

# Action Results, Error Handling & Validation



Deborah Kurata

@deborahkurata | [blogs.msmvps.com/deborahk/](https://blogs.msmvps.com/deborahk/)

# Touch Points



- Michelangelo Buonarroti via Wikimedia Commons

<https://www.flickr.com/photos/34409164@N06/4277702120>

- ✓ Retrieving Data
- ✓ Filtering, Shaping, Querying Data
- ✓ Saving Data
- Error Handling
- User Authentication
- User Authorization

# Getting an Item

Acme Product Management

Basic Product Information

**Product Name**  Product name is required.

**Product Code**  Product Code is required

**Availability**

**Description**

GET Request

api /products/5

API

```
{
  "Description": "Curved claw steel hammer",
  "Price": 8.99,
  "ProductCode": "TBX-0048",
  "ProductId": 5,
  "ProductName": "Hammer",
  "ReleaseDate": "2013-05-21T00:00:00-07:00"
}
```

# Saving an Item

Acme Product Management

Basic Product Information

**Product Name**

**Product Code**

**Availability**

**Description**

PUT/POST Request  
→  
api /products/5

API

```
{  
  "Description": "Curved claw steel hammer",  
  "Price": 8.99,  
  "ProductCode": "TBX-0048",  
  "ProductId": 5,  
  "ProductName": "Hammer-Steel",  
  "ReleaseDate": "2013-05-21T00:00:00-07:00"  
}
```

# Module Objectives

Action Result

Exception Handling

Validation

# Action Results

# Web API Action Method Return Types

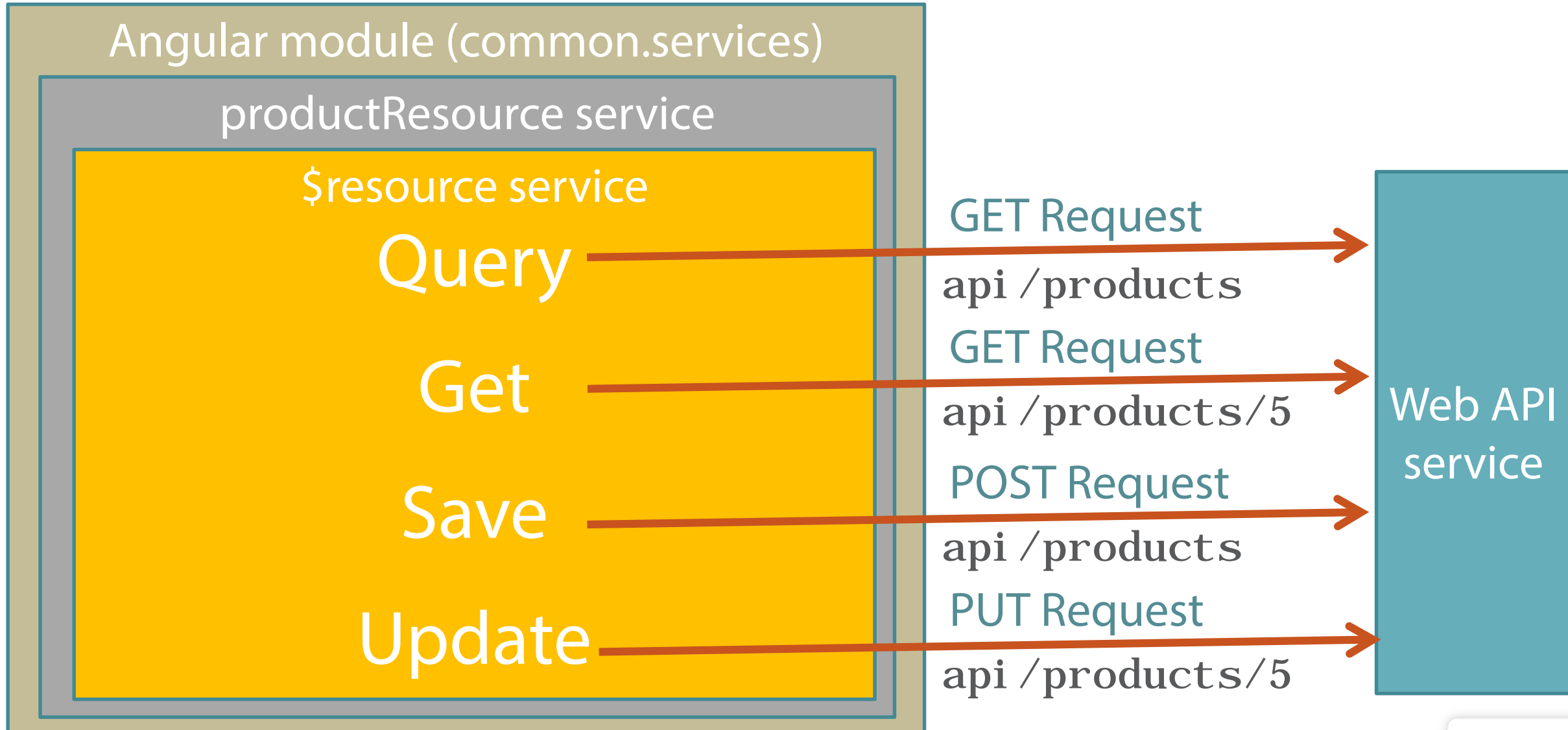
Return Type	Purpose
void	No data to return
<entity type>	Return specific data
HttpResponseMessage	Return detailed responses
IHttpActionResult	Simplify returning responses

# Action Result Helper Methods

Method	Method (cont)
Ok	NotFound
BadRequest	Redirect
Conflict	ResponseMessage
Created	Unauthorized
InternalServerError	



# \$resource Methods



# \$resource Methods

- HTTP GET "class" actions: `Resource.action([parameters], [success], [error])`  

```
productResource.get({ id: 5}, function (data) {  
    vm.products = data;  
})
```
- non-GET instance actions: `instance.$action([parameters], [success], [error])`  

```
vm.product.$update({ id: 5}, function (data) {  
    vm.message = "... Save Complete";  
})
```

# Exception Handling

```
[EnableQuery()]\n[ResponseType(typeof(Product))]\npublic IHttpActionResult Get()\n{\n    try\n    {\n        var productRepository = new ProductRepository();\n        return Ok(productRepository.Retrieve().AsQueryable());\n    }\n    catch (Exception ex)\n    {\n        return InternalServerError(ex);\n    }\n}
```

## Try/Catch Block

# Validation

# Validation

## Client-side (Angular)

- Immediate
- Better user experience
- Better performance

## Server-side (Web API)

- Cannot assume that every client application submits valid data
- Protect against malicious user

```
[Required()]
```

```
[MinLength(4)]
```

```
[MaxLength(12)]
```

```
public string ProductName { get; set; }
```

## Web API Validation Attributes

Data Annotations

Range, CreditCard, EmailAddress, Phone, ...

Custom: To execute a custom validation method

# This Module Covered



Action Results

Exception Handling

Validation



# Touch Points



- Michelangelo Buonarroti via Wikimedia Commons

<https://www.flickr.com/photos/34409164@N06/4277702120>

- ✓ Retrieving Data
- ✓ Filtering, Shaping, Querying Data
- ✓ Saving Data
- ✓ Error Handling

User Authentication

User Authorization