Getting Starting: Angular & Web API

Deborah Kurata

@deborahkurata | blogs.msmvps.com/deborahk/

Module Objectives

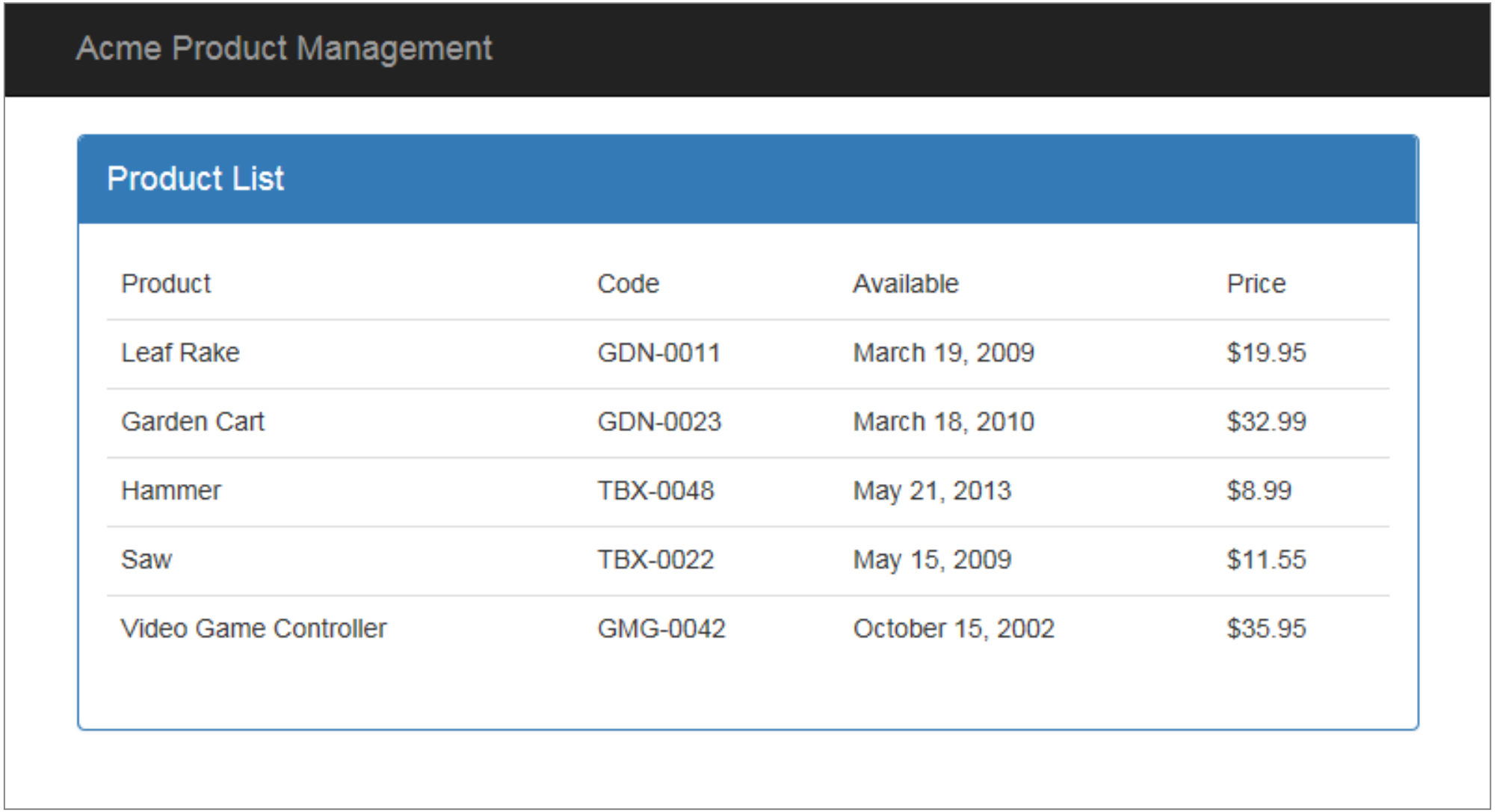
Review Web Application Architecture

Create the Projects

Examine the Anatomy of an Angular App

Examine the Anatomy of an ASP.NET Web API Service

Acme Product Management (APM)



Web Application Architecture from Front to Back

Web Browser

Web Server

URL Request (www.mysite.com)

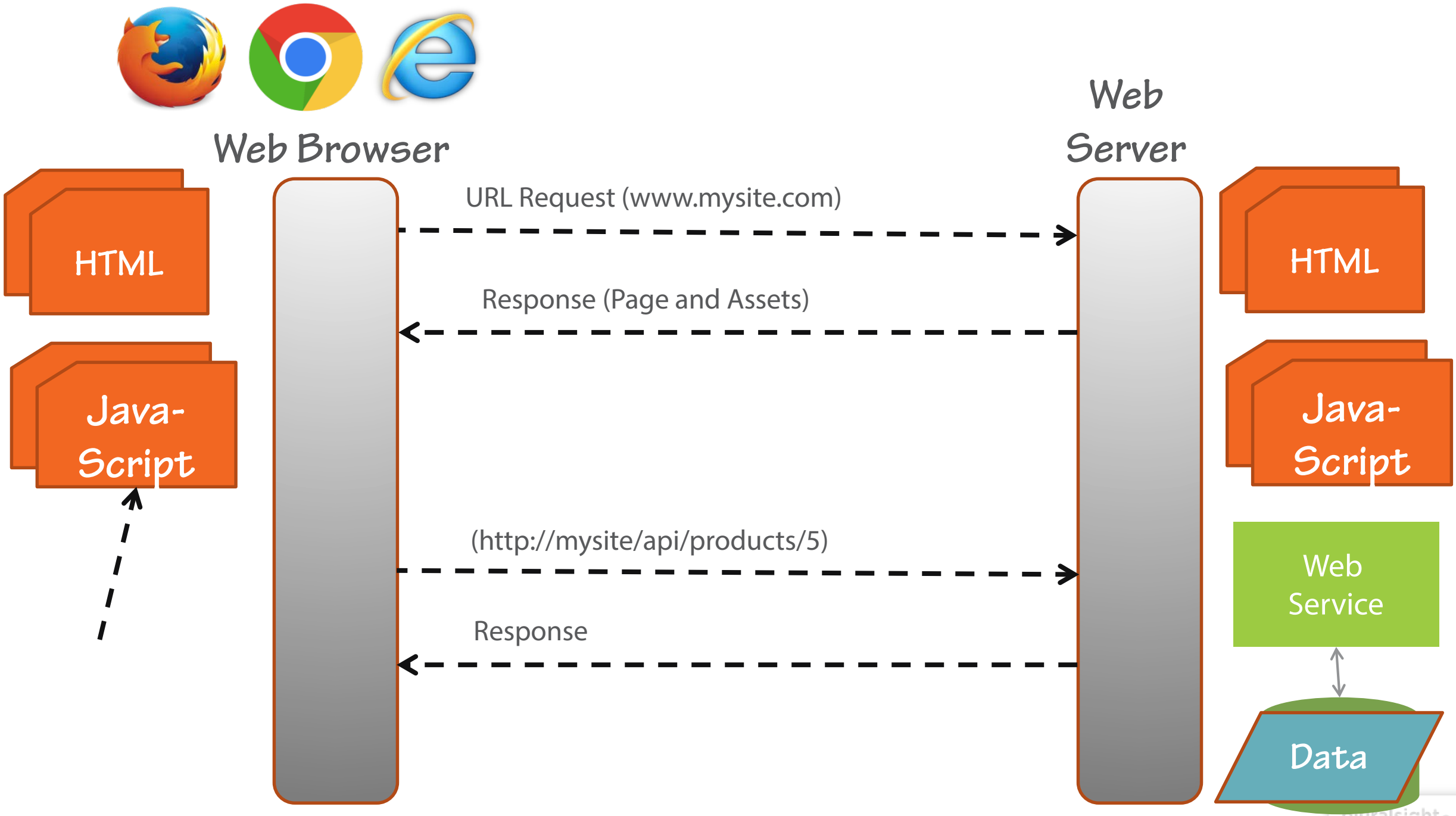
Response (Page and Assets)

HTML

Java-Script

Web Service

DB


Web Browser
Web Server
HTML
Java-Script
URL Request (www.mysite.com)
Response (Page and Assets)
(http://mysite/api/products/5)
Response
HTML
Java-Script
Web Service
Data

Why Angular?

Expressive HTML

Powerful Data Binding

Clean Programming

Built-in Services

Why ASP.NET Web API?

HTTP

RESTful

Supports broad range of clients

Built upon the .NET Framework

Built-in features for routing, actions, and more

Easy to learn

Building an ASP.NET Web API

Part of ASP.NET MVC app

Part of ASP.NET WebForms app

We will use this

Stand alone service

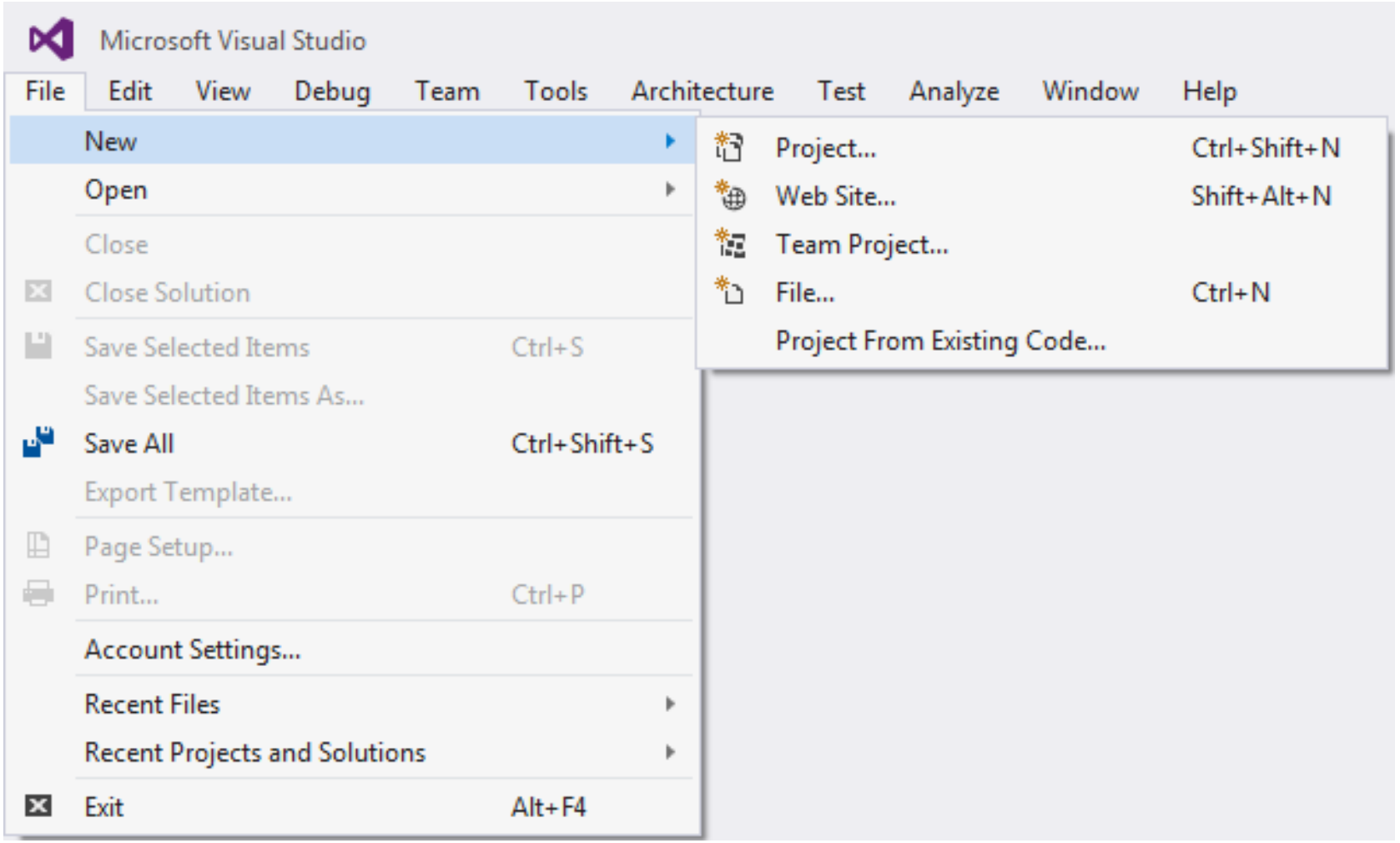
Editor

- Angular
  - Text Editor (NotePad)
  - Code Editor (WebStorm or Visual Studio)
- ASP.NET Web API
  - Visual Studio

Creating the Project: ASP.NET Web API

File | New | Project

ASP.NET template

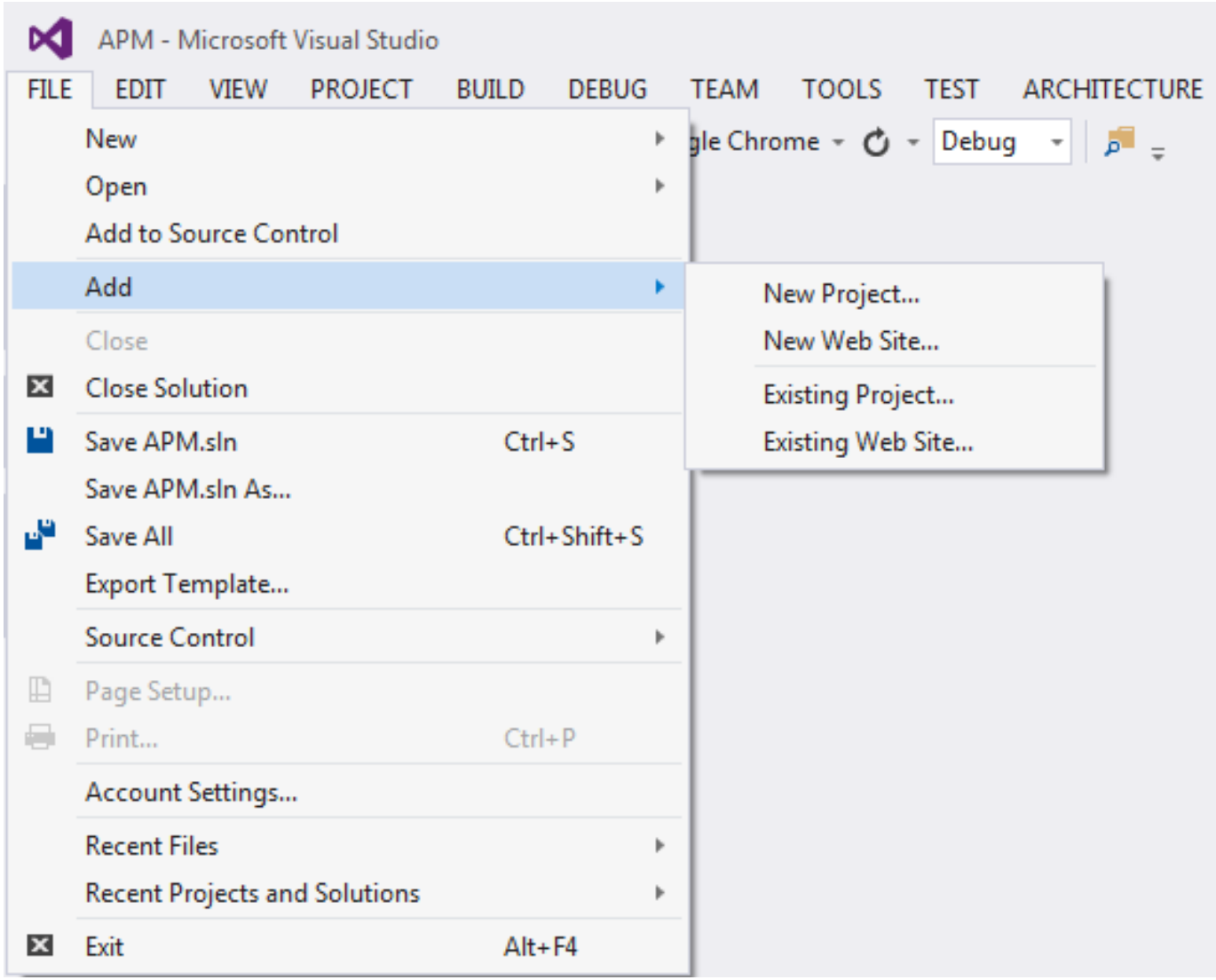


Creating the Project: Angular

Same
ASP.NET Web Project

Separate
ASP.NET Web Project

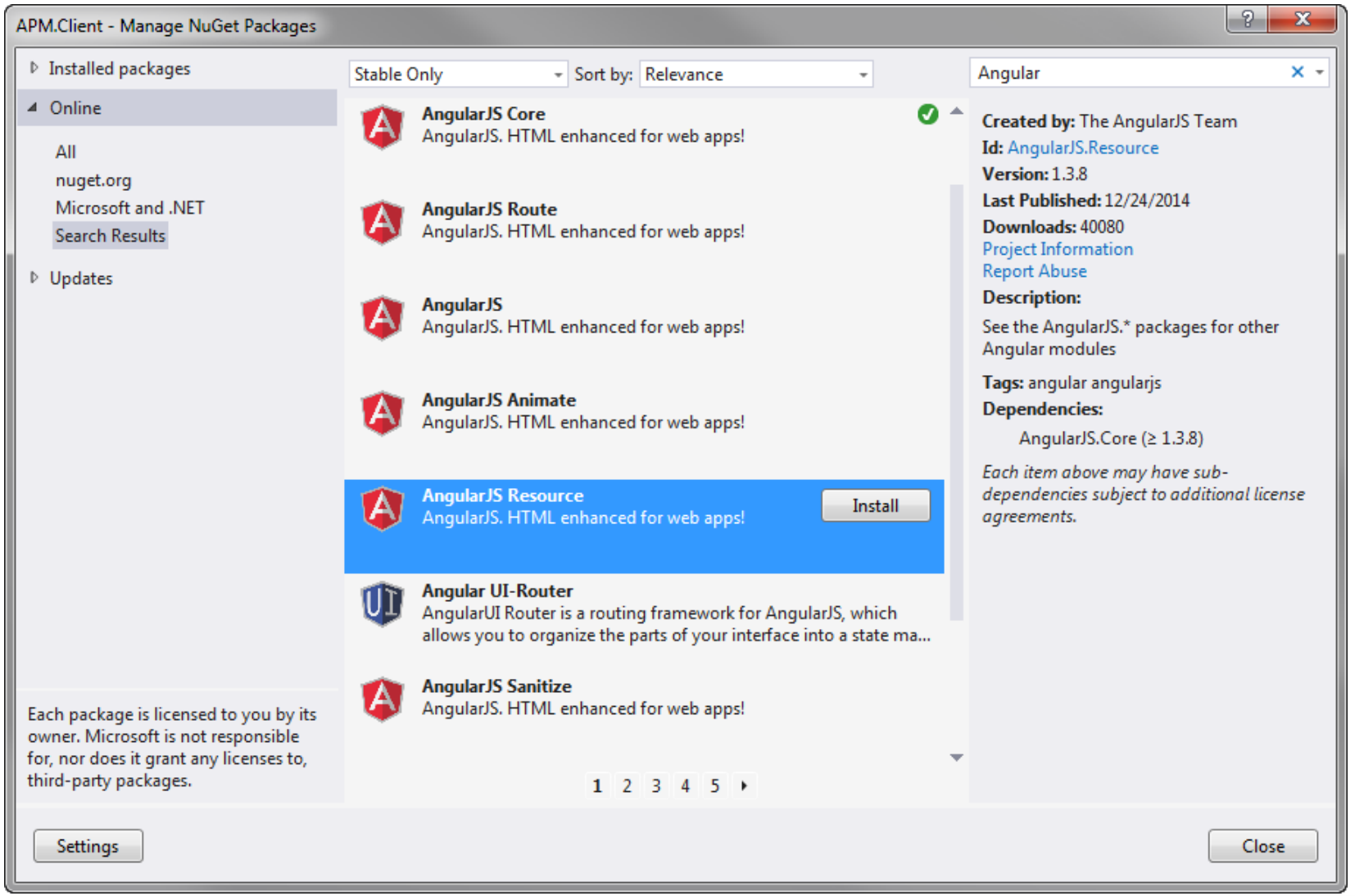
Web Site



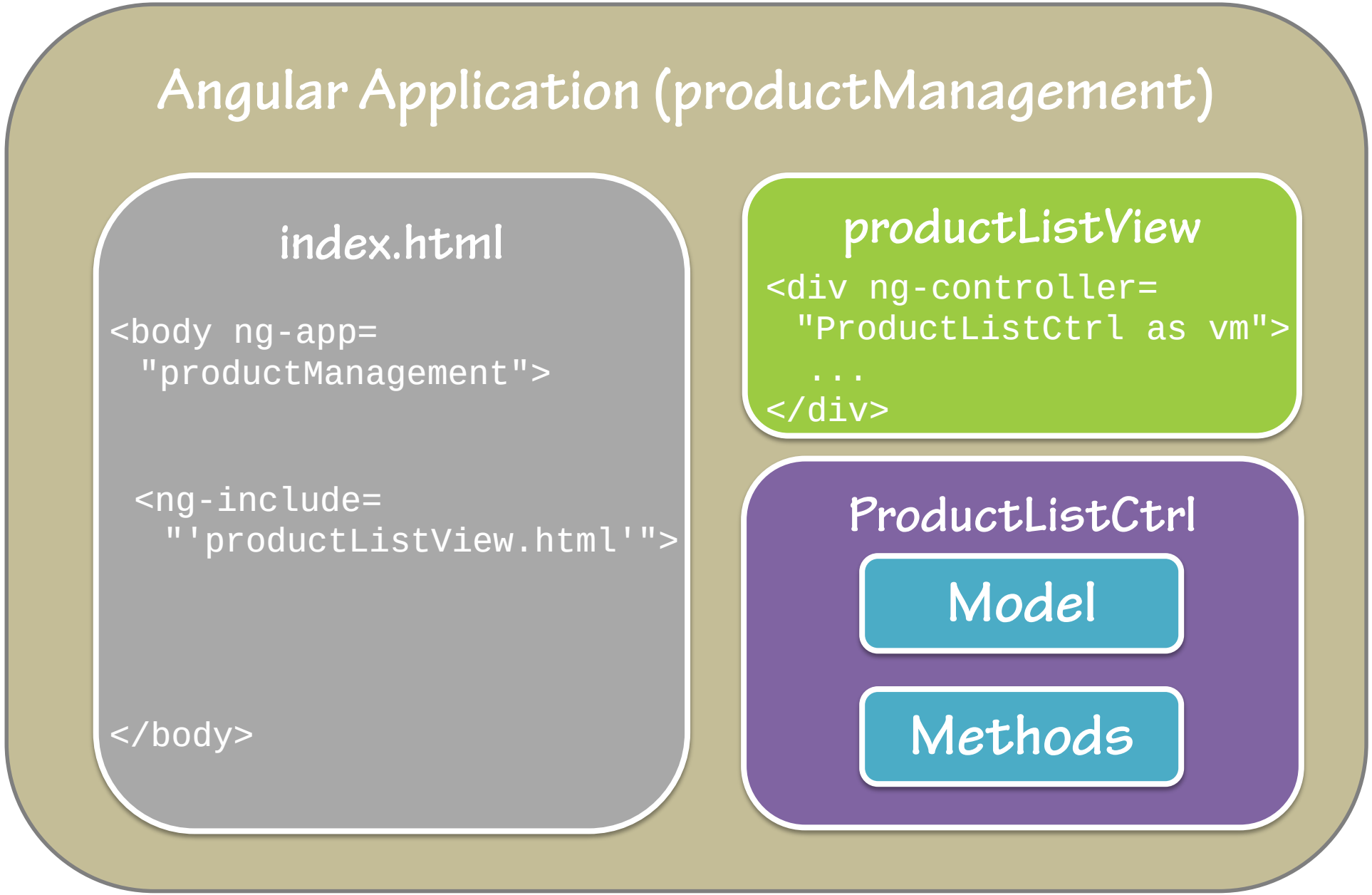
NuGet

AngularJS Core

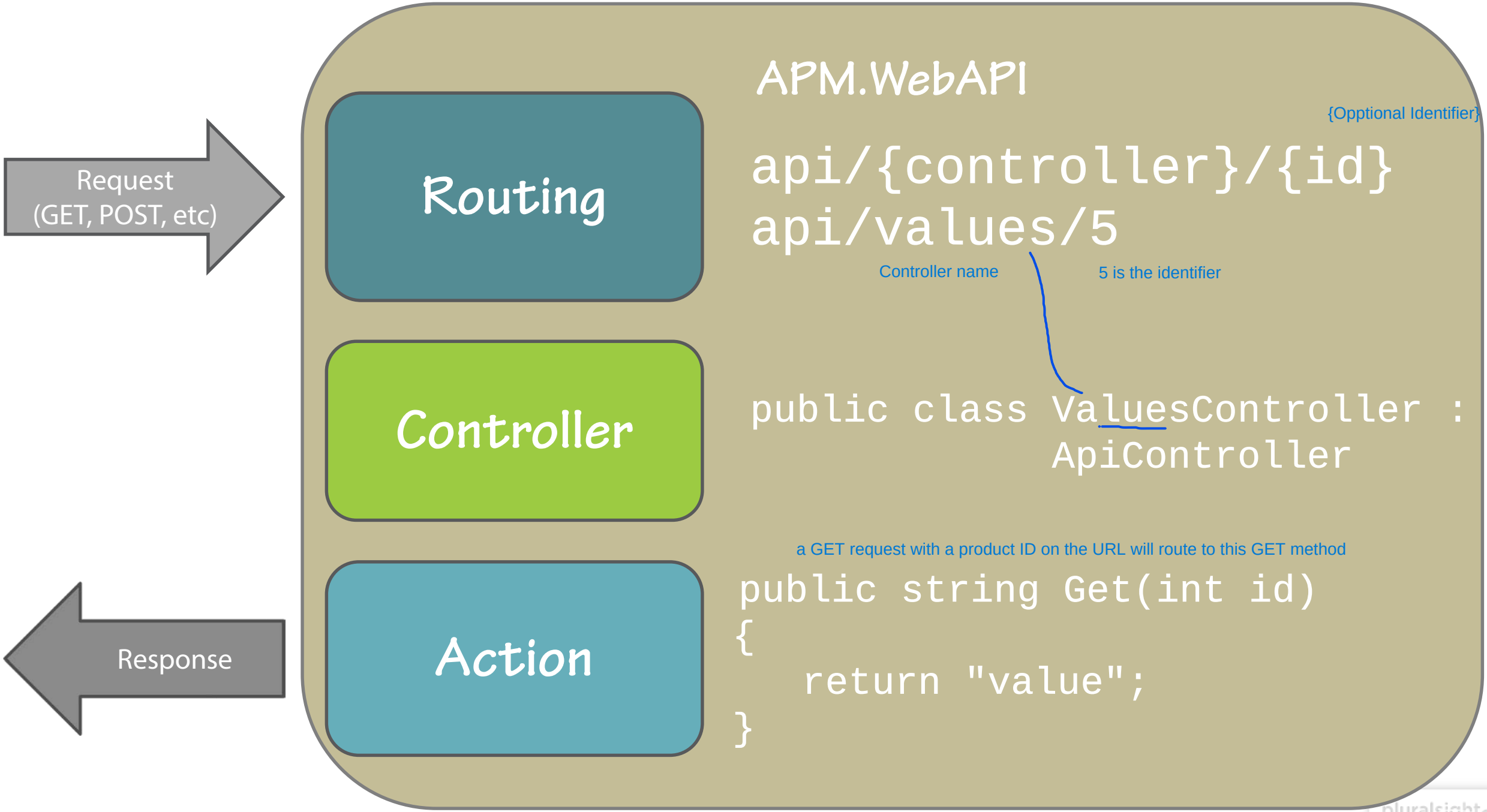
Bootstrap



Anatomy of an Angular Application



Anatomy of an ASP.NET Web API Service


Request
(GET, POST, etc)
Routing
Controller
Action
Response
APM.WebAPI
{Opptional Identifier}
api/{controller}/{id}
api/values/5
Controller name
5 is the identifier
public class ValuesController :
ApiController
a GET request with a product ID on the URL will route to this GET method
public string Get(int id)
{
return "value";
}

This Module Covered

- Web Application Architecture
- Creating the Projects
- Anatomy of an Angular App
- Anatomy of an ASP.NET Web API service