# Blocked Countries and IP Validation API

## Project Overview

This project is a .NET Core API for managing blocked countries and validating IP addresses using third-party geolocation APIs (IPGeolocation.io). The application uses in-memory data storage instead of a database.

### Technologies Used

- .NET Core
- HttpClient
- Swagger
- ConcurrentDictionary
- Newtonsoft.Json

## Setup Instructions

### Prerequisites

- .NET SDK
- Visual Studio

### Installation Steps

1. **Clone the repository**:

   ```
   git clone https://github.com/3mosakr/ATechnologiesTask
   ```

2. **Navigate to the project directory**:

   ```
   cd your-repo
   ```

3. **Install dependencies**:

   ```
   dotnet restore
   ```

4. **Run the project**:

   ```
   dotnet run
   ```

### Configuration

- Set up API keys in appsettings.json:

  ```
   {
    "ApiSettings": {
          "IpGeolocationApiKey": "your-api-key"
    }
   }
  ```

## API Documentation

# Endpoints

## Block Country

- **Endpoint:** POST /api/countries/block
- **Description**: Blocks a country permanently.
- **Request**:

  {

    "US"

  }

- **Response**:

```
{
  "statusCode": 200,
  "status": true,
  "message": "Country: 'United States' is blocked successfully.",
  "data": [
    {
      "countryCode": "US",
      "countryName": "United States"
    }
  ],
  "errors": null
}
```

### Unblock Country

- **Endpoint**: DELETE /api/countries/block/{countryCode}
- **Description**: Unblocks a country.
- **Response**:

```
{
  "statusCode": 200,
  "status": true,
  "message": "Country: 'United States' is unblocked.",
  "data": null,
  "errors": null
}
```

### Get Blocked Countries

- **Endpoint**: GET /api/countries/blocked
- **Description**: Gets a paginated list of blocked countries with optional search.
- **Request**:

  GET /api/countries/blocked?page=1&pageSize=10&search=US

- **Response**:

```
{
  "currentPage": 1,
  "totalPages": 1,
  "totalCount": 2,
  "pageSize": 10,
  "hasPreviousPage": false,
  "hasNextPage": false,
  "statusCode": 200,
  "status": true,
  "message": "Countries with search: 'us'",
  "data": [
    {
      "countryCode": "RU",
      "countryName": "Russia"
    },
    {
      "countryCode": "US",
      "countryName": "United States"
    }
  ],
  "errors": null
}
```

### Temporarily Block Country

- **Endpoint**: POST /api/countries/temporal-block
- **Description**: Temporarily blocks a country for a specified duration.
- **Request**:

```
{
  "countryCode": "US",
  "durationMinutes": 120
}
```

- **Response**:

```
 {
  "statusCode": 200,
  "status": true,
  "message": "Country: 'United States' is temporarily blocked successfully for 120 Minutes.",
  "data": null,
  "errors": null
 }
```

### IP Lookup

- **Endpoint**: GET /api/ip/lookup
- **Description**: Looks up the location data for a given IP address.
- **Request**:

  GET /api/ip/lookup?ipAddress=8.8.8.8

- **Response**:

```
 {
  "statusCode": 200,
  "status": true,
  "message": "Get data successfully for Ip: 8.8.8.8",
  "data": [
    {
      "ip": "8.8.8.8",
      "country_Code2": "US",
      "country_code3": "USA",
      "country_Name": "United States",
      "state_prov": "California",
      "state_code": "US-CA",
      "district": "Santa Clara",
      "city": "Mountain View",
      "isp": "Google LLC"
    }
  ],
  "errors": null
}
```

### Check If IP is Blocked

- **Endpoint**: GET /api/ip/check-block
- **Description**: Checks if the caller's IP address is blocked.
- **Response**:

```
{
  "statusCode": 400,
  "status": false,
  "message": "You Can't Access this site from your country right now.",
  "data": null,
  "errors": null
}
```

### Get Blocked Attempt Logs

- **Endpoint**: GET /api/logs/blocked-attempts
- **Description**: Retrieves a paginated list of blocked attempt logs.
- **Request**:

  GET /api/logs/blocked-attempts?page=1&pageSize=10

- **Response**:

```
{
  "currentPage": 1,
  "totalPages": 1,
  "totalCount": 1,
  "pageSize": 10,
  "hasPreviousPage": false,
  "hasNextPage": false,
  "statusCode": 200,
  "status": true,
  "message": "Get Blocked Attempt Logs",
  "data": [
    {
      "ipAddress": "8.8.8.8",
      "timestamp": "2025-03-01T20:58:45.1175606Z",
      "countryCode": "US",
      "blockedStatus": true,
      "userAgent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0 Safari/537.36"
    }
  ],
  "errors": null
}
```

## Error Handling

- **400 Bad Request**: Invalid input data.
- **404 Not Found**: Resource not found.
- **409 Conflict**: Operation done before.
- **500 Internal Server Error**: An unexpected error occurred.

## Features

#### Blocked Countries Management

- **Add Blocked Country**: Permanently block a country by adding its country code.
- **Remove Blocked Country**: Unblock a country by removing its country code.
- **List Blocked Countries**: Retrieve a paginated list of blocked countries with optional search functionality.

#### IP Lookup

- **Fetch Location Data**: Retrieve location details for a given IP address, including country, state, city, and ISP.

#### Temporary Blocking

- **Temporarily Block Country**: Block a country for a specified duration (e.g., 2 hours). Automatically unblock after expiration using a background service.

#### Logging

- **Log Blocked Attempts**: Log attempts to access the API from blocked countries.
- **Retrieve Logs**: Retrieve a paginated list of blocked attempt logs.

## Background Services

- **Remove Expired Temporary Blocks**: A background service that runs every 5 minutes to remove expired temporary blocks from the in-memory store.

## Contribution Guidelines

### How to Contribute

1. **Fork the repository**: Create a fork of the repository on GitHub.
2. **Create a new branch**: Create a new branch for your feature or bugfix.

   ```
   git checkout -b feature/your-feature-name
   ```

3. **Make your changes**: Implement your changes and commit them with a descriptive message.

   ```
   git commit -m "Add feature: your feature description"
   ```

4. **Push to your branch**: Push your changes to your forked repository.

   ```
   git push origin feature/your-feature-name
   ```

5. **Submit a pull request**: Create a pull request to merge your changes into the main repository.

### License

This project is licensed under the MIT License.