EVENTS SEARCH APP

Created by: **ADITYA PATHAK**

Hi! My name is Aditya Pathak, and I am an aspiring web developer. I am currently doing a B.Sc. degree course in Programming and Data Science from IIT Madras, which covers the core fundamentals of web/software development, as well as the fundamentals of Data Science. I have also completed a Bachelor's Degree in Mathematics from University of Delhi.

The '**Events Search Web App**' is a very simplistic web app which I have developed using Python's Flask as the backend framework, as well as using Python modules like requests and pyquery for scraping data from the web. For the frontend, I have used Vue.js to render webpage components dynamically for a single-page like application. I have also used Bootstrap CSS and JS to make the views responsive to various devices, for a smooth User Experience.

As mentioned above, I have divided the app into the frontend and backend parts. These parts communicate with each other using REST APIs. The frontend sends an HTTP request to the backend for data retrieval, and the backend sends the appropriate response back, after proper input validation.

This app does not store any data. When the backend receives a request for data, it scrapes an events website (eventbrite.com) for the data directly. It then organizes and compiles that data in the form a python iterable, which is then sent as JSON data to the client. If any errors are encountered, they are appropriately handled and the error messages are sent to the frontend along with the proper status codes.

As given in the problem statement, when the user clicks the Get Tickets button for any event, a form opens up, in which they are asked to enter their Email ID. That email is sent to the backend, which returns a url for the actual Get Tickets page, which the frontend then redirects to. I was unclear on what to do with the email once the user submits it, so I just printed it on the server console using the print function.

I also didn’t understand how to implement the 24hr script part. Since the app always fetches the updated data from the web, and then displays it immediately, hence the updation of data becomes irrelevant. But given that it has to be done, I didn’t understand how to store it. All the events of a particular city at a particular time can be stored quite easily, but doing it for any city the user might search for? I tried to implement it, but I got confused, so I left it.

That is all this app does. Thank you.