NET Core Error Responses

Here’s a decent mechanism for returning useful errors from a WEB API.

Aside: We have a standing test API (OGA.RESTAPI_Testing.Service) that will return examples of this, here: http://192.168.1.201:4250/swagger/index.html

We can compose an instance of the ProblemDetail class by doing this in an action method:

```
[HttpGet("Forbidden/withProblemDetail")]
public async Task<IActionResult> ForbiddenwithProblemDetail()
{
    // Create Problem Detail response, as an api-friendly error response for the caller...
    // See this article: <https://learn.microsoft.com/en-us/aspnet/core/web-api/handle-errors?view=aspnetcore-5.0#pd>
    return Problem(
        title: "Resource Access Not Permitted.",
        detail: $"Resource Access Not Permitted.",
        statusCode: StatusCodes.Status403Forbidden
    );
}
```

For cases where we want an Action’s Exception handler to return a ProblemDetail, we can do this:

```
[HttpGet("BadRequest/ProblemDetail")]
public async Task<IActionResult> BadRequestwithProblemDetail()
{
    try
    {
        throw new BusinessRuleBrokenException("Some business rules exception occurred.");
    }
    catch (OGA.SharedKernel.Exceptions.BusinessRuleBrokenException bre)
    {
        // Create Problem Detail response, as an api-friendly error response for the caller...
        // See this article: <https://learn.microsoft.com/en-us/aspnet/core/web-api/handle-errors?view=aspnetcore-5.0#pd>
        return Problem(
            title: "Business Rules Exception",
```

```
            detail: bre.Message,
            statusCode: StatusCodes.Status404NotFound
        );
    }
    catch(Exception ex)
    {
        return BadRequest(ex.Message);
    }
}
```

Revision #1
Created 10 April 2025 05:19:16 by glwhite
Updated 10 April 2025 05:20:09 by glwhite