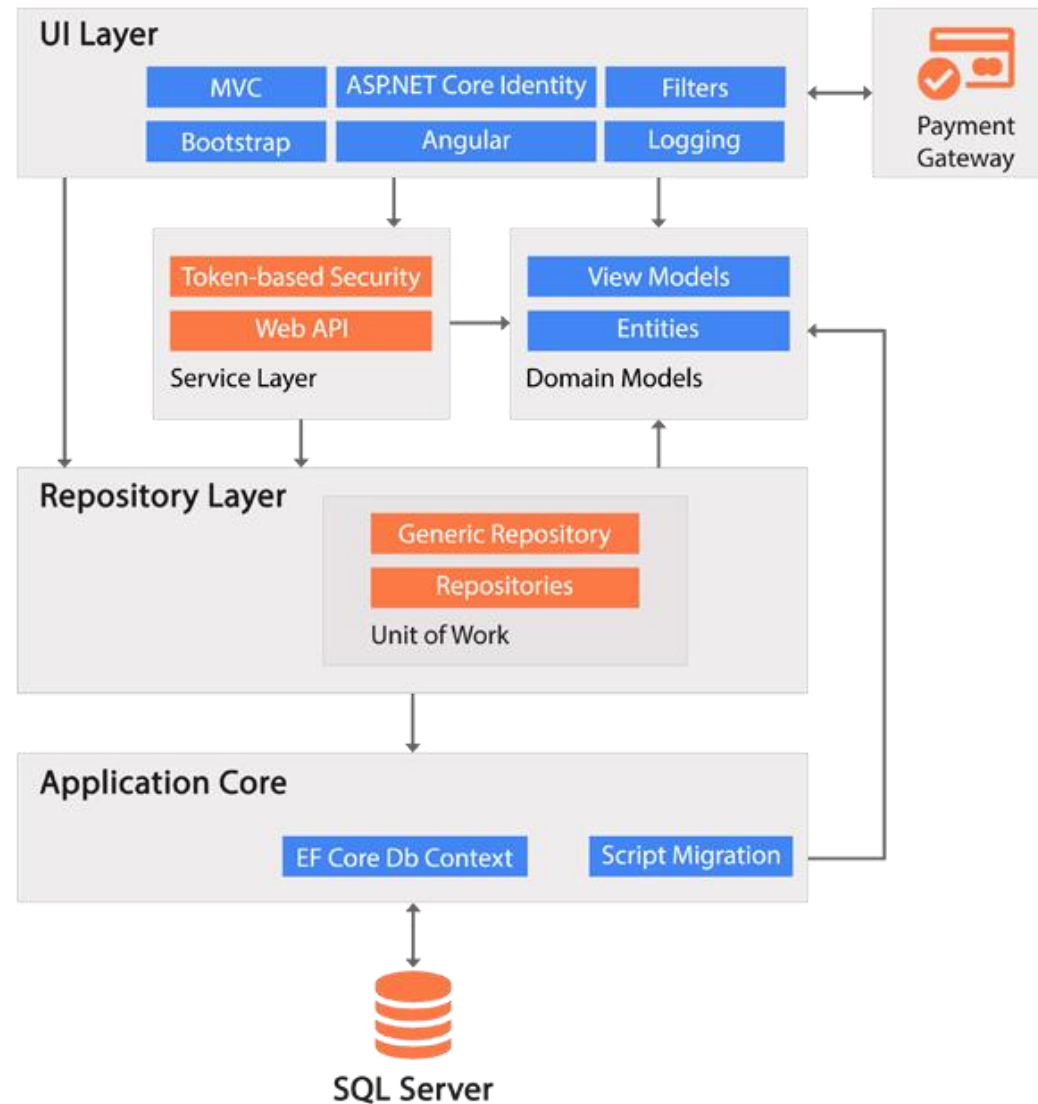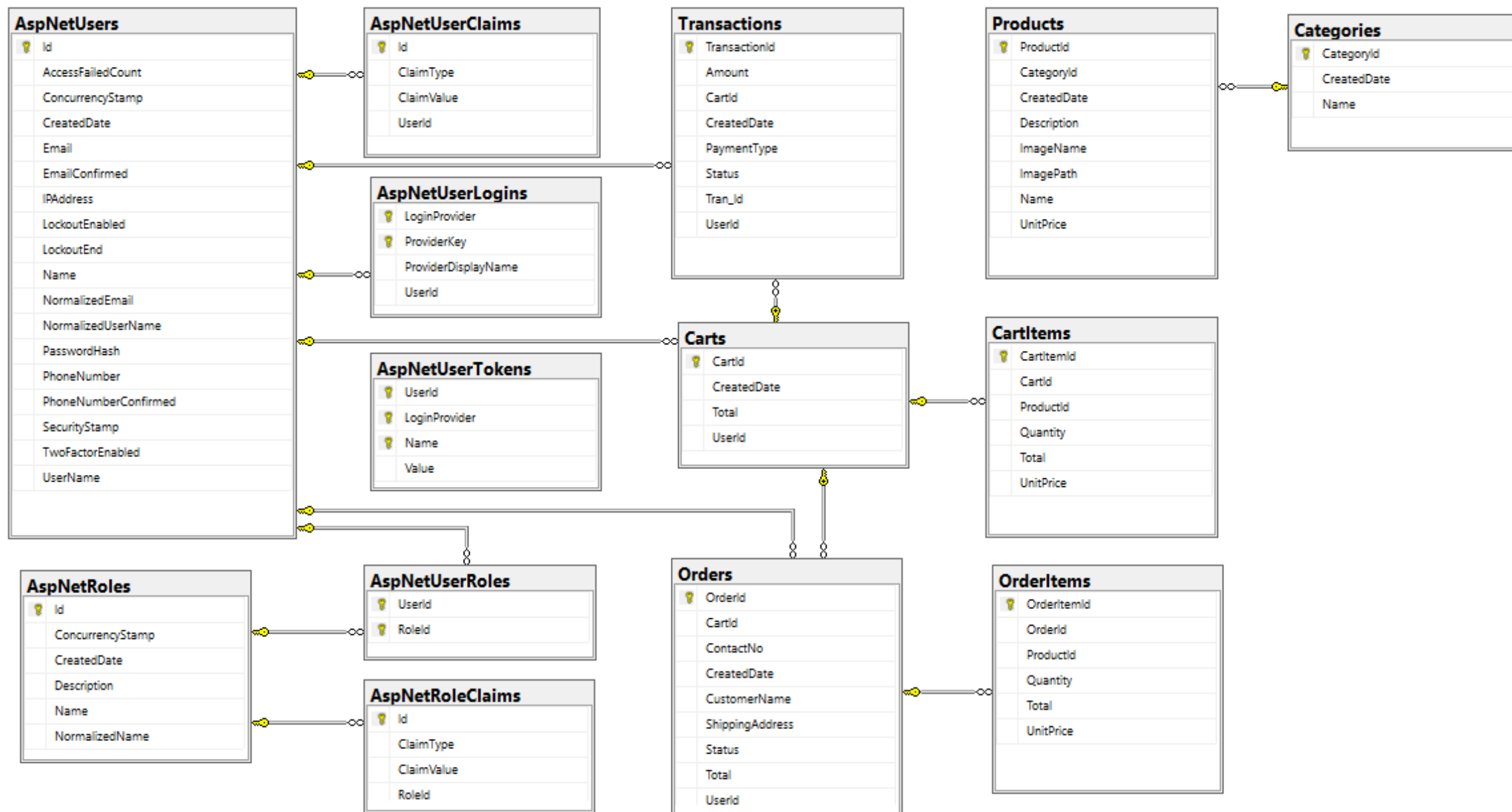# ASP.NET Core With Angular Project

Agenda

- Project Architecture

- Database Schema

- ASP.NET Identity

- ASP.NET Identity Authentication

- ASP.NET MVC Core Request Pipeline

- Middleware

- ASP.NET Core Environments

DotNetTricks

# Project Architecture

Database Schema

ASP.NET Identity

- **One ASP.NET Identity** - One system to be used with all ASP.NET frameworks - ASP.NET MVC, Web Forms, Web Pages, Web API, and SignalR

- **Persistence Control** - You have complete control over the schema of user profile information. It uses Entity Framework Code First to implement all of its persistence mechanism

- **Role Provider** - Roles are stored in a separate database table and you can create, edit and delete the roles.

- **Claims Based -** Supports claims-based authentication, where the user's identity is represented as a set of claims. Claims allow you to add lots of more information about the user than the simple user name-password.

ASP.NET Identity

- **Social Login Providers** - You can easily add social log-ins such as Microsoft Account, Facebook, Twitter, Google etc.

- **OWIN Integration -** It is based on OWIN middleware that can be used on any OWIN-based host. It does not have any dependency on System.Web.

- **NuGet package -** Available as Nuget package which can be installed in the ASP.NET MVC, Web Forms and Web API

⏵DotNetTricks

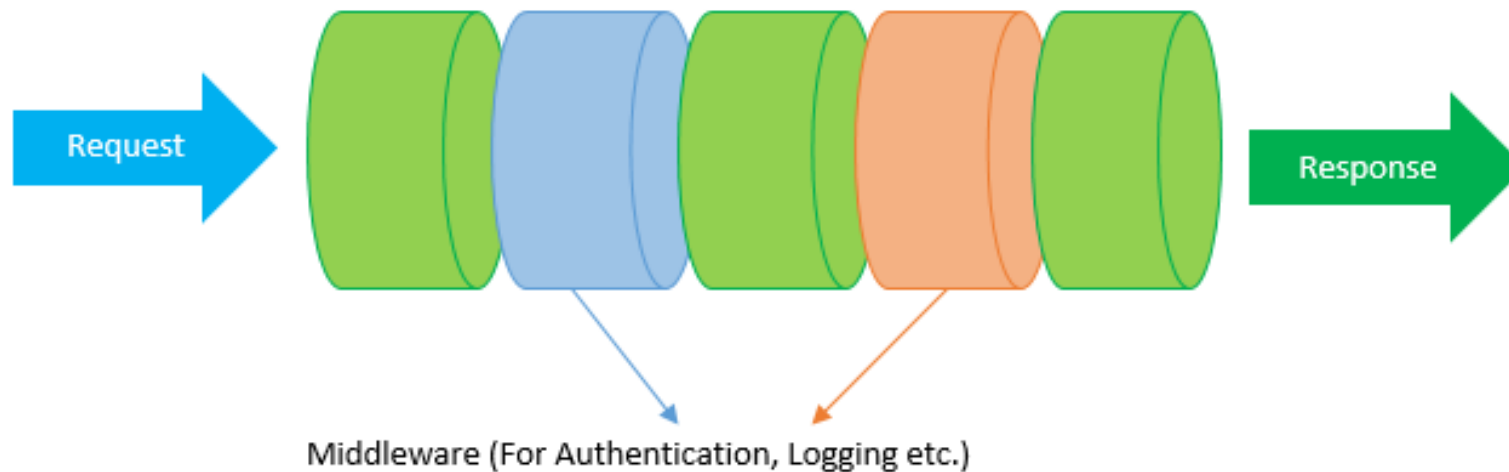# ASP.NET Identity Authentication
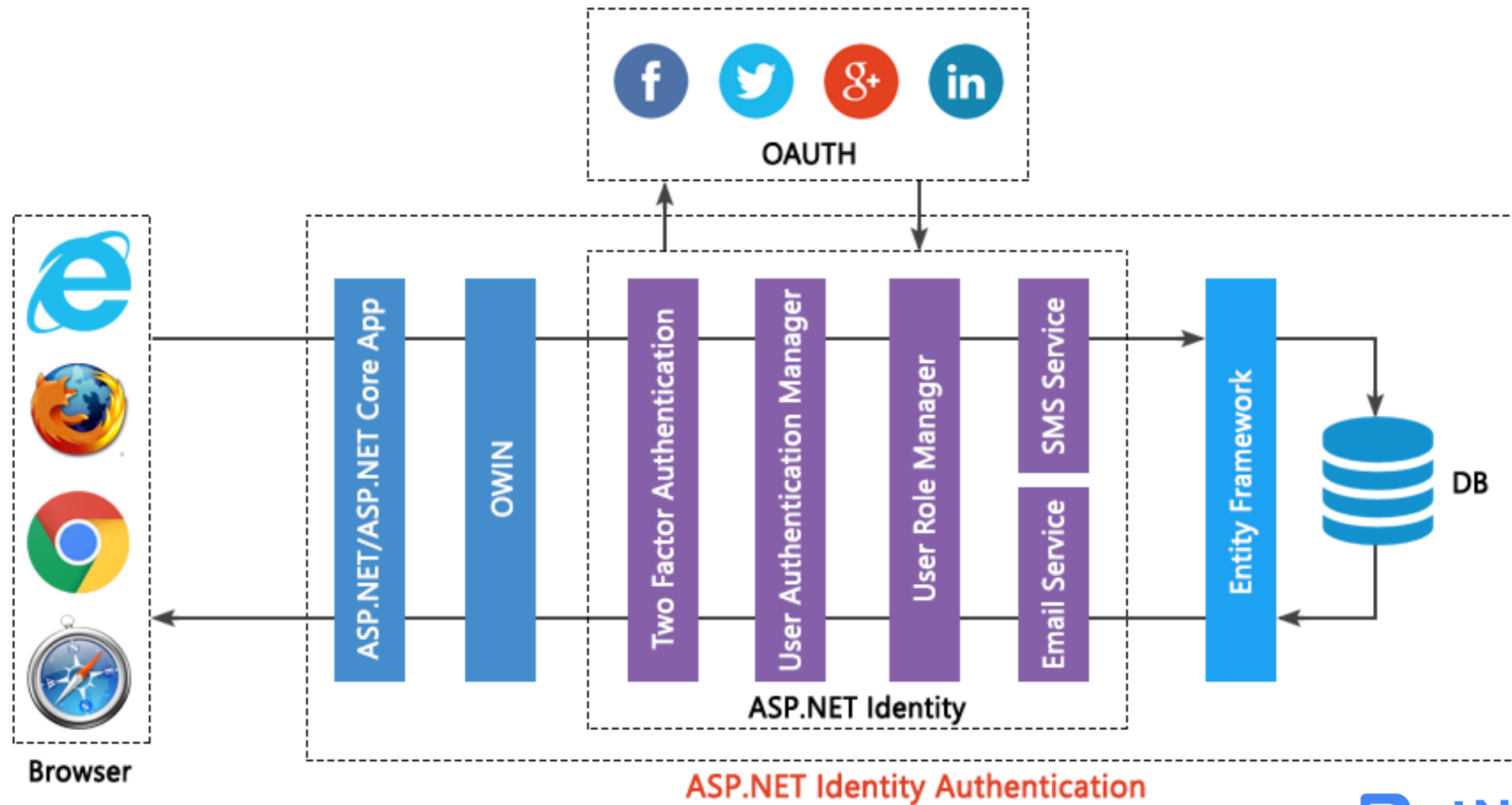
ASP.NET Core Request Processing
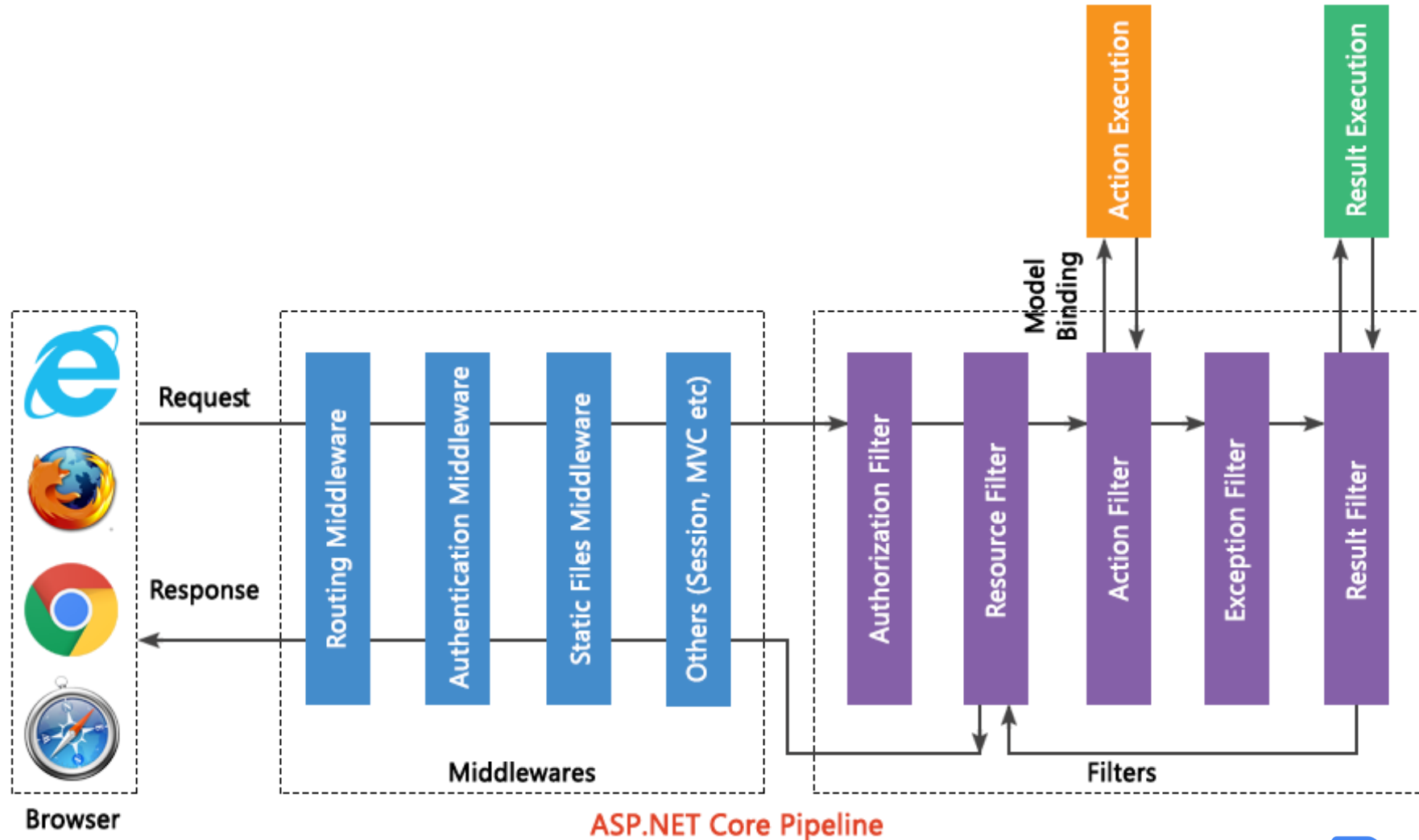
Middleware

- A middleware is a chain of functions which are executed before the request is being processed.
- A middleware can be used for user authentication, logging etc.

Middleware (For Authentication, Logging etc.)

Existing Middleware

- ## Static files

```
app.UseStaticFiles();
```

- ## Routing

```
app.UseRouter(routes);
```

- ## Error Handling

```
app.UseExceptionHandler("/Home/Error");
```

- ## MVC

```
app.UseMvc(routes => {
  routes.MapRoute(
            name: "default",
            template: "{controller=Home}/{action=Index}/{id?}");
});
```

DotNet**Tricks**

ASP.NET Core Environments

ConfigureDevelopment()
ConfigureServicesDevelopment()

ConfigureStaging()
ConfigureServicesStaging()

ConfigureProduction()
ConfigureServicesProduction()

DotNetTricks