

Region and User Management API Documentation

This API provides comprehensive functionality for managing geographical regions and user authentication. It features endpoints for retrieving weather forecasts, creating, reading, updating, and deleting region data, as well as user registration and login with JWT-based authorization.

Endpoint List:

Weather Forecast (GET): Retrieve randomly generated weather data (temperature, summary, date).

Regions Data (GET): Fetch all regions (ID, code, name, image URL).

Create Region (POST): Create a new region with code, name, and image URL.

Get Region by Id (GET): Retrieve details of a specific region by its GUID.

Update an Existing Region (PUT): Update an existing region's code, name, or image URL using its GUID.

Delete an Existing Region (DELETE): Delete a region by its GUID. Returns the deleted region details.

Register a New User (POST): Register a new user with a username, password, and roles.

Existing User Login (POST): Authenticate a user and generate a JWT token for authorization.

Getting Started

Authentication

This API uses JWT (JSON Web Token) authentication for secure access. Users must register and log in to obtain a token, which must be included in the headers of subsequent requests.

Overview

The API provides endpoints for user registration and authentication. Upon successful login, the API returns a JWT token that must be used in the Authorization header for all authenticated requests.

Your token carries many privileges, so be sure to keep it secure! Do not share your token in publicly accessible areas such as GitHub, client-side code, and so forth.

This API allows new users to register by providing a username, password, and roles. Upon successful registration, the API returns a success message.

- **Endpoint:** POST {baseurl}/api/Auth/Register
- **Method:** POST
- **Content-Type:** application/json

Request Body:

Name	Type	Required	Description
username	string	Yes	The email or username of the new user
password	string	Yes	The user's password (must meet security requirements)
roles	array of strings	Yes	List of roles assigned to the user (e.g., admin, reader)

Example Request Body (JSON):

```
{
  "username": "karim@example.com",
  "password": "12345*",
  "roles": [
    "reader"
  ]
}
```

Response:

Success Response (200 OK)

"User Was Registered! Please LogIn"

Endpoint & Environments

Base URL: <https://localhost:7142/>

Get Weather Data:

Overview

This API endpoint provides randomly generated weather data, including temperature values in Celsius and Fahrenheit, a brief weather summary, and the associated date.

Endpoint: [{baseurl}/weatherforecast](#)

Method: GET

Request Details:

Header: application/json

Response Details:

Success Response (200 OK)

```
{
  "date": "2025-02-24",
  "temperatureC": -11,
```

```
"temperatureF": 13,
"summary": "Hot"
},
{
  "date": "2025-02-25",
  "temperatureC": -4,
  "temperatureF": 25,
  "summary": "Cool"
}]
```

Response Fields

Field	Type	Description
date	string	The date for which weather data is generated (YYYY-MM-DD).
temperatureC	int	Temperature in Celsius.
temperatureF	int	Temperature in Fahrenheit.

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }
404	Not Found	{ "error": "Endpoint not found" }
500	Internal Server Error	{ "error": "An unexpected error occurred" }

Get Region Data:

Overview

This API retrieves **comprehensive region data**, structured as an array of objects. Each object contains:

- A **unique Region ID (GUID)**
- An **alphanumeric region code**
- A **descriptive region name**
- The **URL for the associated region image**

Endpoint: [{baseurl}/api/regions](#)

Method: GET

Request Details:

Header: application/json

Response Details:

Success Response (200 OK)

```
[{
  "id": "0ada0027-8694-4cec-d890-08dd50d2769b",
  "code": "BDD",
  "name": "Bangladesh",
  "regionImgUrl": "bangladesh.jpeg"
},
{
  "id": "4220cf6c-e314-4e7e-d891-08dd50d2769b",
  "code": "BDS",
  "name": "badsbda",
  "regionImgUrl": "badasdasd"
}]
```

Response Fields

Field	Type	Description
Id	Guid	Unique identifier for the region.
Code	String	Alphanumeric region code.
Name	String	Name of the region.
regionImgUrl	string	URL of the region's image.

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }
404	Not Found	{ "error": "Endpoint not found" }
500	Internal Server Error	{ "error": "An unexpected error occurred" }

Create Region Data:

Overview

This API allows you to create a new region by submitting a **POST** request with the required region details. The request must include:

- An **alphanumeric region code** (maximum 3 letters)
- A **descriptive region name**
- The **URL for the associated region image**

Endpoint: [{baseurl}/api/regions](#)

Method: POST

Request Details:

Header: application/json

Request Body (Json):

```
{
  "code": "SRL",
  "name": "Srilanka",
  "regionImgUrl": "srilanka.jpeg"
}
```

Response Details:

Success Response (200 OK)

```
[{
  "id": "0ada0027-8694-4cec-d890-08dd50d2769b",
  "code": "BDD",
  "name": "Bangladesh",
  "regionImgUrl": "bangladesh.jpeg"
},
{
  "id": "4220cf6c-e314-4e7e-d891-08dd50d2769b",
  "code": "BDS",
  "name": "badsbda",
  "regionImgUrl": "badasdasd"
}]
```

Response Fields

Field	Type	Description
Id	Guid	Unique identifier for the region.
Code	String	Alphanumeric region code.
Name	String	Name of the region.
regionImgUrl	string	URL of the region's image.

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }

```
{
  "type": "https://tools.ietf.org/html/rfc9110#section-15.5.1",
  "title": "One or more validation errors occurred.",
  "status": 400,
  "errors": {
    "Code": [
      "Code Has To Be Maximum of 3 Letters"
    ]
  },
  "traceId": "00-5acf8c7f9fc0bd76f803fa4e1414b827-28c127a0bfbc1374-00"
}
```

Get Region Data by Id:

Overview

This API retrieves detailed information about a specific region based on its unique identifier (**GUID**). The response includes the **region's ID, alphanumeric code, descriptive name, and associated image URL**.

Endpoint: `{baseurl}/api/regions/{id}`

Method: GET

Request Details:

Header: application/json

Path Parameter:

Name	Type	Required	Description
id	string (GUID)	Yes	The unique identifier of the region

Response Details:

Success Response (200 OK)

```
{
  "id": "0ada0027-8694-4cec-d890-08dd50d2769b",
  "code": "BDD",
  "name": "Bangladesh",
  "regionImgUrl": "bangladesh.jpeg"
}
```

Response Headers

Header	Value
content-type	application/json; charset=utf-8
date	Sun, 23 Feb 2025 11:28:40 GMT
server	Kestrel

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }
404	Not Found	{ "error": "Region not found" }
500	Internal Server Error	{ "error": "An unexpected error occurred" }

Update Region Data:

Overview

This API allows updating an existing region's details using a **PUT** request. The request requires a **region ID (GUID)** and includes updated values for **region code, name, and image URL**.

Endpoint: `{baseurl}/api/regions/{id}`

Method: PUT

Request Details:

Header: application/json

Path Parameter:

Name	Type	Required	Description
id	string (GUID)	Yes	The unique identifier of the region

Request Body:

```
{
  "code": "UAE",
  "name": "UnitedArabEmirates",
  "regionImgUrl": "uae.jpeg"
}
```

Response Details:

Success Response (200 OK)

```
{
  "id": "0ada0027-8694-4cec-d890-08dd50d2769b",
  "code": "UAE",
  "name": "UnitedArabEmirates",
  "regionImgUrl": "uae.jpeg"
}
```

Response Headers

Header	Value
content-type	application/json; charset=utf-8
date	Sun, 23 Feb 2025 11:28:40 GMT
server	Kestrel

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }
404	Not Found	{ "error": "Region not found" }
422	Unprocessable Entity	{ "error": "Validation failed: Code must be exactly 3 letters" }
500	Internal Server Error	{ "error": "An unexpected error occurred" }

Delete Region Data:

Overview

This API allows deleting an existing region using a **DELETE** request. The request requires a **region ID (GUID)** as a path parameter. Upon successful deletion, the API returns the deleted region details.

Endpoint: `{baseurl}/api/regions/{id}`

Method: DELETE

Request Details:

Header: application/json

Path Parameter:

Name	Type	Required	Description
id	string (GUID)	Yes	The unique identifier of the region

Response Details:

Success Response (200 OK)

```
{
  "id": "6884f7d7-ad1f-4101-8df3-7a6fa7387d81",
  "code": "NTL",
  "name": "Northland",
  "regionImgUrl": null
}
```

Response Headers

Header	Value
content-type	application/json; charset=utf-8
date	Sun, 23 Feb 2025 11:28:40 GMT
server	Kestrel

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }
404	Not Found	{ "error": "Region not found" }
500	Internal Server Error	{ "error": "An unexpected error occurred" }

User Registration:

Overview

This API allows new users to register by providing a **username**, **password**, and **roles**. Upon successful registration, the API returns a success message.

Endpoint: [{baseurl}/Auth/Register](#)

Method: POST

Request Details:

Header: application/json

Request Body:

Name	Type	Required	Description
username	string	Yes	The email or username of the new user
password	string	Yes	The user's password (must meet security requirements)
roles	array of strings	Yes	List of roles assigned to the user (e.g., admin, reader)

Request Body (Json Format):

```
{
  "username": "karim@example.com",
  "password": "12345*",
  "roles": [
    "reader"
  ]
}
```

Response Details:

User Was Registered! Please Login

Success Response (200 OK)

```
{
  "username": "karim@example.com",
  "password": "12345*",
}
```

Response Headers

Header	Value
content-type	application/json; charset=utf-8
date	Sun, 23 Feb 2025 11:28:40 GMT
server	Kestrel

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }
409	Conflict	{ "error": "User already exists" }
500	Internal Server Error	{ "error": "An unexpected error occurred" }

User Login:

Overview

This API allows registered users to log in by providing their **username** and **password**. Upon successful authentication, a **JWT token** is returned, which can be used for authorization in subsequent requests.

Endpoint: [{baseurl}/Auth/Login](#)

Method: POST

Request Details:

Header: application/json

Request Body:

Name	Type	Required	Description
username	string	Yes	The email or username of the user
password	string	Yes	The user's password

Request Body (Json Format):

```
{  
  "username": "karim@example.com",  
  "password": "12345*"  
}
```

Response Details:

User Was Registered! Please Login

Success Response (200 OK)

```
{
  "jwtToken":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJodHRwOi8vc2NoZW1hcy54bWxzb2FwLm9yZy93cy8yMDA1LzA1L2lkZW50aXR5L2NsYWlscy9lbWVpbGFkZHI3Mi0iOiJrYXJpbUBleGFtcGxlLmNvbSIsImh0dHA6Ly9zY2hlbWFzLm1pY3Jvc29mdC5jb20vd3MvMjAwO C8wNi9pZGVudGl0eS9jbGFpbXMvcm9sZSI6IlJlYWRlciIsImV4cCI6MTc0MDMxNDk2MywiaXNzIjoiaHR0cHM6Ly9sb2NhbGhvc3Q6NzE0Mi8iLCJhdWQiOiJodHRwczovL2xvY2FsaG9zdDo3MTQyLyJ9.h3AbiLOm6UrNPNRG3_QsuCmvyRMLyrrSWy-n-BfeOrs"
}
```

Response Headers

Header	Value
content-type	application/json; charset=utf-8
date	Sun, 23 Feb 2025 11:28:40 GMT
server	Kestrel

Possible Error Responses

Status Code	Meaning	Example Response
400	Bad Request	{ "error": "Invalid request format" }
401	Unauthorized	{ "error": "Authentication failed" }
500	Internal Server Error	{ "error": "An unexpected error occurred" }