

Health Checks v ASP.NET Core

Michal Altair Valášek



Altairis, s. r. o.

www.altair.blog

michal.valasek@altairis.cz

Tech·Ed

 **GOPAS**



Co jsou health checks?

- HTTP endpoint, který umožňuje ověřit, že je aplikace vnitřně v pořádku
 - Obsahuje jeden nebo více checků
- Logika checku je na programátorovi, např.
 - Dostupnost databáze
 - Možnost zápisu na disk (volné místo, práva, existence adresáře)
 - Dostupnost externích služeb
 - Stav služeb na pozadí
 - ...

Jak vytvořit health check?

- `Func<HealthCheckResult>`
 - Delegát, který vrátí výsledek
 - Vhodné pro jednoduché, jednořádkové checky
- Instance activated `IHealthCheck`
 - Pro běžné použití, přímo se při inicializaci konfiguruje
- Type activated `IHealthCheck`
 - Může používat dependency injection
 - Složitější použití více instancí téhož checku s různými parametry

Výsledek checku

Healthy	všechno je v pořádku
Degraded	stav není nominální, ale nejedná se o havárii
Unhealthy	testovaný parametr je nefunkční

- Výsledek může obsahovat textovou stavovou informaci
- Lze přidat libovolná metadata

Health check endpoint

- HTTP endpoint, který indikuje stav aplikace
 - **200 OK** všechny testy jsou Healthy nebo Degraded
 - **503 Service Unavailable** nejméně jeden test je Unhealthy
 - (nastavení lze změnit)
- Ve výchozím nastavení nevrací žádnou jinou informaci
 - Lze ale snadno změnit a vrátit např. JSON dokument
- Napojte ho na externí monitorovací službu
 - Test je spuštěn na základě HTTP requestu

Health check publisher

- Lze nastavit, že se checky mají spouštět automaticky
- Nastavuje se
 - `Delay` čas od spuštění aplikace do prvního testu
 - `Period` interval mezi jednotlivými testy
- Publisher může (a měl by) stav aplikace někam aktivně hlásit!

DEMO

Základní použití health checku

Tags a Predicate

- Jednotlivým checkům lze přidávat tagy
 - Libovolný string, použitelný pro filtrování
- Při tvorbě endpointů lze checky filtrovat podle tagů nebo jiných parametrů
 - Více endpointů pro různé skupiny checků

AspNetCore.Diagnostics.HealthChecks

- Nezávislý open source projekt
 - <https://github.com/Xabaril/AspNetCore.Diagnostics.HealthChecks>
- Nabízí desítky hotových checků
 - Dostupnost databází
 - Dostupnost cloudových úložišť
 - Dostupnost obecných síťových služeb
 - Dostupnost DNS záznamu
 - Dostupnost mail serverů a služeb
 - Paměť, disk, běh služby
 - ...

HealthChecks UI

- Webové rozhraní pro sledování stavu aplikace
 - Napojeno na health checks infrastrukturu
 - Možnost prohlížet z browseru
- Může zachovávat historii kontrol a statistiky
 - V takovém případě potřebuje úložiště
 - Vestavěná podpora pro SQL Server a Sqlite
 - InMemory úložiště nepřežije restart aplikace

DEMO

Filtrování checků pomocí tagů

Použití hotových checků

Použití HealthChecks UI

Health checks a background services

- Jeden z mála způsobů, jak kontrolovat že background worker běží
 - Bohužel na to neexistuje vestavěný mechanismus
 - Je ale jednoduché implementovat heartbeat logiku
- Pozor na vhodné nastavení hranic timeoutů, aby nedocházelo k falešným poplachům

DEMO

Kontrola background workerů

Čím health checky kontrolovat?

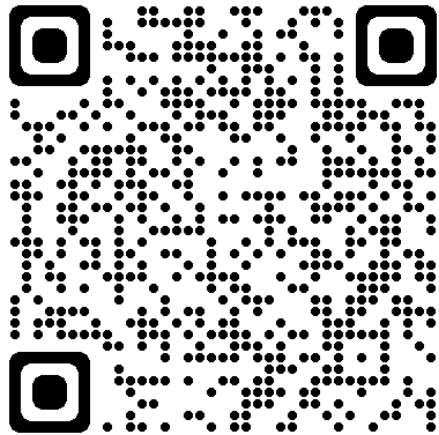
- On premises řešení
 - Obecně nedoporučuji, zejména pro menší projekty
 - Obvykle netriviální nastavení
 - Musíte je umístit mimo monitorovanou infrastrukturu
 - Nagios, Zabbix...
- Hostovaná řešení
 - Placená, ale obvykle nabízejí free tarif, který pro základní použití stačí
 - Vybírejte podle způsobu notifikací, nezávislém na vaší infrastruktuře
 - Uptime Robot, Wedos Online...

Podrobněji o health checks



Seriál ASP.NET Core Health Checks na kanálu Z-TECH:

https://www.youtube.com/playlist?list=PLFZurxJN0pMbFy_R9q7MQwPAwA_bwlZyG



✉ michal.valasek@altairis.cz

f facebook.com/rider.cz

g github.com/ridercz

t twitter.com/ridercz

in linkedin.com/in/ridercz

📄 www.altairis.cz

www.rider.cz

www.altair.blog

Tech·Ed

 **GOPAS**

