**The ASP.NET Life Cycles**

1. **Application life cycle**
2. **Request life cycle**

Application life cycle tracks the life of a web application from the moment it starts to the moment it is terminated. It allow you to perform actions when the application

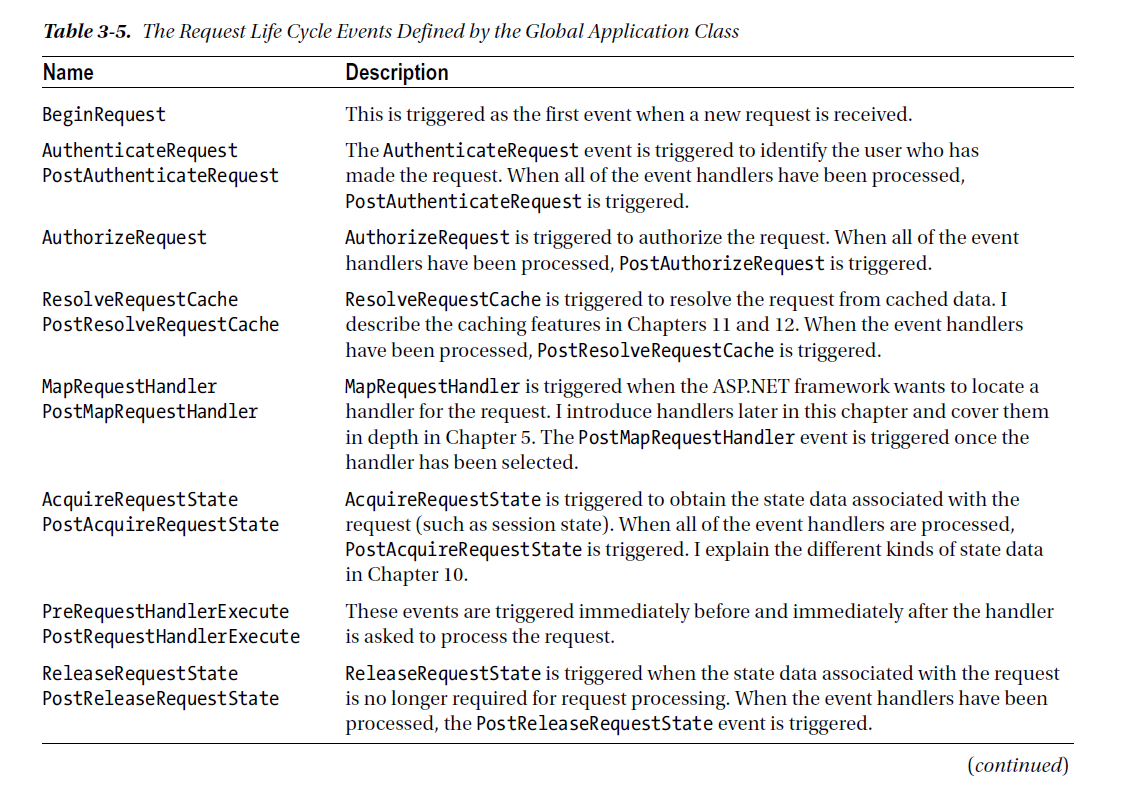
starts and when it is shut down in a controlled way. Divided into

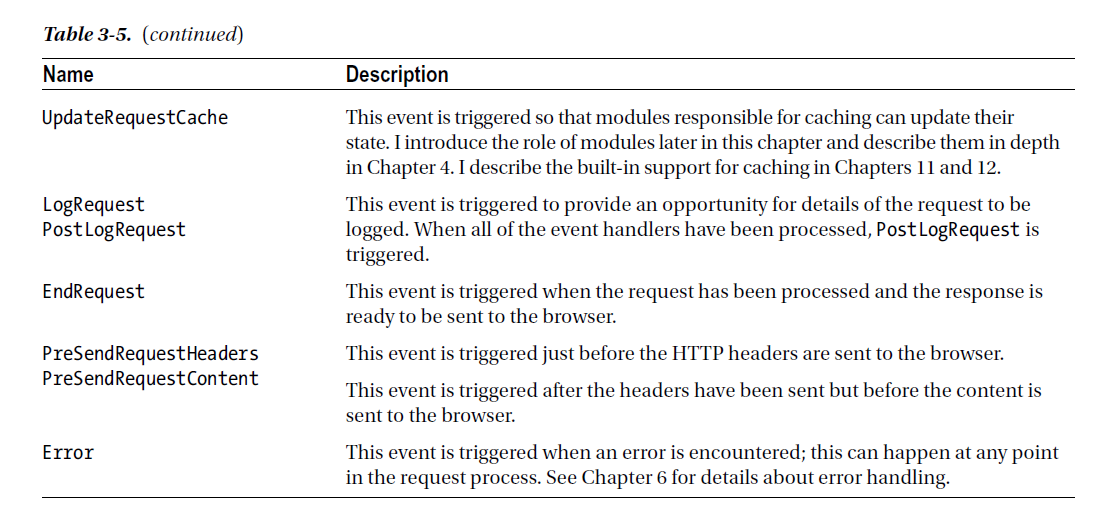
* Application\_Start()
* Application\_End()

Request life cycle defines the path that an HTTP request follows as it moves through the ASP.NET platform from the point at which the initial request is received until the response is sent.



**Request Lifecycle In Detail**





**Detailed view of Request Lifecycle**

****

The ASP.NET Context Objects

ASP.NET provides a set of objects that are used to provide context information about the current request, the response that will be returned to the client, and the web application itself.

The class at the heart of the context is System.Web.HttpContext. It is universally available throughout the ASP.NET framework and the MVC framework, and it acts as a gateway to other context objects and to ASP.NET platform features and services, like HttpContext.Current.Request, HttpContext.Current.Response, HttpContext.Current.Session etc.

ASP.NET Modules

Modules are classes that handle life-cycle events to monitor or manipulate requests or responses.

**Different types of Module:-**

1. Create a class that implements IHttpModule. Methods to implement are,

* Init(app)
* Dispose()

After we provide implementation, we need to register this module in web.config under <system.webServer> / <modules>

1. **Creating Self-registering Modules**

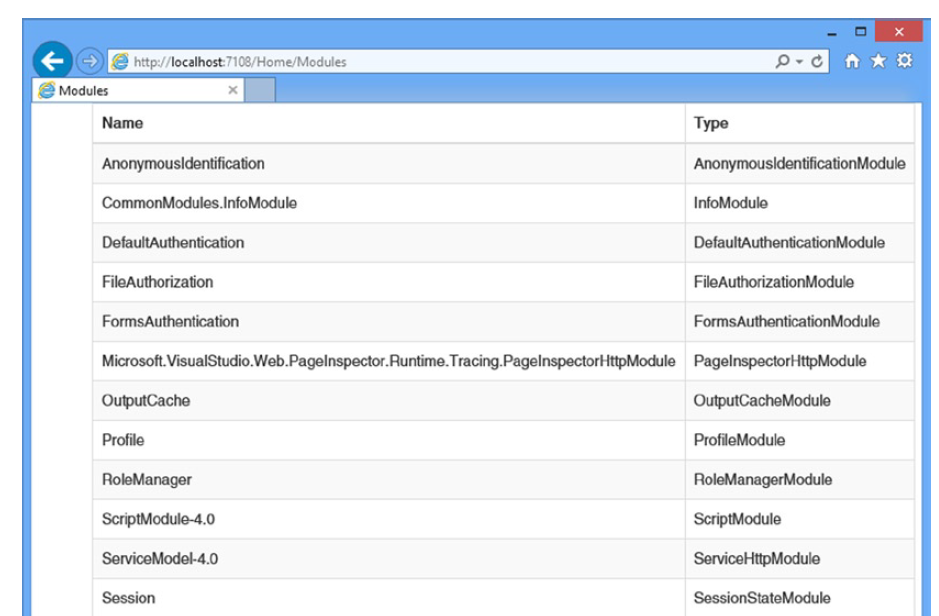
Define them in a separate project from the rest of the web application so that the output

from the project can be used multiple times. So addition to web.config is not needed.

1. **Module Events**

To share data between modules. Define an event in one module, and raise it with the data needs to be shared. Then all the modules with subscribed events can get the data.

1. **Built in modules**



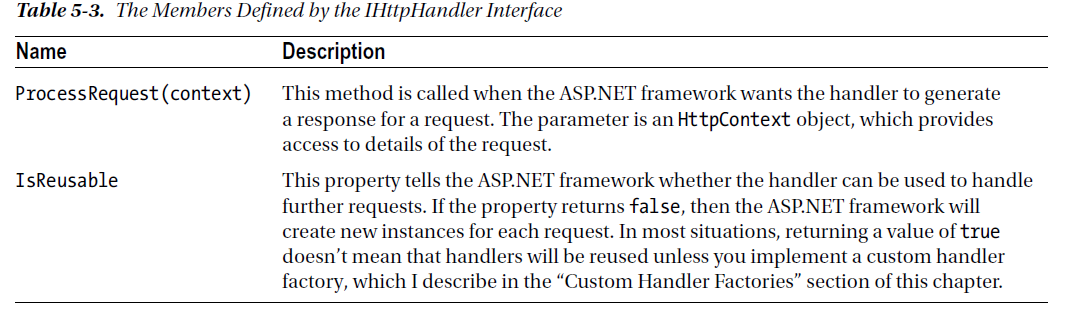
ASP.NET Handlers

Responsible for generating the response content for requests.



**Creating custom handler**

Create a class that implements IHttpHandler interface.



Handlers are registered in the system.webServer/handlers section of the Web.config file.