Endpoint Documentation Event Organizer API

Base URL:

http://localhost:3000/api

User Endpoints

1. Register User

URL: /users/register Method: POST

Description: Register new users.

Request Body:

```
1 {
2  | "name": "John Doe",
3  | "email": "john.doe@example.com",
4  | "password": "password123"
5 }
```

Response Success:

```
1 {
2 | "id": 1,
3 | "message": "User berhasil didaftarkan"
4 }
```

Response Error:

```
1 {
2 | "message": "Email sudah terdaftar"
3 }
```

2. Login User

URL: /users/login Method: POST

Description: Login user and get a JWT token.

Request Body:

```
1 {
2 | "email": "john.doe@example.com",
3 | "password": "password123"
4 }
```

```
1 {
2 | "token": "eyJhbGci0iJIUzI1NiIsInR5cCI6IkpXVCJ9..."
3 }
```

```
1 {
2 | "message": "email tidak terdaftar"
3 }
```

3. Get All User

URL: /users Method: GET

Description: Get a list of all users.

Headers:

Headers 8 hidden						
Key	Value	Description	••• Bulk Edit Presets v			
✓ Authorization	Bearer <token></token>					
Key	Value	Description				

Response Success:

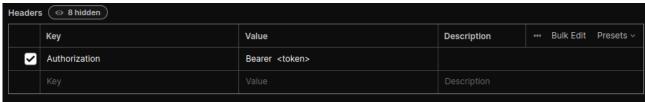
```
"id": 1,
 3
 4
         "name": "John Doe",
 5
         "email": "john.doe@example.com"
 6
       3,
7
         "id": 2,
 8
 9
         "name": "Jane Doe",
10
         "email": "jane.doe@example.com"
11
```

4. Get User by ID

URL: /users/:id Method: GET

Description: Register new users.

Headers:



```
1 {
2  | "id": 1,
3  | "name": "John Doe",
4  | "email": "john.doe@example.com"
5 }
```

```
1 {
2 | "message": "User tidak ditemukan"
3 }
```

5. Update User

URL: /users/:id Method: PUT

Description: Update user data.

Headers:

Headers 8 hidden					
Key	Value	Description	••• Bulk Edit Presets v		
✓ Authorization	Bearer <token></token>				
Key	Value	Description			

Request Body:

```
1  {
2     "name": "John Updated",
3     "email": "john.updated@example.com",
4     "password": "newpassword123"
5  }
```

Response Success:

```
1 {
2 | "message": "User berhasil diupdate"
3 }
```

Response Error:

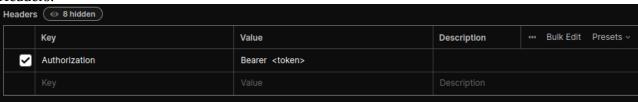
```
1 {
2 | "message": "User tidak ditemukan"
3 }
```

6. Delete User

URL: /users/:id Method: DELETE

Description: Delete users based on ID.

Headers:



Response Success:

```
1 {
2 | "message": "User berhasil dihapus"
3 }
```

Response Error:

```
1 {
2 | "message": "User tidak ditemukan"
3 }
```

Category Endpoints

7. Get All Categories

URL: /category Method: GET

Description: Get a list of all categories.

Response Success:

8. Get Category by ID

URL: /category/:id Method: GET

Description: Get detailed categories based on ID.

Response Success:

Response Error:

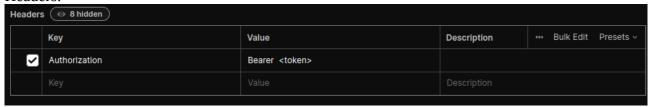
```
1 {
2 | "message": "Kategori tidak ditemukan"
3 }
```

9. Create Category

URL: /category Method: POST

Description: Creating a new category.

Headers:



Request Body:

```
1 {
2 | "name": "Education"
3 }
```

Response Success:

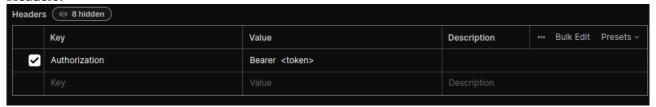
```
1 {
2 | "id": 3,
3 | "message": "Kategori berhasil ditambahkan"
4 }
```

10. Update Category

URL: /category/:id Method: PUT

Description: Update the category based on ID.

Headers:



Request Body:

```
1 {
2  | "name": "Updated Category"
3 }
```

```
1 {
2 | "message": "Kategori berhasil diupdate"
3 }
```

```
1 {
2 | "message": "Kategori tidak ditemukan"
3 }
```

11. Delete Category

URL: /category/:id Method: DELETE

Description: Delete categories based on ID.

Headers:

Headers 8 hidden					
Key	Value	Description	••• Bulk Edit Presets v		
✓ Authorization	Bearer <token></token>				
Key	Value	Description			

Response Success:

```
1 {
2 | "message": "Kategori berhasil dihapus"
3 }
```

Response Error:

```
1 {
2 | "message": "Kategori tidak ditemukan"
3 }
```

Event Endpoints

12. Get All Events

URL: /event Method: GET

Description: Get a list of all events.

```
1
 2
 3
         "id": 1,
         "name": "Tech Conference",
         "description": "Annual tech conference",
 5
         "date": "2023-12-15",
6
         "time": "09:00:00",
7
8
         "location": "Jakarta",
 9
         "category_id": 1,
10
         "user_id": 1
11
```

13. Get Event by ID

URL: /event/:id Method: GET

Description: Get detailed event based on ID.

Response Success:

Response Error:

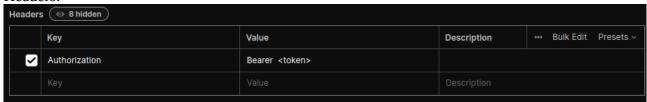
```
1 {
2 | "message": "Event tidak ditemukan"
3 }
```

14. Create Event

URL: /event Method: POST

Description: Make a new event.

Headers:



Request Body:

```
1 {
2  | "id": 2,
3  | "message": "Event berhasil ditambahkan"
4 }
```

15. Update Event

URL: /event/:id Method: PUT

Description: Update the event based on ID.

Headers:

ŀ	Headers						
		Key	Value	Description	••• Bulk Edit Presets v		
	V	Authorization	Bearer <token></token>				
			Value	Description			

Request Body:

Response Success:

```
1 {
2 | "message": "Event berhasil diupdate"
3 }
```

Response Error:

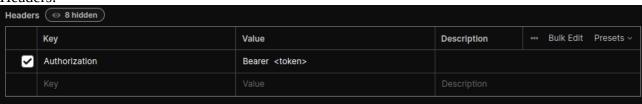
```
1 {
2 | "message": "Event tidak ditemukan"
3 }
```

16. Delete Event

URL: /event/:id Method: DELETE

Description: Delete the event based on ID.

Headers:



Response Success:

```
1 {
2 | "message": "Event berhasil dihapus"
3 }
```

Response Error:

```
1 {
2 | "message": "Event tidak ditemukan"
3 }
```

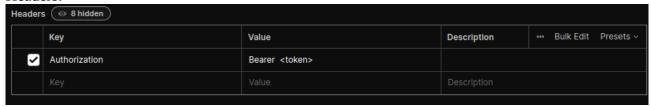
Order Endpoints

17. Get All Orders

URL: /order Method: GET

Description: Get a list of all orders.

Headers:



Response Success:

18. Get Order by ID

URL: /order/:id Method: GET

Description: Get detailed orders based on ID.

Headers:

Н	Headers						
		Key	Value	Description	••• Bulk Edit Presets v		
	V	Authorization	Bearer <token></token>				
		Key	Value	Description			

Response Success:

Response Error:

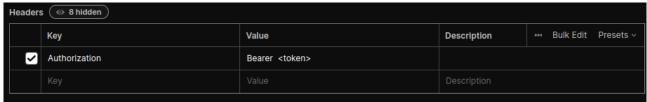
```
1 {
2 | "message": "Order tidak ditemukan"
3 }
```

19. Create Order

URL: /order Method: POST

Description: Make a new order.

Headers:



Request Body:

```
1 {
2 | "id": 2,
3 | "message": "Order berhasil dibuat"
4 }
```

20. Update Order

URL: /order/:id Method: PUT

Description: Update orders based on ID.

Headers:

Н	Headers 8 hidden						
		Key	Value	Description	••• Bulk Edit Presets v		
	V	Authorization	Bearer <token></token>				
			Value	Description			

Request Body:

Response Success:

Response Error:

```
1 {
2 | "message": "Order tidak ditemukan"
3 }
```

21. Delete Order

URL: /order/:id Method: DELETE

Description: Delete orders based on ID.

Headers:



```
1 {
2 | "message": "Order berhasil dihapus"
3 }
```

```
1 {
2 | "message": "Order tidak ditemukan"
3 }
```

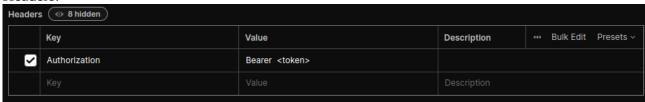
Email Endpoints

22. Send Email

URL: /email Method: POST

Description: Send email.

Headers:



Request Body:

Response Success:

```
1 {
2 | "message": "Email berhasil dikirim"
3 }
```

Response Error:

```
1 {
2 | "message": "Gagal mengirim email"
3 }
```

Location Endpoints

23. Get Location

URL: /location?address=<address>

Method: GET

Description: Get a location based on the address.

Response Success:

Response Error:

```
1 {
2 | "message": "Lokasi tidak ditemukan"
3 }
```

Weather Endpoints

24. Get Weather

URL: /weather?city=<city>

Method: GET

Description: Get weather data based on the city.

Response Success:

```
1  {
2     "city": "Jakarta",
3     "temperature": 30,
4     "humidity": 70,
5     "weather": "Cloudy"
6  }
```

Response Error:

```
1 {
2 | "message": "Kota tidak ditemukan"
3 }
```