Traka Coding Challenge

Thank you for taking our coding challenge. We would like you to write an API controller for a very fictitious system. Don't worry about completing all of the tasks - we really want to see how you approach coding and how you solve problems.

A .NET 5 solution has been provided with a number of integration tests already created. It also has Swagger enabled to test the API from within your browser.

Either load the solution in Visual Studio, or use the dotnet command line.

Repositories are provided for you that return some example data. These are `UserRepository` and `SystemRepository` and implement the `IRepository<T>` interface:

```
public interface IRepository<T> where T:Entity
{
    Task<IEnumerable<T>> GetAll();

    Task<T> GetByIdAsync(int id);

    Task Add(T value);

    Task Update(int id, T value);

    Task Delete(int id);
}
```

In the database are some `User` objects and some `TrakaSystem` objects. Each system represents a key cabinet with a number of iFobs represented by `Fob`.

Scope of work

You must implement the following endpoints in `UserController`. You can make use of the integration test project, the Swagger UI or any other API testing tool to ensure the correct data is returned in JSON format. The first method has been created already.

Method	Path	Request Body	Expected Result
GET	/user		A list of all users in the database
GET	/user/{id}		A single user record or 404 if user does not exist
POST	/user	JSON user	Adds the user to the database

Method	Path	Request Body	Expected Result
PUT	/user/{id}	JSON encoded user	Updates the user with the given ID with the name from the supplied user
DELETE	/user/{id}		Removes the given user from the database
GET	/user/{id}/fobs		Should return the two `Fob` records currently held by the given user (serial numbers "A6E77B080000" and "BC2433AF8299")

When you have finished, commit your changes to the git repository.

Bonus Task

In the `SystemRepository`, write a method to assign a fob to a given user. This should take in a system ID, fob position and a user ID and set the `CurrentUser` on the fob record. It must throw an exception if the fob does not have a `FobStatus` of `InSystem`.

Add an endpoint to the API to invoke this method (e.g. /system/{systemId}/assignfob/{position}/{userId})

Sample Data

Users

```
[
  {
    "username": "Sneezy",
    "id": 1
  },
  {
    "username": "Sleepy",
    "id": 2
  },
  {
    "username": "Grumpy",
    "id": 3
  }
]
```

Systems

```
[
  {
    "name": "System 1",
    "fobs": [
      {
        "position": 1,
          "serialNumber": "5D7C7C080000",
          "currentStatus": 1,
          "currentUser": null
        },
        {
          "position": 2,
          "serialNumber": "A6E77B080000",
          "currentStatus": 2,
          "currentUser": 1
        },
        {
          "position": 3,
          "serialNumber": "167E7C080000",
          "currentStatus": 2,
          "currentUser": 2
        },
        {
          "position": 4,
          "serialNumber": "F4422FFB4448",
          "currentStatus": 1,
          "currentUser": null
        },
        {
          "position": 5,
          "serialNumber": "243D7C080000",
          "currentStatus": 1,
          "currentUser": null
        }
      ],
      "id": 1
    },
    {
      "name": "System 2",
      "fobs": [
        {
          "position": 1,
          "serialNumber": "A5CE14F9FDD4",
          "currentStatus": 1,
          "currentUser": null
        },
        {
          "position": 2,
          "serialNumber": "F9537C080000",
          "currentStatus": 1,
          "currentUser": null
        },
        {
          "position": 3,
          "serialNumber": "7AAFF3EF301F",
          "currentStatus": 3,
          "currentUser": null
        },
        {
          "position": 4,
          "serialNumber": "1A4E265F130A",
          "currentStatus": 1,
          "currentUser": null
        },
        {
          "position": 5,
          "serialNumber": "BC2433AF8299",
          "currentStatus": 2,
          "currentUser": 1
        }
      ],
      "id": 2
    }
  ]
```