.NET ACTION FILTERS

Part 1

@Philip Kayode, PHILSKAY

Action filter

```csharp
[AttributeUsage(AttributeTargets.Class | AttributeTargets.Method)]
public class ModelStateCheck : Attribute, IAsyncActionFilter
{
    public async Task OnActionExecutionAsync(ActionExecutingContext context,
ActionExecutionDelegate next)
    {
        if (!context.ModelState.IsValid)
        {
            List<string> errorMessage = ListModelErrors(context);

            HttpContext httpContext = context.HttpContext;

            httpContext.Response.StatusCode = (int)HttpStatusCode.OK;
            // return model errors response
            await httpContext.Response.WriteAsJsonAsync(new ResponseMessage<string>()
            {
                status = false,
                result = errorMessage,
                status_code = (int)HttpStatusCode.NotImplemented,
                error_message = "Invalid validation"
            });
            return;
        }
        await next();
    }


 private List<string> ListModelErrors(ActionContext context) => context.ModelState
                                        .SelectMany(x => x.Value.Errors)
                                        .Select(x => x.ErrorMessage).ToList();

}
```

Action filter attribute

@Philip Kayode, PHILSKAY

Set action filter attribute on controller class or action

The filter is automatically called to check the model state validations before executing the actions

```
//Action filter for model validation check
[ModelStateCheck]
public class AuthController : ControllerBase
{

}
```

To globally set your filter, you can create a base controller class that inherit from controllerBase class. Inside your custom base class, the filter attribute can be assigned. Then you use your base class across all controllers instead of ControllerBase or you do the next slide.

@Philip Kayode, PHILSKAY

Set action filter globally in the startup.cs

The filter is automatically called to check the model state validations before executing the actions

```
services.AddControllers(options =>
{
    options.Filters.Add<ModelStateCheck>();
});
```

@Philip Kayode, PHILSKAY