Ease.io Back End Code Challenge

Goal

The purpose of this code challenge is to learn how you approach and solve problems. We will go over your solution during the technical interview.

Task

Using .NET Core and C#, write both a Web API and a client for that API (console app) to maintain a database of GUIDs and associated metadata (user and expiration date).

Web API

Create

Creates a new GUID and stores it in the database, along with provided metadata.

1. Generates a GUID if one is not provided.
2. Expires all GUIDs within 30 days if an expiration date is not provided.
3. GUIDs are formatted as 32 hexadecimal characters, all uppercase.

Example 1

```
URL: POST /guid/43B56A5297824C08AC2C04C7E1EAD36C

Input:
{
    "expires": "2024-10-01T14:18:00.000Z",
    "user": "John Doe"
}
Output:
{
    "guid": "43B56A5297824C08AC2C04C7E1EAD36C",
    "expires": "2024-10-01T14:18:00.000Z",
    "user": "John Doe"
}
```

Example 2

```
URL: POST /guid

Input:
```

```
{
    "user": "John Doe"
}
Output:
{
    "guid": "43B56A5297824C08AC2C04C7E1EAD36C",
    "expires": "2024-10-01T14:18:00.000Z",
    "user": "John Doe"
}
```

Read

Returns metadata for valid GUIDs. GUIDs are only valid until they expire.

Example

```
URL: GET /guid/43B56A5297824C08AC2C04C7E1EAD36C

Output:
{
    "guid": "43B56A5297824C08AC2C04C7E1EAD36C",
    "expires": "2024-10-01T14:18:00.000Z",
    "user": "John Doe"
}
```

Update

Updates GUID metadata. The GUID itself cannot be updated.

Example

```
URL: PUT /guid/43B56A5297824C08AC2C04C7E1EAD36C

Input:
{
    "expires": "2024-10-02T20:20:00.000Z"
}
Output:
{
    "guid": "43B56A5297824C08AC2C04C7E1EAD36C",
    "expires": "2024-10-02T20:20:00.000Z",
    "user": "John Doe"
}
```

Delete

Deletes a GUID and its metadata.

```
URL: DELETE /guid/43B56A5297824C08AC2C04C7E1EAD36C
```

No output.

Error Codes

All APIs return:

1. 200's on accepted/successful requests.
2. 400's on client errors.
3. 500's on server errors.

Client app should handle errors gracefully and provide meaningful feedback to the user.

Suggestions

1. Use Redis for caching.
2. Provide tests.