

Dokumentacja Projektu – „Projekt dwóch mikroserwisów – ProductService i InventoryService”

Piotr Buturla nr. Albumu 80486

1. Link do repozytorium

Repozytorium z pełnym kodem obu usług (.NET 8 Minimal API):

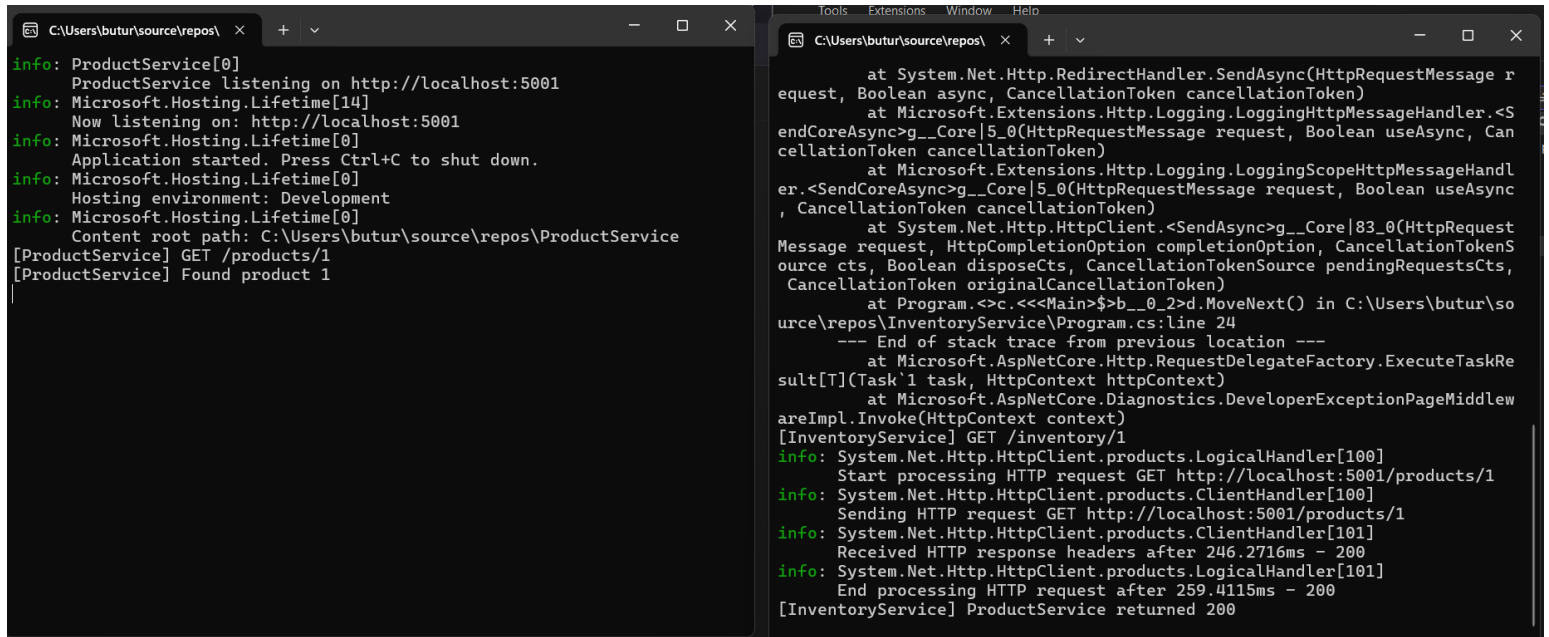
https://github.com/Vaxi2137/przetwarzanie_rozproszone.git

Repo zawiera:

- ProductService (serwis produktów)
- InventoryService (serwis magazynowy)
- README z instrukcją uruchomienia

2. Dokumentacja działania systemu (zrzuty ekranu)

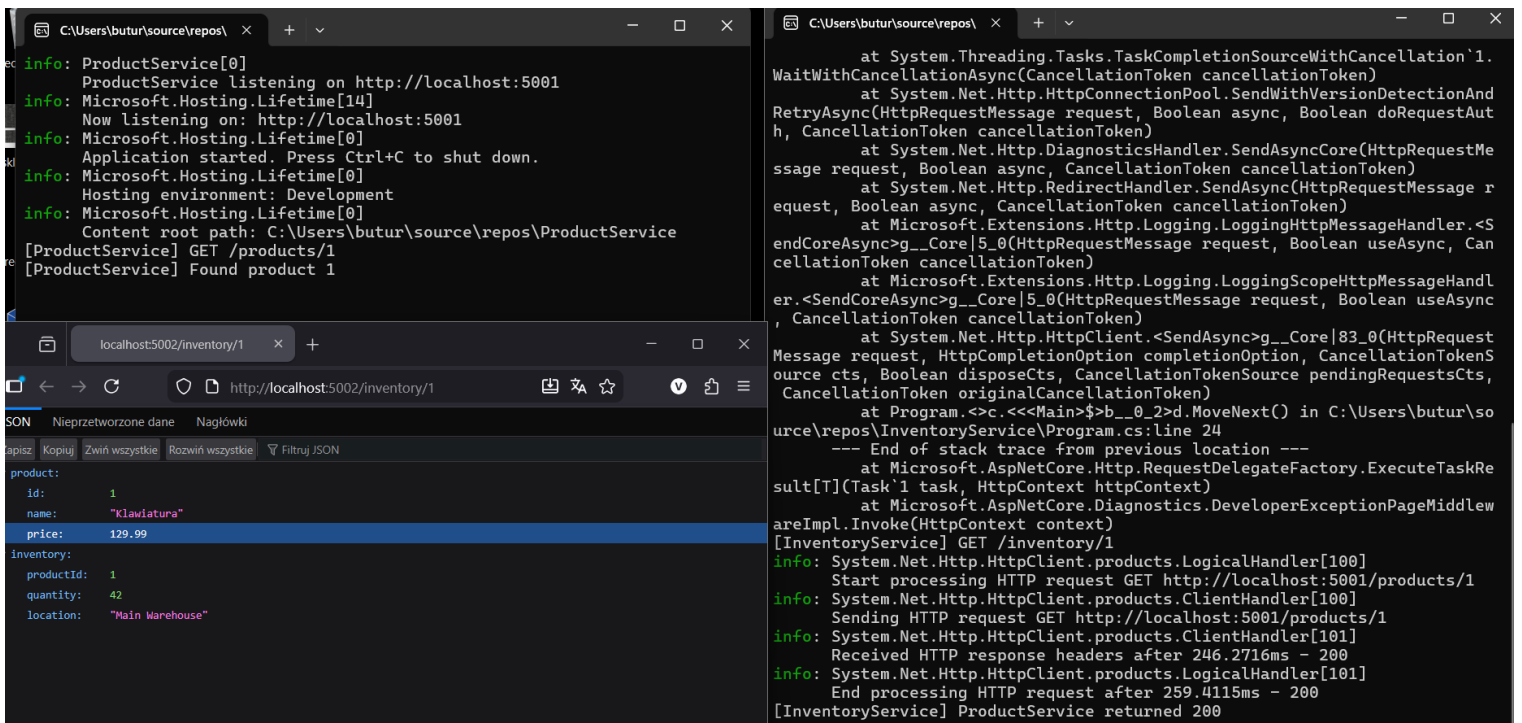
Screenshot 1 — Uruchomione serwisy



```
info: ProductService[0]
ProductService listening on http://localhost:5001
info: Microsoft.Hosting.Lifetime[14]
Now listening on: http://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
Content root path: C:\Users\butur\source\repos\ProductService
[ProductService] GET /products/1
[ProductService] Found product 1

at System.Net.Http.RedirectHandler.SendAsync(HttpRequestMessage request, Boolean async, CancellationToken cancellationToken)
at Microsoft.Extensions.Http.Logging.LoggingHttpMessageHandler.<SendCoreAsync>g__Core|5_0(HttpRequestMessage request, Boolean useAsync, CancellationToken cancellationToken)
at Microsoft.Extensions.Http.Logging.LoggingScopeHttpMessageHandler.<SendCoreAsync>g__Core|5_0(HttpRequestMessage request, Boolean useAsync, CancellationToken cancellationToken)
at System.Net.Http.HttpClient.<SendAsync>g__Core|83_0(HttpRequestMessage request, HttpCompletionOption completionOption, CancellationTokenSource cts, Boolean disposeCts, CancellationTokenSource pendingRequestsCts, CancellationToken originalCancellationToken)
at Program.<>c.<<Main>$>b__0_2.MoveNext() in C:\Users\butur\source\repos\InventoryService\Program.cs:line 24
--- End of stack trace from previous location ---
at Microsoft.AspNetCore.Http.RequestDelegateFactory.ExecuteTaskResult[T](Task<T> task, HttpContext httpContext)
at Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddlewareImpl.Invoke(HttpContext context)
[InventoryService] GET /inventory/1
info: System.Net.Http.HttpClient.products.LogicalHandler[100]
Start processing HTTP request GET http://localhost:5001/products/1
info: System.Net.Http.HttpClient.products.ClientHandler[100]
Sending HTTP request GET http://localhost:5001/products/1
info: System.Net.Http.HttpClient.products.ClientHandler[101]
Received HTTP response headers after 246.2716ms - 200
info: System.Net.Http.HttpClient.products.LogicalHandler[101]
End processing HTTP request after 259.4115ms - 200
[InventoryService] ProductService returned 200
```

Screenshot 2 — Scenariusz sukcesu



```
info: ProductService[0]
ProductService listening on http://localhost:5001
info: Microsoft.Hosting.Lifetime[14]
Now listening on: http://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
Content root path: C:\Users\butur\source\repos\ProductService
[ProductService] GET /products/1
[ProductService] Found product 1

at System.Threading.Tasks.TaskCompletionSourceWithCancellation`1.WaitWithCancellationAsync(CancellationToken cancellationToken)
at System.Net.Http.HttpConnectionPool.SendWithVersionDetectionAndRetryAsync(HttpRequestMessage request, Boolean async, Boolean doRequestAuth, CancellationToken cancellationToken)
at System.Net.Http.DiagnosticsHandler.SendAsyncCore(HttpRequestMessage request, Boolean async, CancellationToken cancellationToken)
at System.Net.Http.RedirectHandler.SendAsync(HttpRequestMessage request, Boolean async, CancellationToken cancellationToken)
at Microsoft.Extensions.Http.Logging.LoggingHttpMessageHandler.<SendCoreAsync>g__Core|5_0(HttpRequestMessage request, Boolean useAsync, CancellationToken cancellationToken)
at Microsoft.Extensions.Http.Logging.LoggingScopeHttpMessageHandler.<SendCoreAsync>g__Core|5_0(HttpRequestMessage request, Boolean useAsync, CancellationToken cancellationToken)
at System.Net.Http.HttpClient.<SendAsync>g__Core|83_0(HttpRequestMessage request, HttpCompletionOption completionOption, CancellationTokenSource cts, Boolean disposeCts, CancellationTokenSource pendingRequestsCts, CancellationToken originalCancellationToken)
at Program.<>c.<<Main>$>b__0_2.MoveNext() in C:\Users\butur\source\repos\InventoryService\Program.cs:line 24
--- End of stack trace from previous location ---
at Microsoft.AspNetCore.Http.RequestDelegateFactory.ExecuteTaskResult[T](Task<T> task, HttpContext httpContext)
at Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddlewareImpl.Invoke(HttpContext context)
[InventoryService] GET /inventory/1
info: System.Net.Http.HttpClient.products.LogicalHandler[100]
Start processing HTTP request GET http://localhost:5001/products/1
info: System.Net.Http.HttpClient.products.ClientHandler[100]
Sending HTTP request GET http://localhost:5001/products/1
info: System.Net.Http.HttpClient.products.ClientHandler[101]
Received HTTP response headers after 246.2716ms - 200
info: System.Net.Http.HttpClient.products.LogicalHandler[101]
End processing HTTP request after 259.4115ms - 200
[InventoryService] ProductService returned 200
```

localhost:5002/inventory/1

http://localhost:5002/inventory/1

JSON Nieprzetworzone dane Nagłówki

apisać Kopiuć Zwiń wszystkie Rozwiń wszystkie Filtruj JSON

```
product:
  id: 1
  name: "Klawiatura"
  price: 129.99
inventory:
  productId: 1
  quantity: 42
  location: "Main Warehouse"
```

Screenshot 3 — Scenariusz błędu (404)

```
info: ProductService[0]
ProductService listening on http://localhost:5001
info: Microsoft.Hosting.Lifetime[14]
Now listening on: http://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
Content root path: C:\Users\butur\source\repos\ProductService
[ProductService] GET /products/1
[ProductService] Found product 1
[ProductService] GET /products/999
[ProductService] Not found 999

at System.Threading.Tasks.TaskCompletionSourceWithCancellation`1.
WaitWithCancellationAsync(CancellationTok...
at System.Net.Http.HttpConnectionPool.SendWithVersionDetectionAnd
RetryAsync(HttpRequestMessage request, Boolean async, Boolean doRequestAut
h, CancellationToken cancellationTok...
at System.Net.Http.DiagnosticsHandler.SendAsyncCore(HttpRequestMe
ssage request, Boolean async, CancellationTok...
at System.Net.Http.RedirectHandler.SendAsync(HttpRequestMessage r
equest, Boolean async, CancellationTok...
at Microsoft.Extensions.Http.Logging.LoggingHttpMessageHandler.<S
endCoreAsync>g__Core|5_0(HttpRequestMessage request, Boolean useAsync, Can
cellationToken cancellationTok...
at Microsoft.Extensions.Http.Logging.LoggingScopeHttpMessageHandl
er.<SendCoreAsync>g__Core|5_0(HttpRequestMessage request, Boolean useAsync
, CancellationTok...
at System.Net.Http.HttpClient.<SendAsync>g__Core|83_0(HttpRequest
Message request, HttpCompletionOption completionOption, CancellationTok...
at Program.<>c.<<<Main>$>b__0_2d.MoveNext() in C:\Users\butur\so
urce\repos\InventoryService\Program.cs:line 24
--- End of stack trace from previous location ---
at Microsoft.AspNetCore.Http.RequestDelegateFactory.ExecuteTaskRe
sult[T](Task`1 task, HttpContext httpContext)
at Microsoft.AspNetCore.Diagnostics.DeveloperExceptionPageMiddlew
areImpl.Invoke(HttpContext context)
[InventoryService] GET /inventory/1
info: System.Net.Http.HttpClient.products.LogicalHandler[100]
Start processing HTTP request GET http://localhost:5001/products/1
info: System.Net.Http.HttpClient.products.ClientHandler[100]
Sending HTTP request GET http://localhost:5001/products/1
info: System.Net.Http.HttpClient.products.ClientHandler[101]
Received HTTP response headers after 246.2716ms - 200
info: System.Net.Http.HttpClient.products.LogicalHandler[101]
End processing HTTP request after 259.4115ms - 200
[InventoryService] ProductService returned 200
[InventoryService] GET /inventory/999
info: System.Net.Http.HttpClient.products.LogicalHandler[100]
Start processing HTTP request GET http://localhost:5001/products/999
info: System.Net.Http.HttpClient.products.ClientHandler[100]
Sending HTTP request GET http://localhost:5001/products/999
```

localhost:5002/inventory/999

message: "Product not found in catalog"

Podsumowanie

System składa się z dwóch mikroservisów komunikujących się przez HTTP. InventoryService pobiera dane z ProductService i zwraca wynik działania. Udokumentowano poprawne działanie systemu w scenariuszu sukcesu oraz błędu.