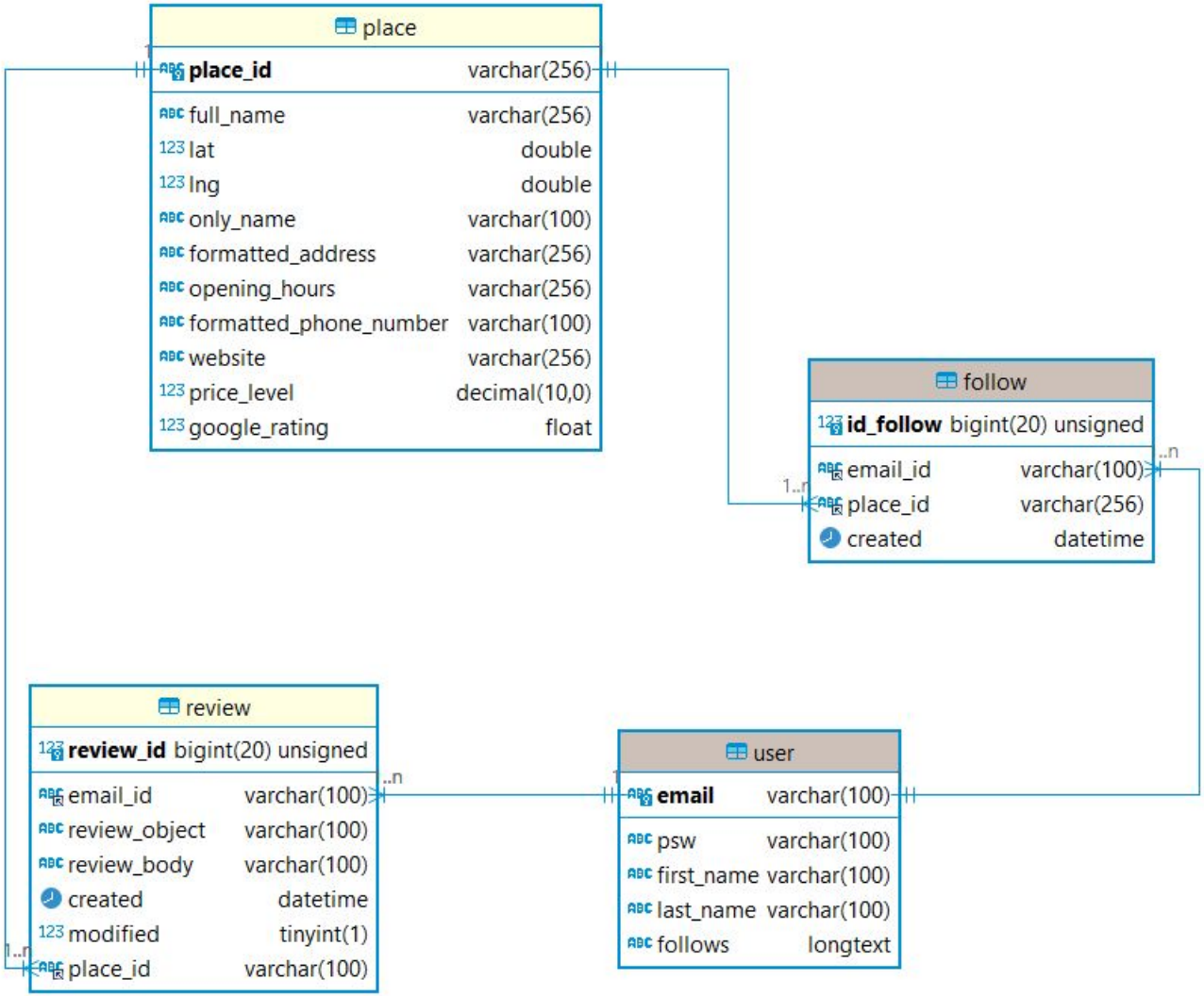
- react-google-maps/api: usa l'api di Google per creare i componenti map, marker e infoview
- use-places-autocomplete: usa l'api di Google per mostrare suggerimenti di Google Maps
- react-router-dom: per la creazione di route per React
- tanstack/react-query: gestisce in maniera intelligente la cache delle fetch
- js-cookie: per l'utilizzo dei cookie (utile per l'autenticazione)

REST

REST é un’architettura stateless che si basa su HTTP, nel nostro caso i messaggi vengono scambiati in formato JSON.

Esempio di ricezione da parte di express in formato JSON

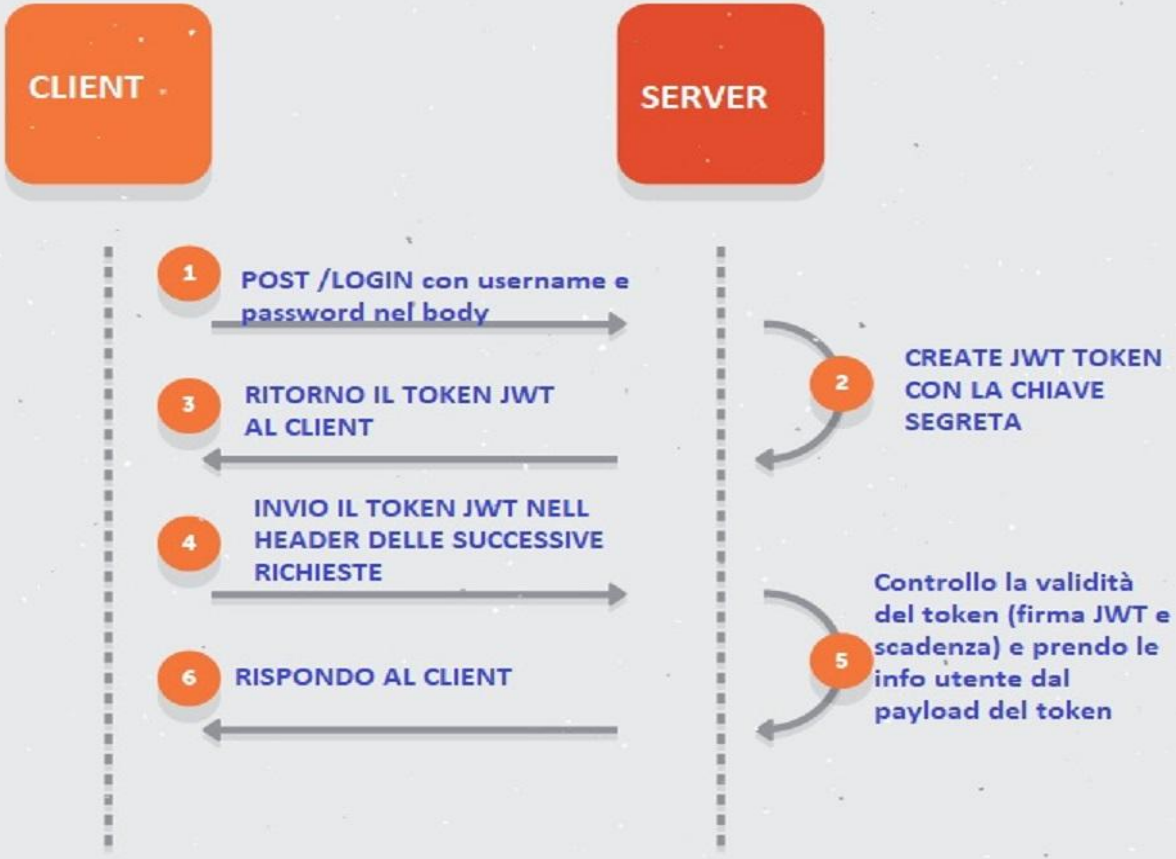

```
{
    "message": "Get detailed place OK",
    "det_place": {
        "place_id": "ChIJo7QUJoegLhMR289JoJPvntI",
        "full_name": "Testone Perugia Centro Storico, Piazza Giacomo Matteotti, Perugia, PG, Italia",
        "lat": 43.11093030000001,
        "lng": 12.3897769,
        "only_name": "Testone Perugia Centro Storico",
        "formatted_address": "Piazza Giacomo Matteotti, 24, 06121 Perugia PG, Italia",
        "opening_hours": "[\"lunedì: 12:00–15:00, 19:00–23:00\",\"martedì: 12:00–15:00, 19:00–23:00\",\"mercoledì:
            19:00–23:00\",\"sabato: 12:00–15:00, 19:00–23:00\",\"domenica: 12:00–15:00, 19:00–23:00\"]",
        "formatted_phone_number": "075 937 5924",
        "website": "https://www.magnatestone.it/",
        "price_level": "1",
        "google_rating": 4.1
    }
}
```


place
place_id varchar(256)
full_name varchar(256)
lat double
lng double
only_name varchar(100)
formatted_address varchar(256)
opening_hours varchar(256)
formatted_phone_number varchar(100)
website varchar(256)
price_level decimal(10,0)
google_rating float
follow
id_follow bigint(20) unsigned
email_id varchar(100)
place_id varchar(256)
created datetime
review
review_id bigint(20) unsigned
email_id varchar(100)
review_object varchar(100)
review_body varchar(100)
created datetime
modified tinyint(1)
place_id varchar(100)
user
email varchar(100)
psw varchar(100)
first_name varchar(100)
last_name varchar(100)
follows longtext
1..n
..n

Token Authentication

Tecnica utilizzata per effettuare l'autenticazione in maniera stateless:

- Express utilizza la libreria jsonwebtoken per la gestione dei token
- La password inoltre viene hashata prima di essere inserita nel database
- Il token viene salvato nel Front End tramite l'utilizzo di cookies



Funzionalitá

Una breve lista sulle principali funzionalitá del web-service

- Autenticazione: si possono creare nuovi profili, effettuare l'accesso ed effettuare il logout.
- Mappa: visualizzazione della mappa con i marker salvati nel database. I marker, se selezionati, mostreranno informazioni relative al ristorante/pizzeria.
- Ricerca: si possono effettuare ricerche "assistite" con le api di google per trovare ristoranti/pizzerie.
- Mostra/crea: recensioni: l'utente, se autenticato, puó scrivere delle recensioni.
- Preferiti: l'utente puó salvare i ristoranti/pizzerie per poi visualizzarli piú tardi.