

Design a RESTful API:

You have to build a CRUD API. You can use any framework/language you prefer. The API should have the following endpoints:

| HTTP Method | Endpoint | Description |
|-------------|---|---|
| POST | /api/events | Create a new event record. |
| GET | /api/events/{id} | Read an event record by referencing its <i>id</i> . |
| GET | /api/events?page[number]=1&page[size]=5 | Read all events, and events can be paginated by providing page[number] and page[size] parameters. |
| PUT | /api/events/{id} | Update an existing event record by referencing its <i>id</i> . |
| DELETE | /api/events/{id} | Delete an event record by referencing its <i>id</i> . |

Table 1: API endpoints

What we want from you is as follows:

1. Implement all the endpoints mentioned in **Table 1**.
2. Implement the necessary model and migration for the Event record. The event record should have three attributes: Name (String), Location (String) and Date (DateTime)
3. Persists events data to a Postgres or MySQL database

Design an App:

You have to build a front-end app. You can use any front-end framework/language. The look and feel of the front end should be similar to the below screenshots (do not worry about the look and feel, if you can build the App with the given look and feel, you will get bonus points):

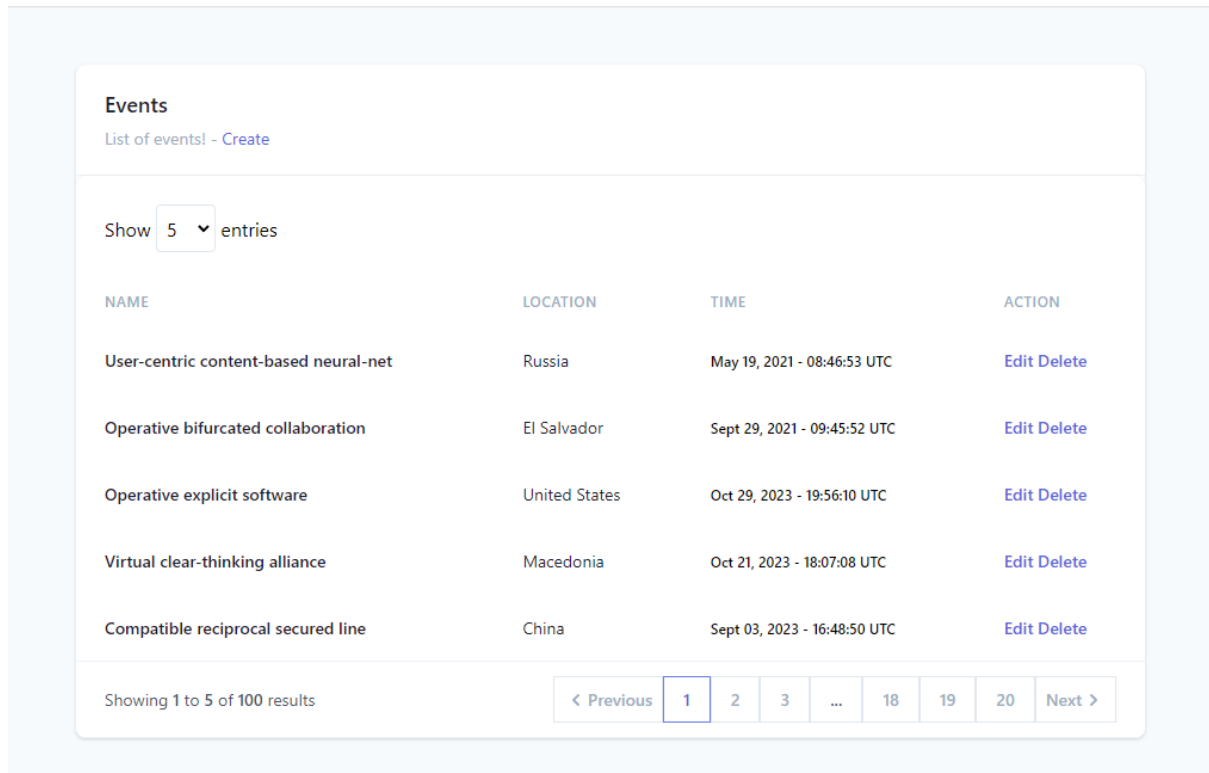


Figure 1: List of events page UI

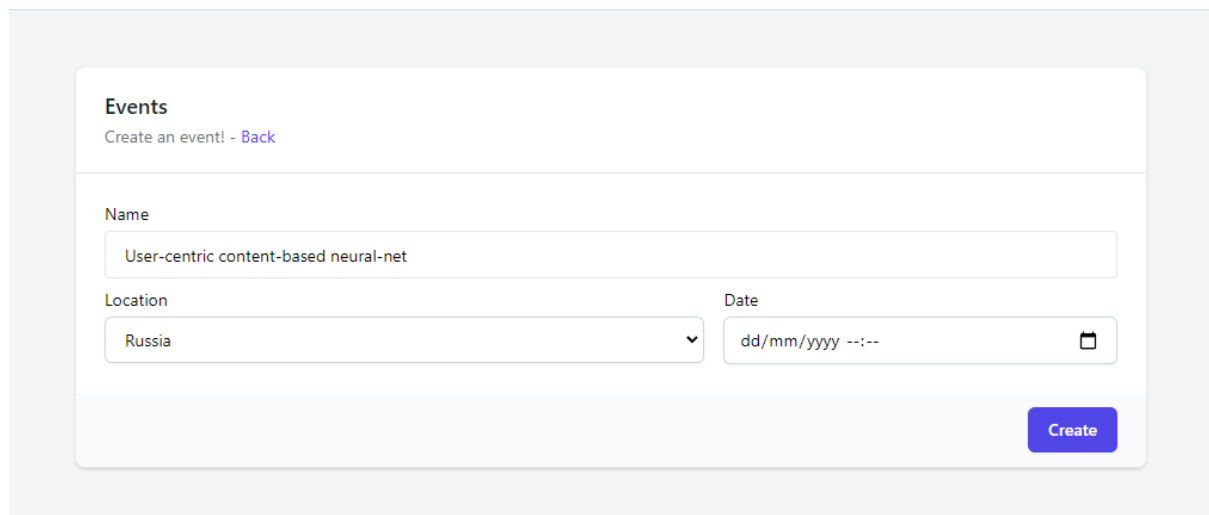


Figure 2: Create an event page UI

Events
Edit an event! - [Back](#)

Name
User-centric content-based neural-net

Location
Russia

Date
dd/mm/yyyy --:--

[Update](#)

Figure 3: Edit an event page UI

What we want from you is as follows:

1. Design the **Create an event page** as shown in **Figure 2** and use **POST /api/events** endpoint
 - a. **Back** link should return to List of events page (see #3)
2. Design the **Edit an event page** as shown in **Figure 3** and use **PUT /api/events/{id}** endpoint
 - a. **Back** link should return to List of events page (see #3)
3. Design the **List of events page** as shown in **Figure 1** and use **GET /api/events?page[number]=1&page[size]=5** endpoint.
 - a. List of events has to be paginated as shown in **Figure 1**, user can specify how many events to load per page. Default is 5 entries.
 - b. **Create** link should show Create an event page (see #1)
 - c. Individual event record can be edited via **Edit** link (see #3)
 - d. Individual event record can be deleted via **Delete** link and use **DELETE /api/events/{id}** endpoint

N.B. Use the GIT version control system and track your progress with commits, then publish the repository in github.com as a Public repository. Make sure to write down instructions on README about how to run your project.