Controlleurs et Endpoints

Controllers are the classes that handle HTTP Requests by holding method that correspond to API endpoints. First, create a *Controllers* folder to store you controllers. To create a controller, create a class with the following naming convention : the name of the resource, starting with a uppercase letter, then *Controller*.

Then, the class must extend the `ControllerBase` class and be annotated with the `[ApiController]` and `[Route]` annotation :

- `[ApiController]` enables automatic model validation, we'ill come to that later
- `[Route]` maps the controllers Route, takes a string parameter to map the route

Example :

```
namespace TodoAPI.Controllers {
  [ApiController]
  [Route("api/[controller]")]
  public class TodosController : ControllerBase {
  }


}
```

This controller handles request that start with : *https://mytodo.io/api/Todos*

To handle requests, a controller must have API endpoints method. Those method handle HTTP requests and return HTTP responses.

To be an enpoint method, a method must be annotated with a HTTP Verb annotation have a IActionResult return type. The HTTP Verb annotation take an optionanl parameter for the route mapping. If this argument is not provided, the method with handle the HTTP verb for the Controller's route.

Let's say we have a model object like that in our Models folder :

```
namespace TodoAPI.Models {
  public class Todo {
    public string Name {get;set;}
    public string Description {get;set;]
  }
```

```
}
```

Verb Mapping

You can just return the response object Within the `Ok()` method that will take care of returning the right HTTP status code and serialize the object to JSON.

Example :

```
[HttpGet]
public IActionResult GetTodos() => Ok( new List<Todo>(){new Todo(){Name = "Hello",
Description  = "World"}});
```

This method will handle GET requests on the endpoint */api/Todos*

Body Parameters

You can get a parameter from the request body :

```
[HttpPost]
public IActionResult CreateTodo([FromBody] Todo todo){


}
```

Query Parameter

This method will handle POST requests on the endpoint */api/Todos* and have a Todo JSON representation as body.

You can also retrieve a GET query parameter :

```
[HttpGet]
public IActionResult SearchTodos([FromQuery("name")] string name){
□
}
```

Path Parameters

You can also retrieve a path parameter :

```
[HttpGet("{todoId:Guid}")]
public IActionResult GetTodo(Guid todoId){


}
```

Revision #6
Created 2020-08-23 12:52:53 UTC by Arsène Lapostolet
Updated 2022-09-28 17:38:15 UTC by Arsène Lapostolet