



GOLD PARTNER:



Hlavní odborný partner:



# ASP.NET Core Dependency Injection & Unit-testing

**Robert Haken**

MVP, MCT, MCSD, MCPD

software architect, HAVIT, s.r.o.

haken@havit.cz, @RobertHaken

16.–19. května 2016

**Tech·Ed**  
**DevCon** 

PRAHA 2016

# OCHUTNÁVKA

ASP.NET Core - MVC6 Controller - IEmailSender Dependency Injection demo

# Dependency Injection

*"Dependency Injection is a set of software design principles and patterns that enable us to develop loosely coupled code."*

Mark Seeman

# Dependency Injection Benefits

*loosely coupled code =>*

- Late-binding
- Extensibility
- Parallel Development
- Maintainability
- Testability

# DEMO

Price Resolver - Dependency Injection, Unit-Testing

# "Good" Unit Test

- automated + repeatable
- fully isolated
- consistent in its results
- runs quickly
- full control of the unit under test (all dependencies)
- relevant tomorrow
- easy to implement
- able to run it at the push of a button
- if fails => easy to detect what was expected

# Dependency Injection Containers

- Object Composition
- Object Lifetime
- Interception
  
- Code-based / XML-based configuration
  - Auto-wiring, convention-based configuration, ...

# ASP.NET Core DI "Container" - Register

- Startup.cs
  - `Configure[Env]Services(IServiceCollection services)`
- API - Lifestyles
  - `services.AddTransient<TAbstraction, TImplementation>()`
  - `services.AddScoped<TAbstraction, TImplementation>()`
  - `services.AddSingleton<TAbstraction, TImplementation>()`
  - `services.AddInstance<TAbstraction>(instance)`
  - plus factory-method verze



# DEMO

ASP.NET Core DI - LifeStyles

# ASP.NET Core DI "Container" - Register

- Startup.cs
  - `Configure[Env]Services(IServiceCollection services)`
- API - Lifestyles
  - `services.AddTransient<TAbstraction, TImplementation>()`
  - `services.AddScoped<TAbstraction, TImplementation>()`
  - `services.AddSingleton<TAbstraction, TImplementation>()`
  - `services.AddInstance<TAbstraction>(instance)`
  - plus factory-method verze

# ASP.NET Core DI "Container" - Injection

## Controllers

- constructor injection
- Action - method injection [FromServices]

## Views

- property injection @inject TService PropertyName

## Models

- property injection [FromServices]

## Startup: Configure()

- method injection

# ASP.NET Core DI "Container" - Injection II.

## Middleware

- constructor injection
- method injection `Invoke(...)`

## Filters

- "attribute" injection [`ServiceFilter(typeof(...))`]
- "attribute" injection [`TypeFilter(typeof(...))`]

## Other

- constructor injection

# Dependency Injection Patterns

- Composition Root
- Register-Resolve-Release
- Constructor Injection
- Property Injection
- Method Injection
- Ambient Context
- ~~Service Locator~~

Aktuální a navazující kurzy sledujte na [www.gopas.cz](http://www.gopas.cz)

## DÁREK PRO VÁS!

Vyplňte dotazníkové hodnocení a...

...získejte tričko **TechEd-DevCon 2016!**



**SOUTĚŽ! SOUTĚŽ! SOUTĚŽ!**

**TechEd party!**

**Xbowling Strašnice, 18. 5. 2016**

**Bud'te The Best IT Pro  
nebo The Best Developer**