**Event Listing Mini APP**

**Overview**

This is a mini-application that lists events and shows detailed information for each event. The application is using a custom WordPress plugin to manage event data, and Next.js for the front-end to display the events.

**Table of Contents**

* Installation
* Usage
* Architecture
* API Documentation

**Installation**

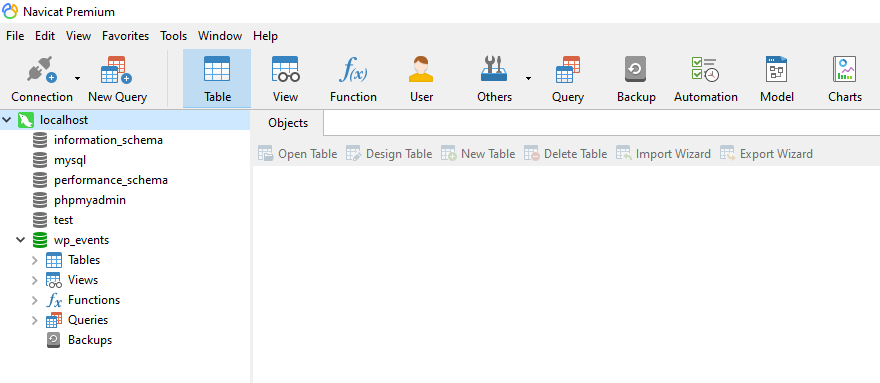
**Back-end (WordPress) Installation**

Set up a local server environment to run PHP and MySQL services on a local machine.

Log in to the MySQL server using a tool like phpMyAdmin, Navicat or the MySQL command.

Create a new database for my WordPress installation.

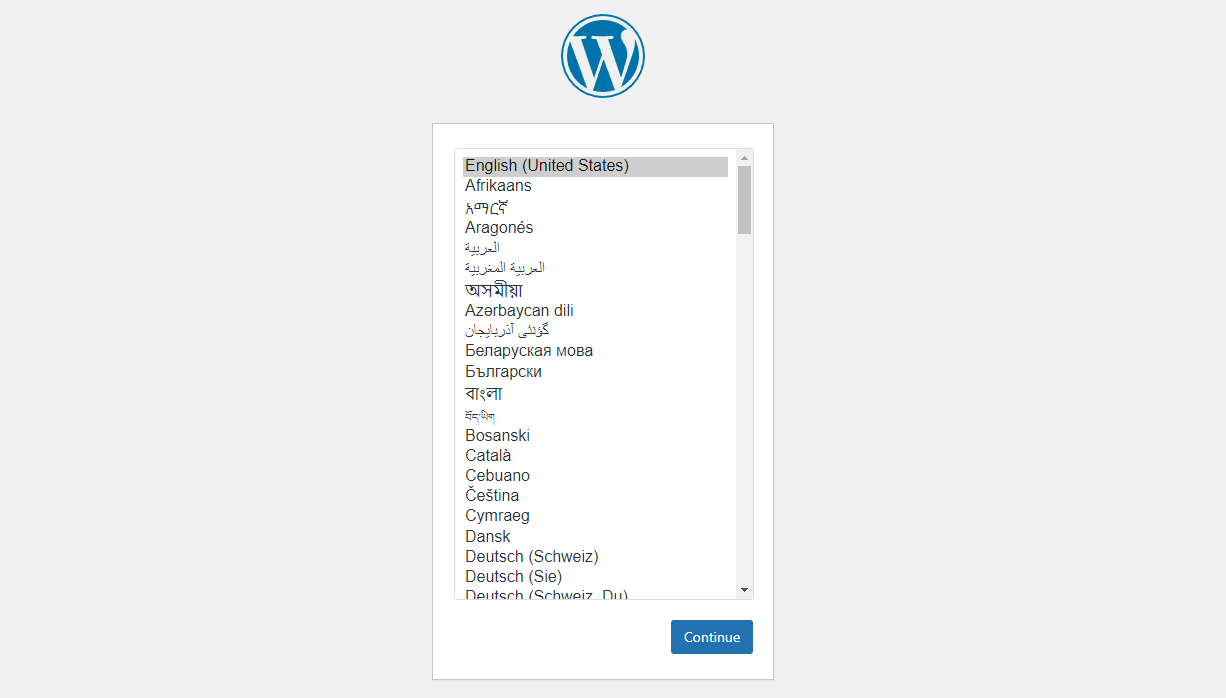
Choose a name for a database (e.g. "wp\_events").

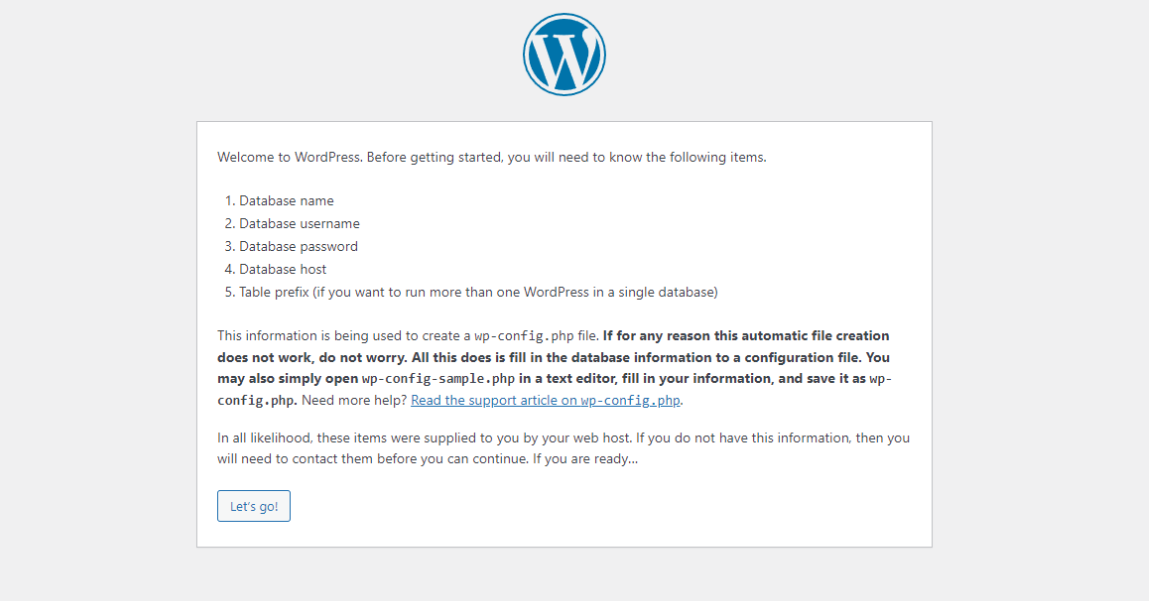


Download WordPress project from <https://wordpress.org/download/.>

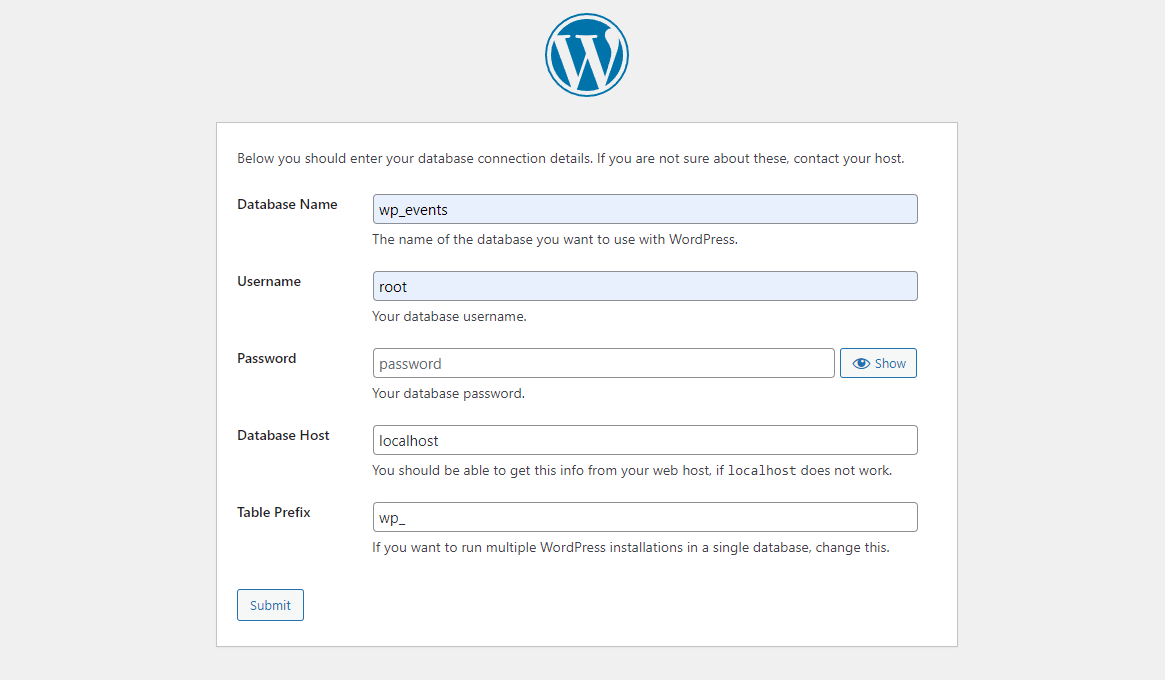
Run WordPress project by Apache service.

<http://localhost/wordpress/wp-admin/setup-config.php>

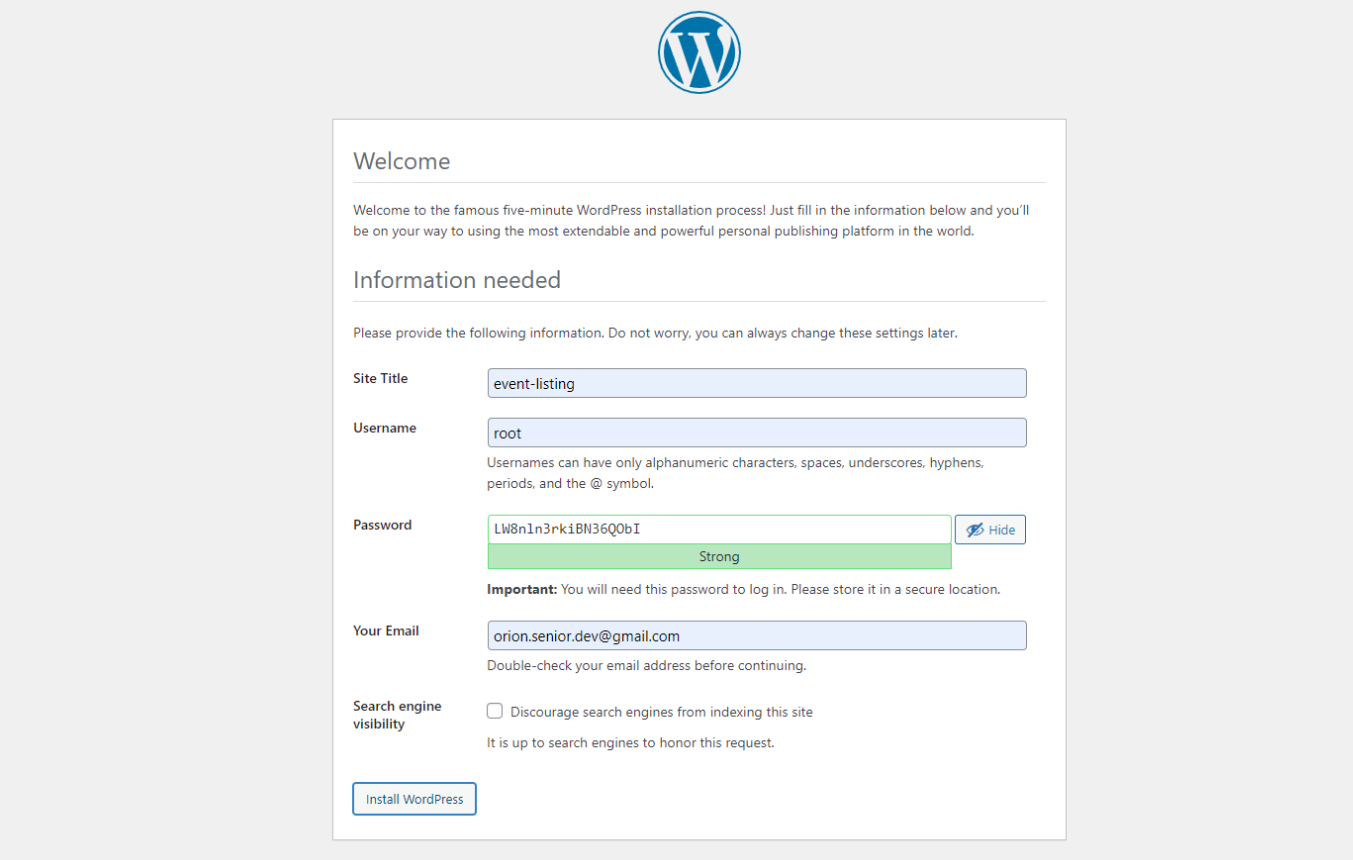




Configure information to connect Database.

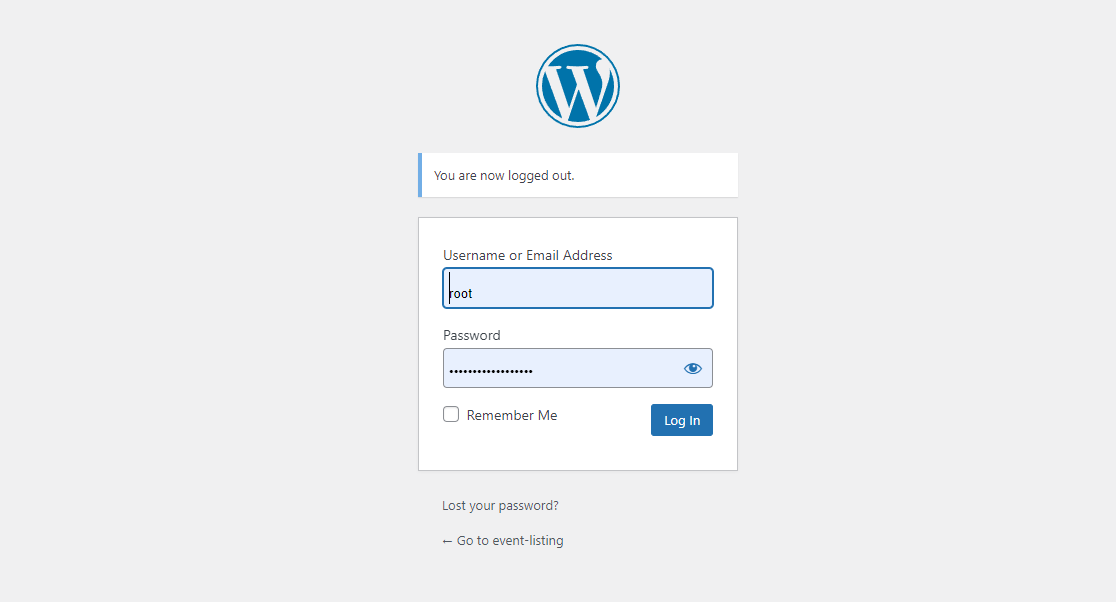


Configure personal information of wordpress platform.



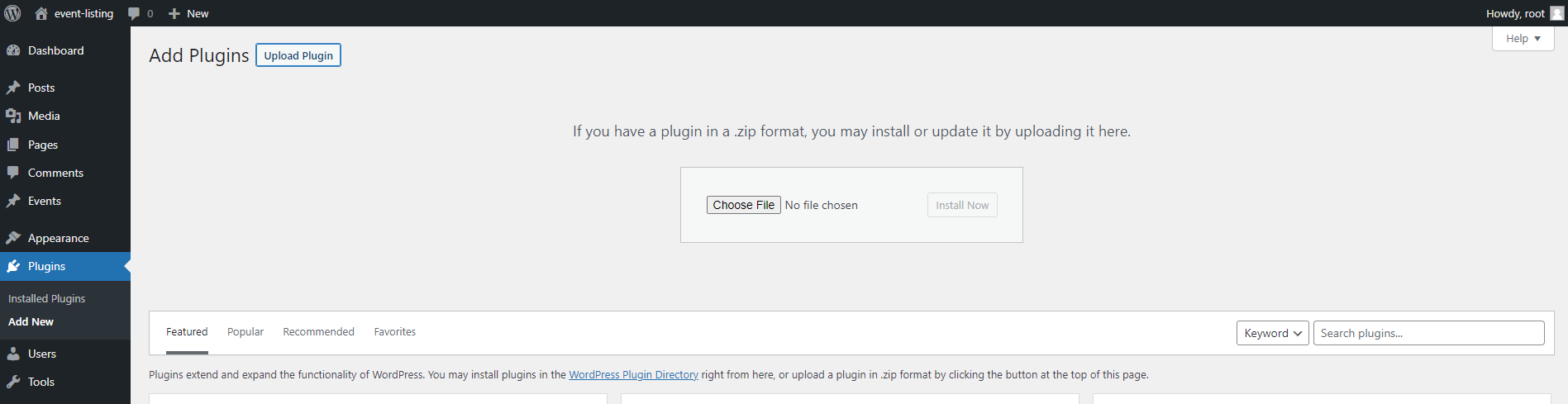
Log in to WordPress website's admin dashboard.

<http://localhost/wordpress/wp-admin/> .



In the left-hand menu, click on "Plugins" and then select "Add New".

At the top of the page, click on the "Upload Plugin" button.

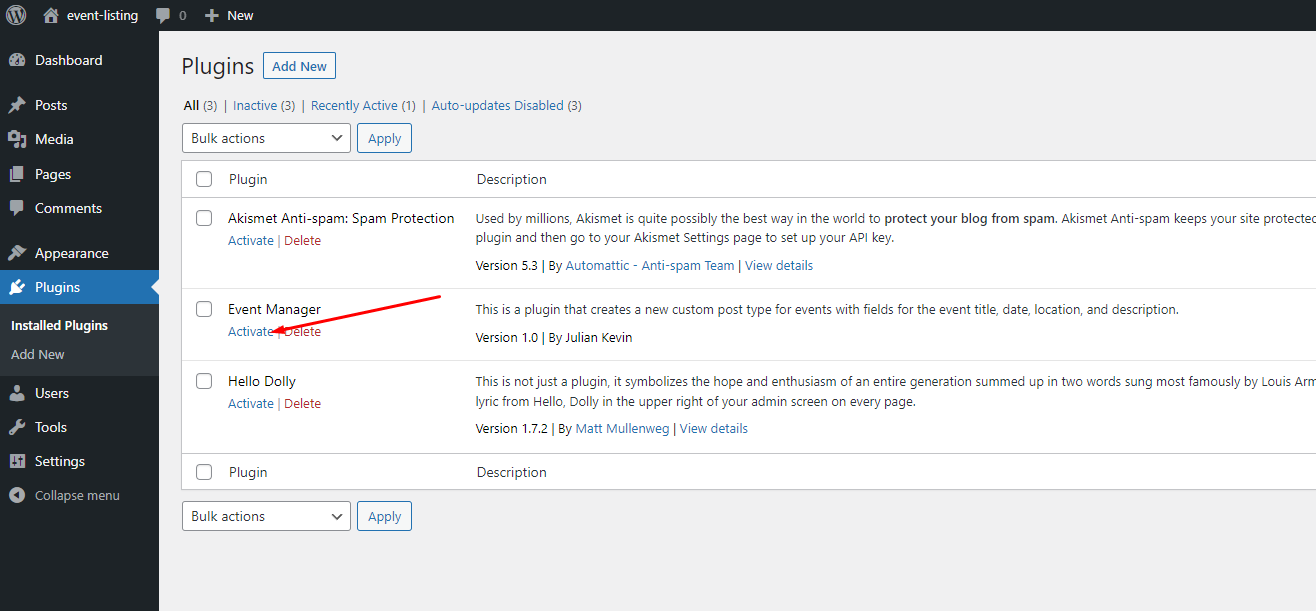


Click on the "Choose File" button and select the event-manager.zip file containing event-manager plugin.

Once the file is selected, click on the "Install Now" button.

After the plugin is uploaded, you will be redirected to a page where you can activate the plugin.

Click on the "Activate Plugin" button to activate custom plugin.



**Front-end (Next.js) Installation**

Install Node.js and NPM or Yarn on system.

Clone Next.js app from <https://github.com/unicorn-talent/event-listing.git.>

Navigate to the project directory.

Set API endpoint on .env.local file.

NEXT\_PUBLIC\_API\_URL=http://localhost/wordpress/wp-json/event-manager/

Install Dependecies

npm install

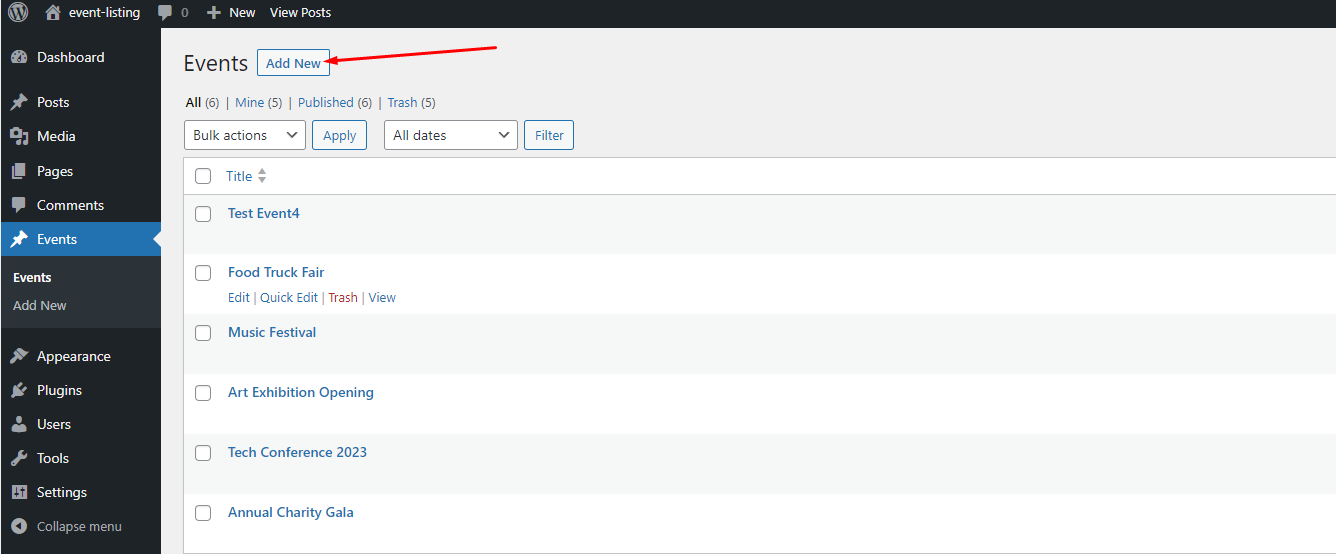
or

yarn install

**Usage**

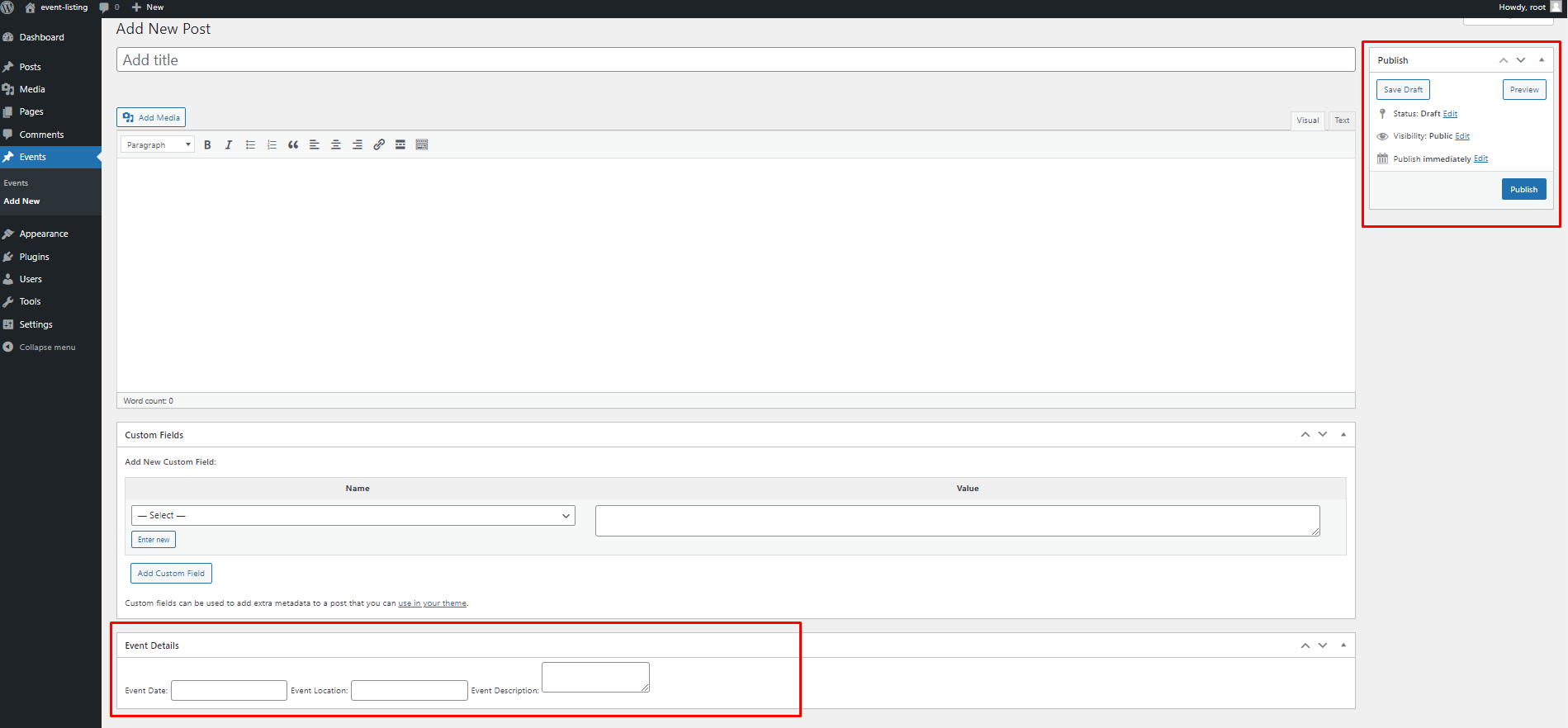
**Back-end (WordPress)**

In the left-hand menu, click on "Events" and then select "Add New" at the top of the page.



You can add new events with the custom fields.

When creating or editing a post, you will see the custom fields for event\_date, event\_location, and event\_description.



Fill in the relevant information for each field, and publish or update the post.

**Front-end (Next.js)**

Start the Development Server

npm run dev

# or

yarn dev

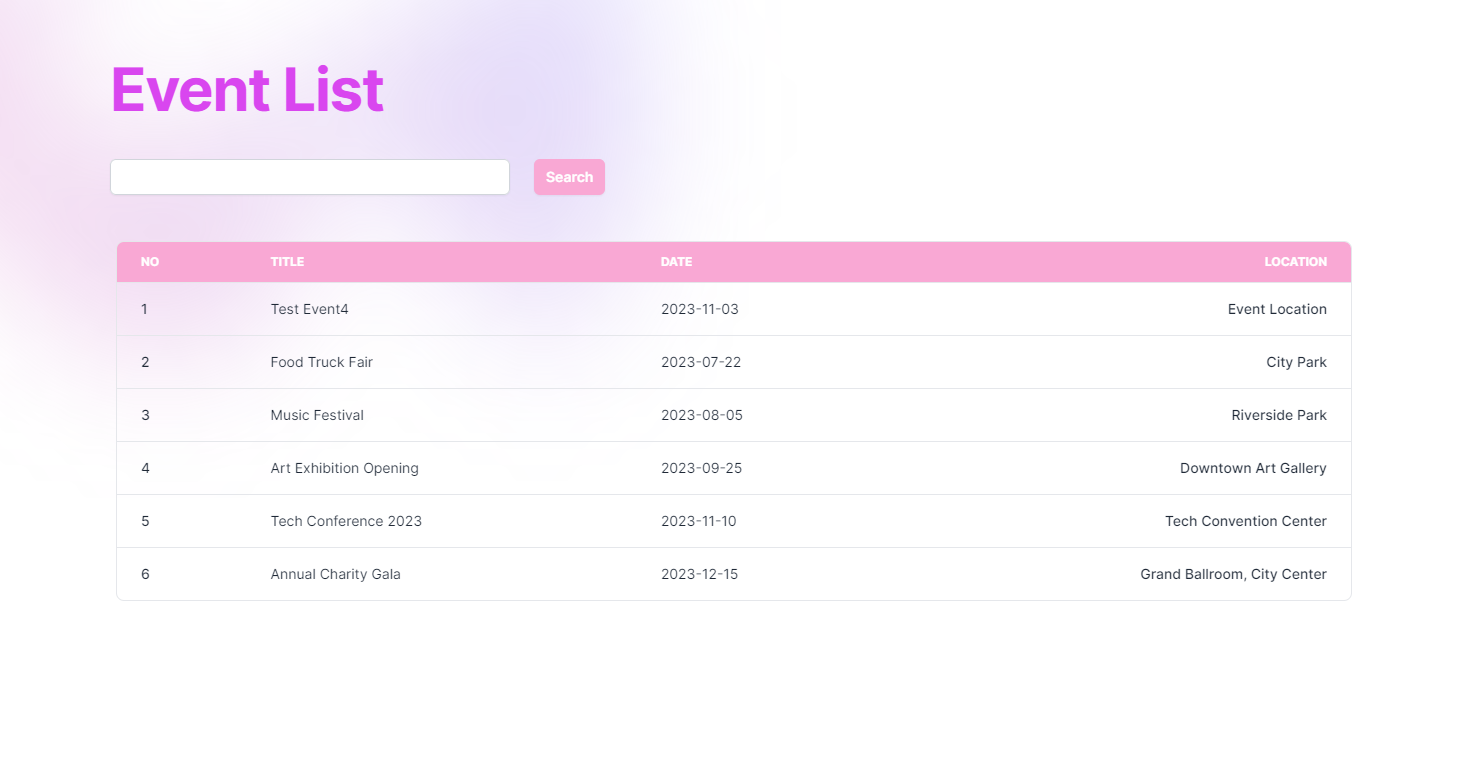
# or

pnpm dev

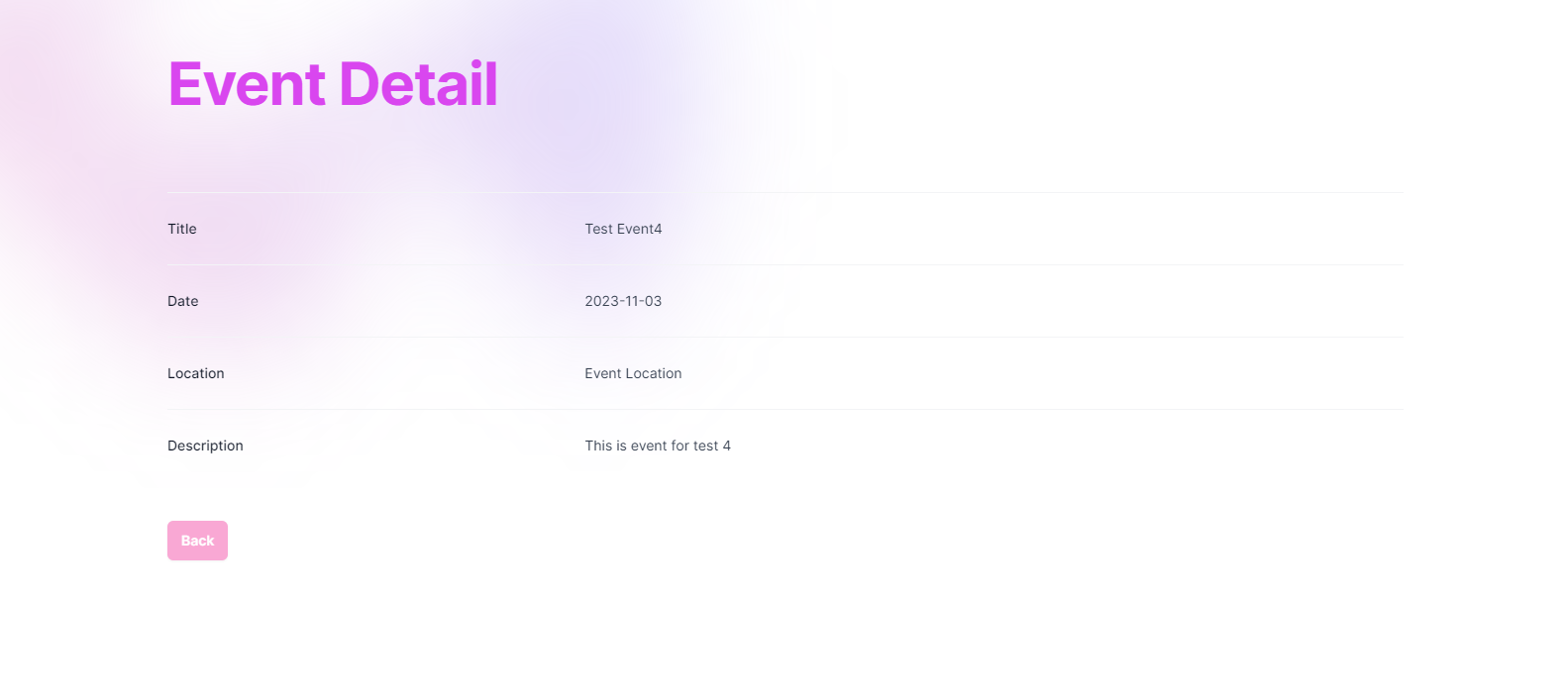
# or

bun dev

Open http://localhost:3000 with your browser to see the result.



Open http://localhost:3000/events/[event\_id] to see detail page.



**Architecture**

**Back-end**

A custom WordPress plugin to creates a new custom post type for events with fields for the event title, date, location, and description.

The plugin includes endpoints for retrieving event lists and individual event details.

- Get All Events Function

get\_all\_events\_callback()

A function to retrieve all events from the WordPress database.

- Get Event By ID Function

get\_event\_by\_id\_callback($Request)

A function to retrieve an event by its ID from the WordPress database.

- Get Event By Search Text Function

get\_events\_by\_title\_callback($Request)

A function to retrieve events by title search from the WordPress database.

**Front-end**

Front-end consists of Event Listing Page and Event Detail Page.

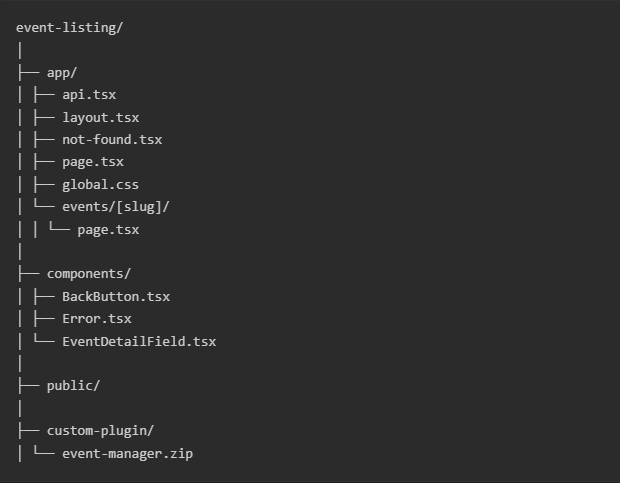
Event Listing Page displays a list of events from the custom WordPress plugin using the WP REST API.

If there are no events in WordPress database or an error occurred in the REST API server, you will see an error page on the Event Listing Page.

Event Listing Page allows you to search for specific events by their titles.

Event Detail Page displays more information about the event including a Title, Date, Location, Description, fetched from the WP REST API.

If event id was wrong, you will see a not found page on the Event Detail Page.



**API Documentation**

- Get All Events Function

<http://localhost/wordpress/wp-json/event-manager/get-all-events>

Function

function get\_all\_events\_callback() {

…

}

Response JSON data.

{

"code": 200,

"message": "Success",

"data": [

{

"id": ID,

"title": event\_title,

"date": event\_date,

"location": event\_location

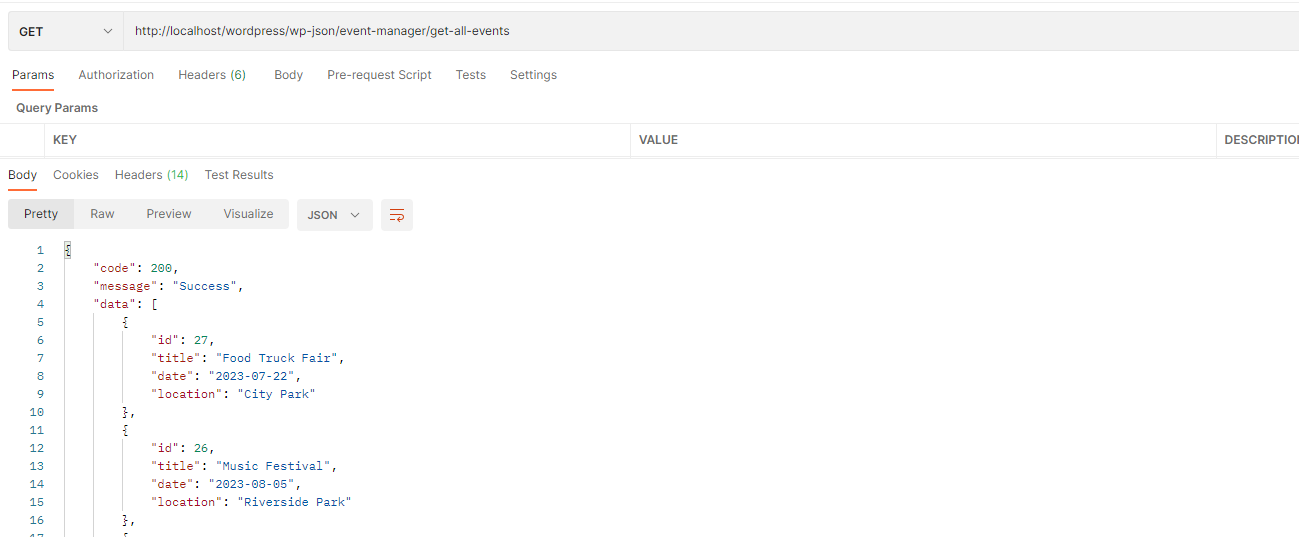
},

...

]

}

Test on Postman.



- Get Event By ID Function

<http://localhost/wordpress/wp-json/event-manager/get-event/[event_id]>

Function

function get\_event\_by\_id\_callback($request) {

...

}

Response JSON data

{

"code": 200,

"message": "Success",

"data": {

"title": event\_title,

"date": event\_date,

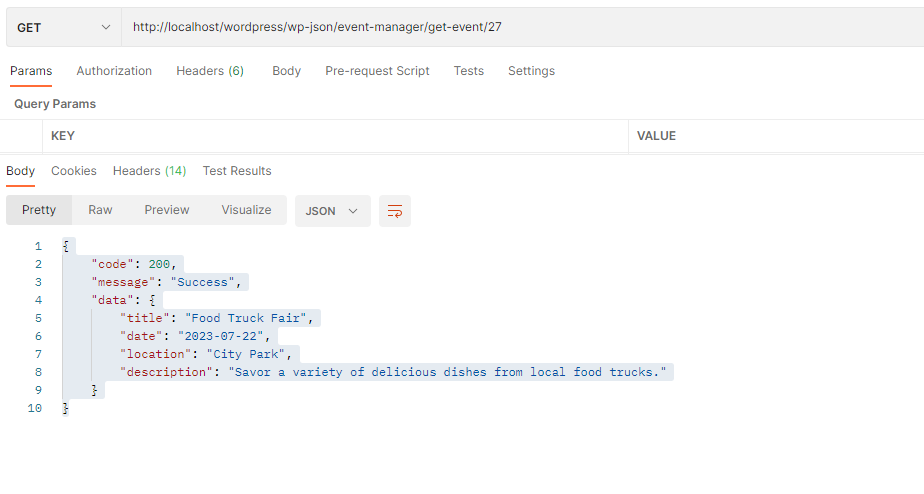
"location": event\_location,

"description": event\_description

}

}

Test on Postman



- Get Event By Search Text Function

<http://localhost/wordpress/wp-json/event-manager/get-events-by-title?search_text=text>

Function

function get\_events\_by\_title\_callback($request) {

...

}

Response JSON data

{

"code": 200,

"message": "Success",

"data": [

{

"id": ID,

"title": event\_title,

"date": event\_date,

"location": event\_location

}

]

}

Test on Postman

