Introduction to

.NET Aspire

Presenter Name
Company Name

𝕏 /@Account

▶ /@Account

.NET Aspire

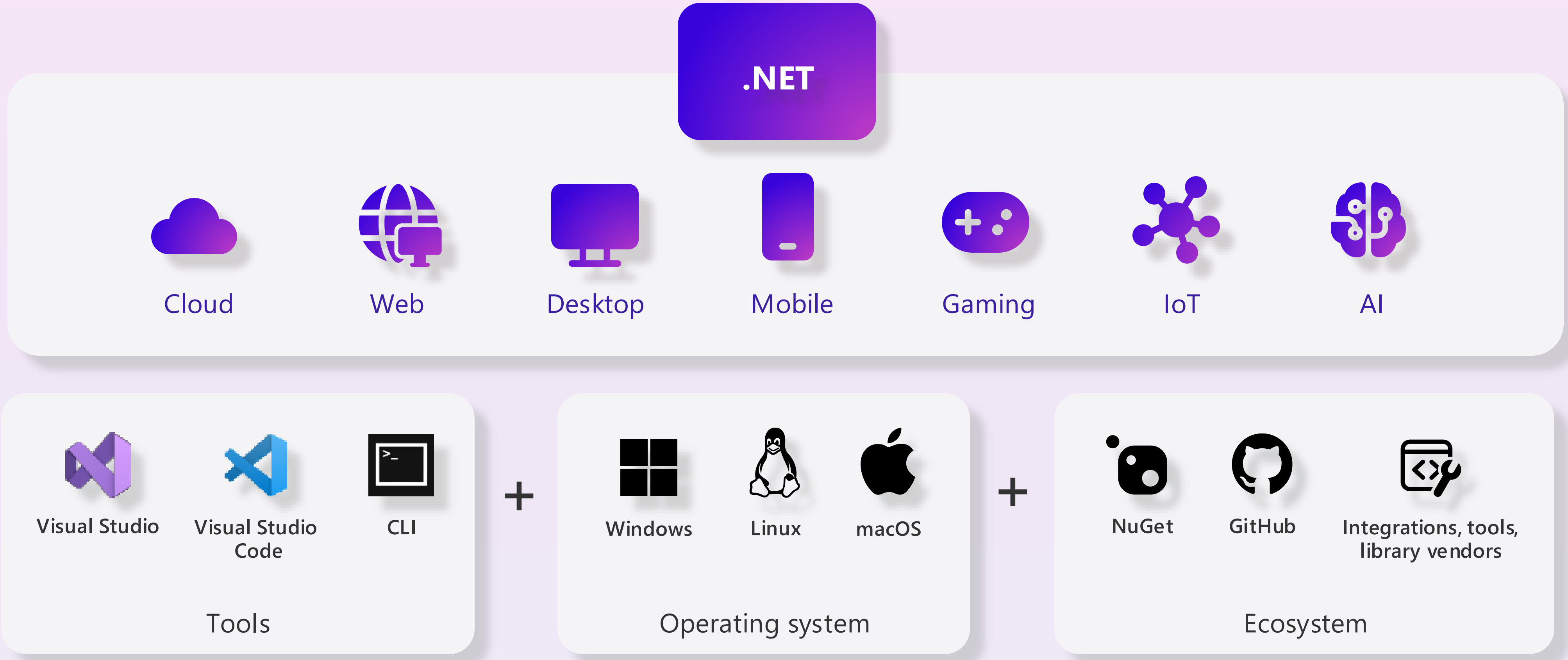
All Learning Resources

aka.ms/letslearn/dotnet/aspire

Workshop Repository

github.com/dotnet-presentations/dotnet-aspire-workshop

Build anything with a unified platform



.NET

Cloud
Web
Desktop
Mobile
Gaming
IoT
AI

Visual Studio
Visual Studio Code
CLI

Tools

+

Windows
Linux
macOS

Operating system

+

NuGet
GitHub
Integrations, tools, library vendors

Ecosystem

Every App Needs

Observability

Resiliency

Scalability

Manageability

.NET 9 Includes

Observability

- Built in metrics with dimensions
- DI integration for metrics
- Better Logging support (faster, can object serialization)
- Enrichment
- Redaction
- Testing fakes for Logging & Metrics

Resiliency

- New Polly based resiliency packages
- SignalR Stateful Reconnect

Scalability

- AOT (increased density)
- Performance
- Chiseled Ubuntu

Manageability

- Certificate auto-rotation support in Kestrel

It’s Still Not Easy 🙁

Complex

Getting Started

Choices

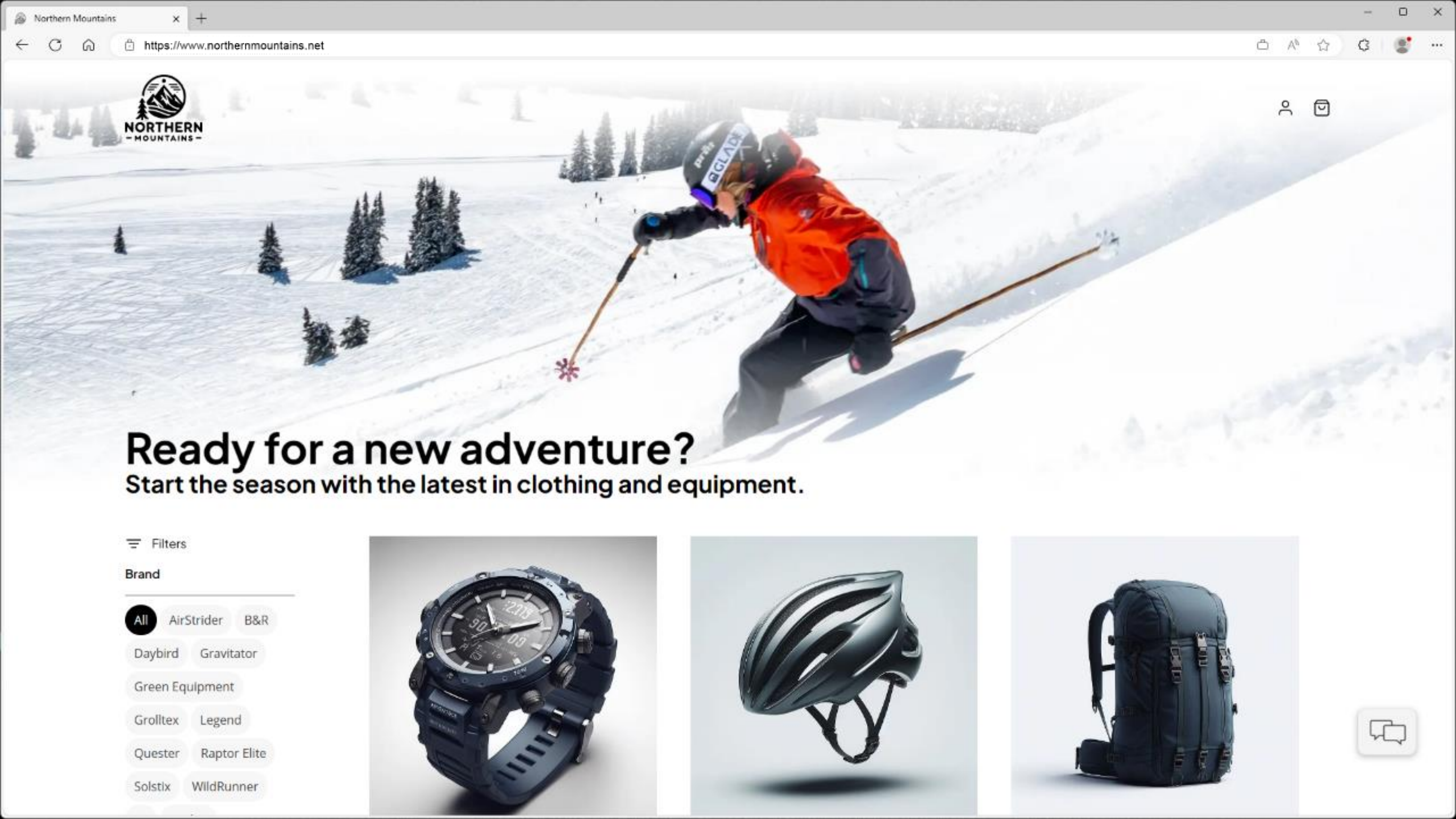
Paved Path

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

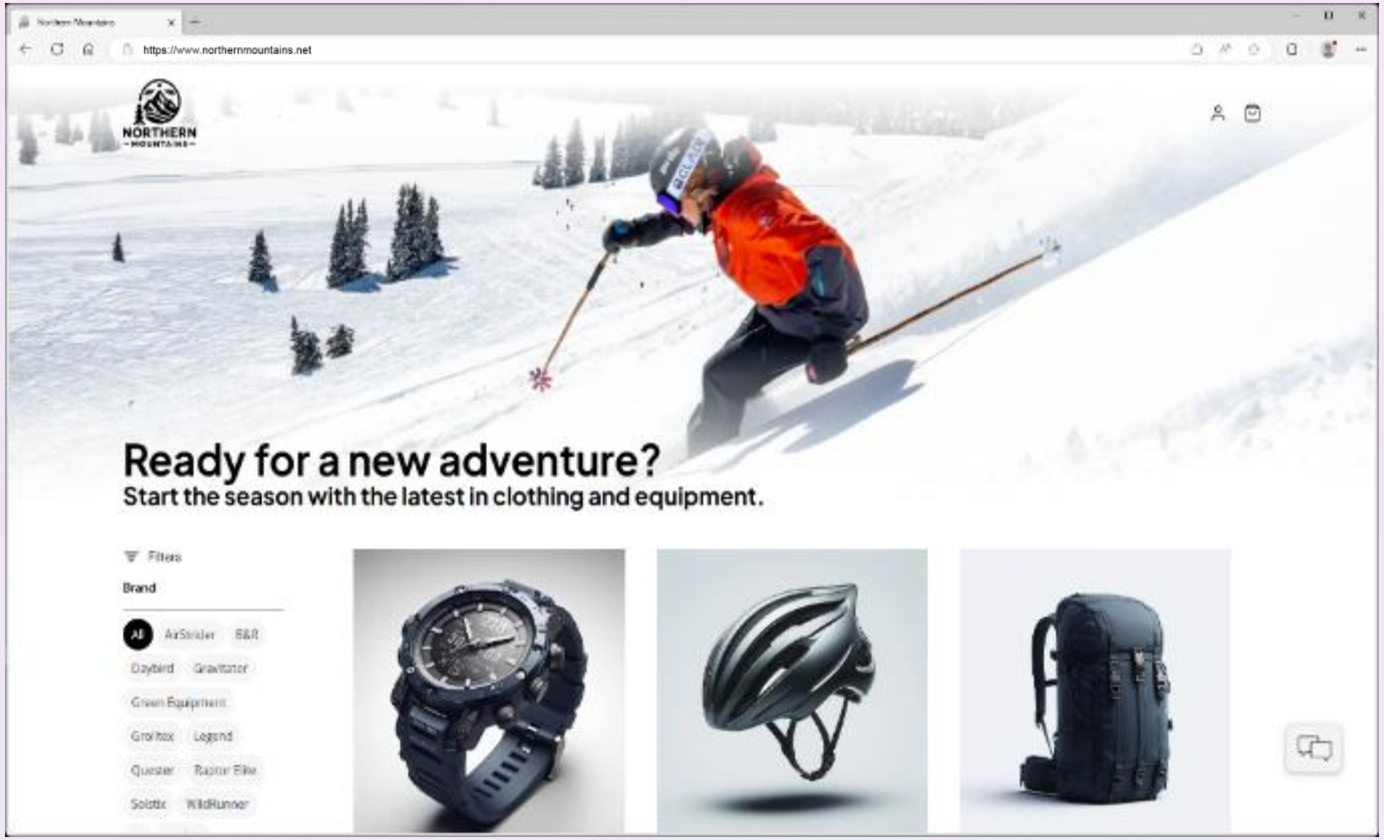
.NET Aspire

What does that mean? Who is it for? Should I use it?



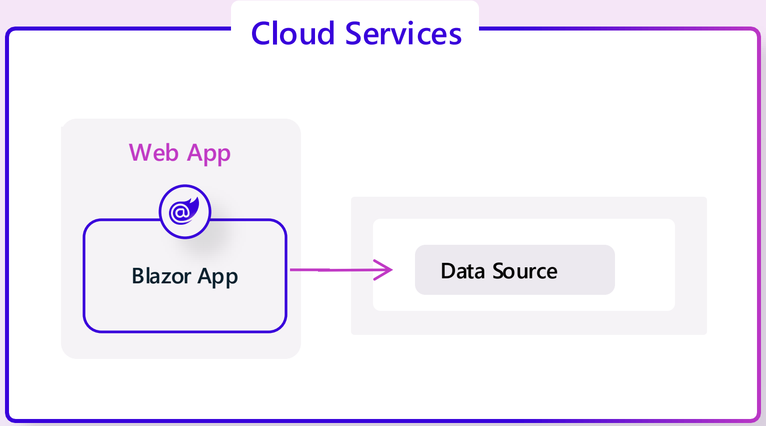
Ready for a new adventure?

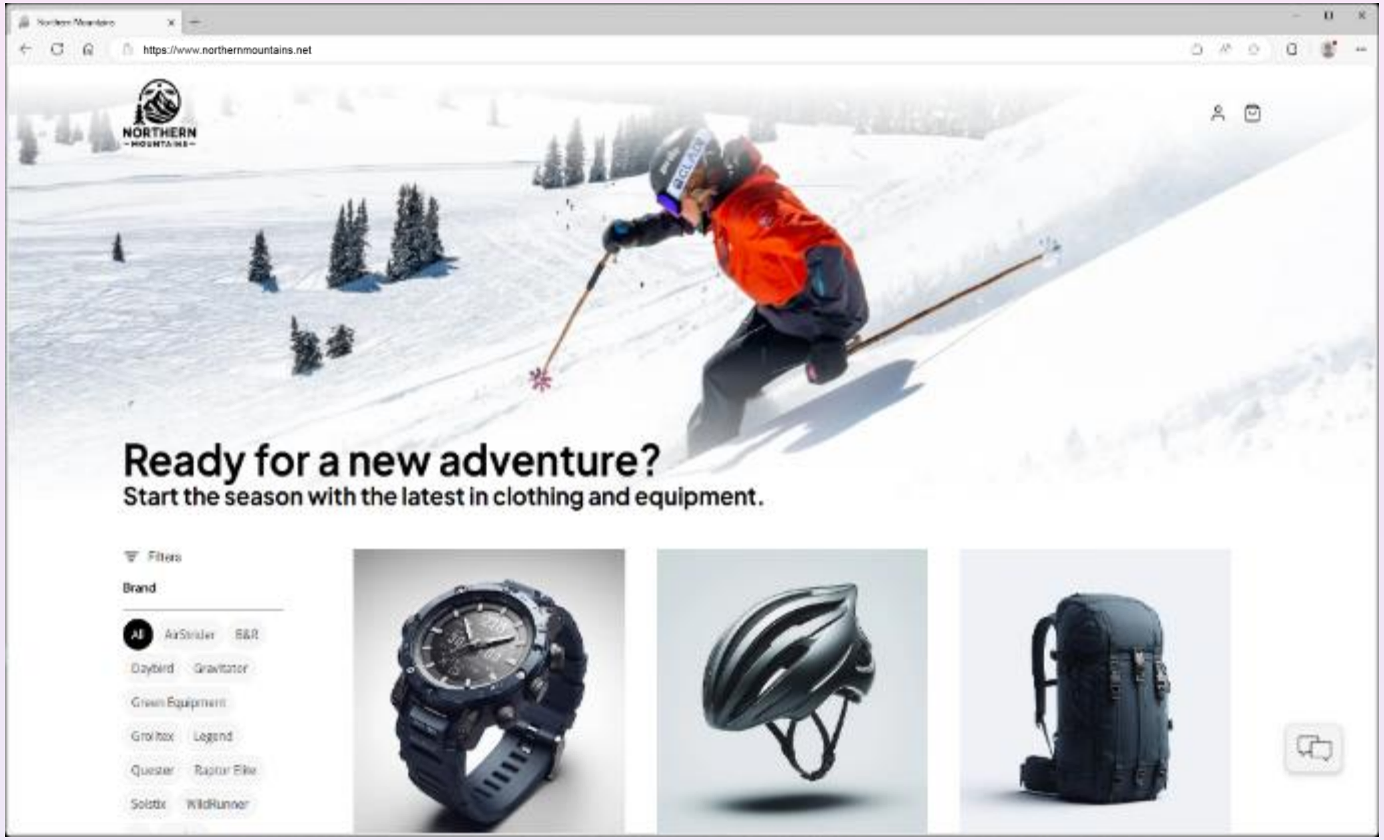
Start the season with the latest in clothing and equipment.



Ready for a new adventure?

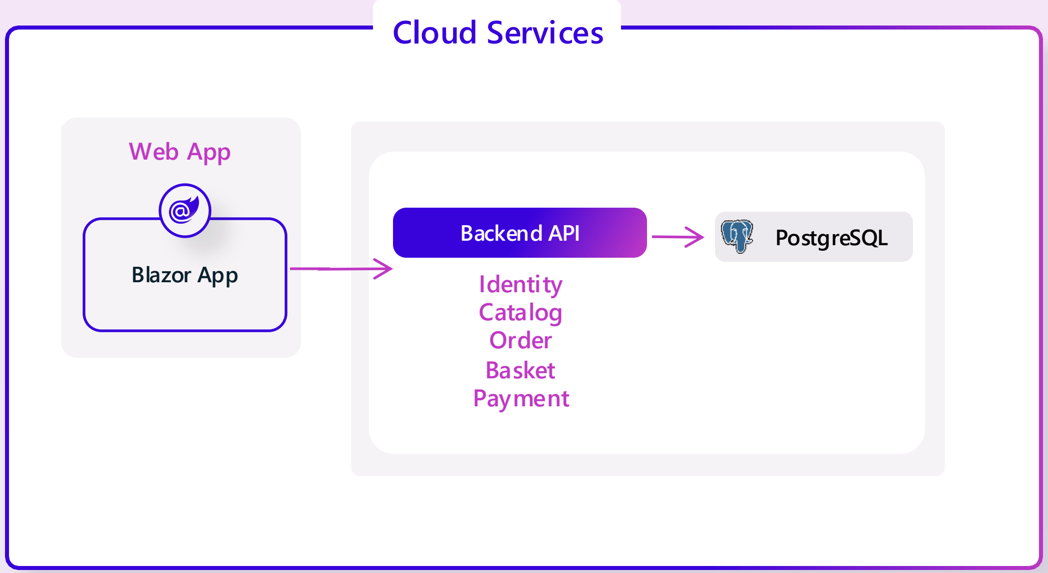
Start the season with the latest in clothing and equipment.





Ready for a new adventure?

Start the season with the latest in clothing and equipment.


Cloud Services

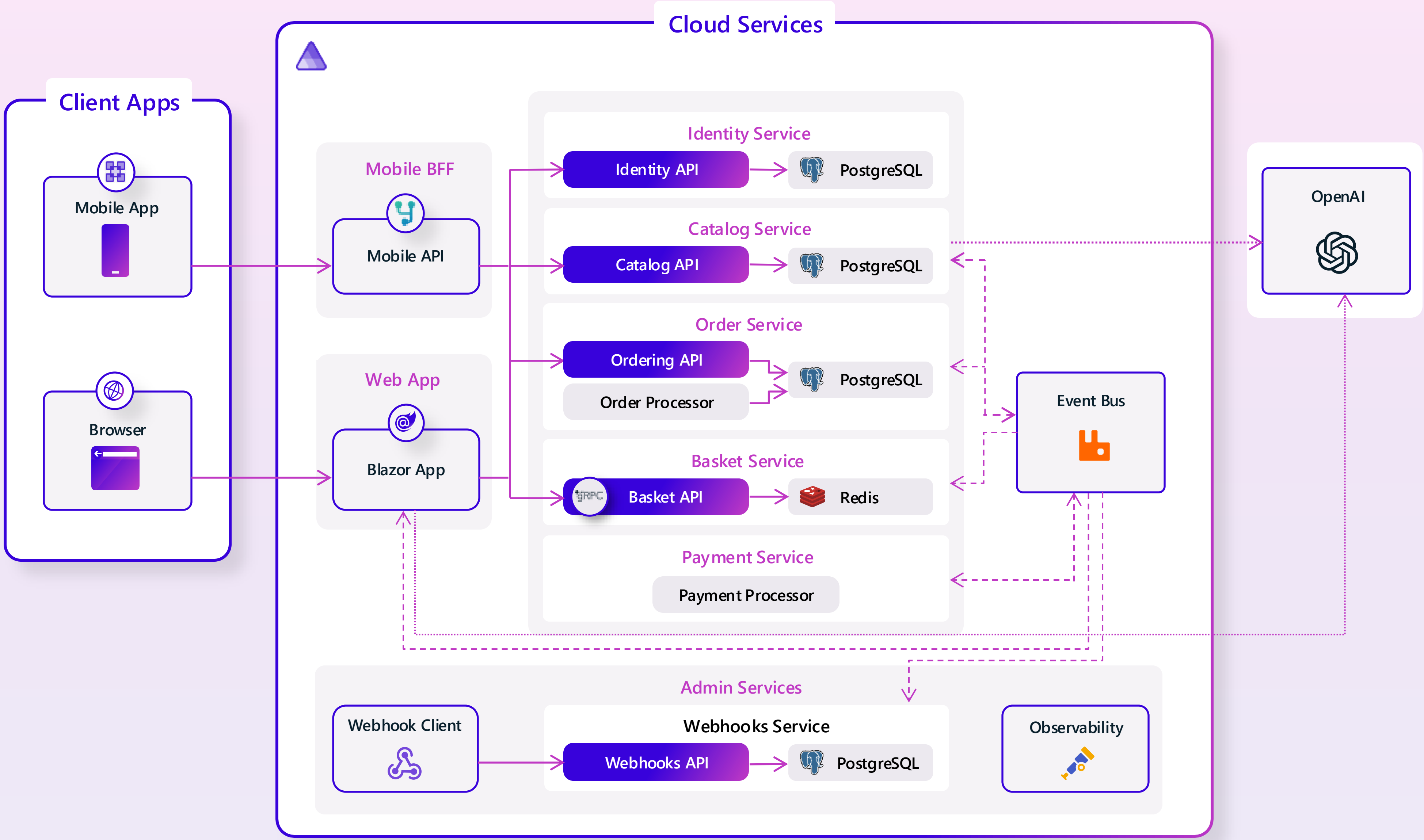
Web App

Blazor App

Backend API

Identity
Catalog
Order
Basket
Payment

PostgreSQL


Client Apps
Mobile App
Browser
Cloud Services
Mobile BFF
Mobile API
Web App
Blazor App
Identity Service
Identity API
PostgreSQL
Catalog Service
Catalog API
PostgreSQL
Order Service
Ordering API
Order Processor
PostgreSQL
Basket Service
gRPC
Basket API
Redis
Payment Service
Payment Processor
Event Bus
OpenAI
Admin Services
Webhook Client
Webhooks Service
Webhooks API
PostgreSQL
Observability

.NET Aspire

Works with ALL scale of applications and grows with you!

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

- Smart Defaults
- Orchestration
- Integrations
- Developer Dashboard
- Service Discovery
- Deployment

DEMO

Initial App Walkthrough

Clone: https://github.com/dotnet-presentations/dotnet-aspire-workshop

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

- Smart Defaults
- Developer Dashboard
- Orchestration
- Service Discovery
- Integrations
- Deployment

.NET Aspire Service Defaults

Observability

Resiliency

Health Checks

Observability

```
public static IHostApplicationBuilder ConfigureOpenTelemetry(this
                                        IHostApplicationBuilder builder)
{
    builder.Logging.AddOpenTelemetry(logging =>
    {
        logging.IncludeFormattedMessage = true;
        logging.IncludeScopes = true;
    });

    builder.Services.AddOpenTelemetry()
        .WithMetrics(metrics =>
        {
            metrics.AddAspNetCoreInstrumentation()
                .AddHttpClientInstrumentation()
                .AddRuntimeInstrumentation();
        })
        .WithTracing(tracing =>
        {
            tracing.AddAspNetCoreInstrumentation()
                .AddHttpClientInstrumentation();
        });

    builder.AddOpenTelemetryExporters();

    return builder;
}
```

Observability

```
private static IHostApplicationBuilder AddOpenTelemetryExporters(this
                                               IHostApplicationBuilder builder)
{
    var otlpEndpoint = builder.Configuration["OTEL_EXPORTER_OTLP_ENDPOINT"];
    var useOtlpExporter = !string.IsNullOrWhiteSpace(otlpEndpoint);

    if (useOtlpExporter)
    {
        builder.Services.AddOpenTelemetry().UseOtlpExporter();
    }

    // Uncomment the following lines to enable the Azure Monitor exporter
    //if (!string.IsNullOrEmpty(builder.Configuration["APPINSIGHTS_CONNECTION"]))
    //{
    //    builder.Services.AddOpenTelemetry()
    //       .UseAzureMonitor();
    //}

    return builder;
}
```

Resiliency

```
builder.Services.ConfigureHttpClientDefaults(http =>
{
    // Turn on resilience by default
    http.AddStandardResilienceHandler();
});
```

Health Checks

```
public static IHostApplicationBuilder AddDefaultHealthChecks(this
                                              IHostApplicationBuilder builder)
{
    builder.Services.AddHealthChecks()
        // Add a default liveness check to ensure app is responsive
        .AddCheck("self", () => HealthCheckResult.Healthy(), ["live"]);

    return builder;
}

public static WebApplication MapDefaultEndpoints(this WebApplication app)
{
    if (app.Environment.IsDevelopment())
    {
        app.MapHealthChecks("/health");
        app.MapHealthChecks("/alive", new HealthCheckOptions
        {
            Predicate = r => r.Tags.Contains("live")
        });
    }

    return app;
}
```

DEMO

.NET Aspire - Smart Defaults

Q&A

.NET Aspire - Smart Defaults

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

Smart Defaults

Developer Dashboard

Orchestration

Service Discovery

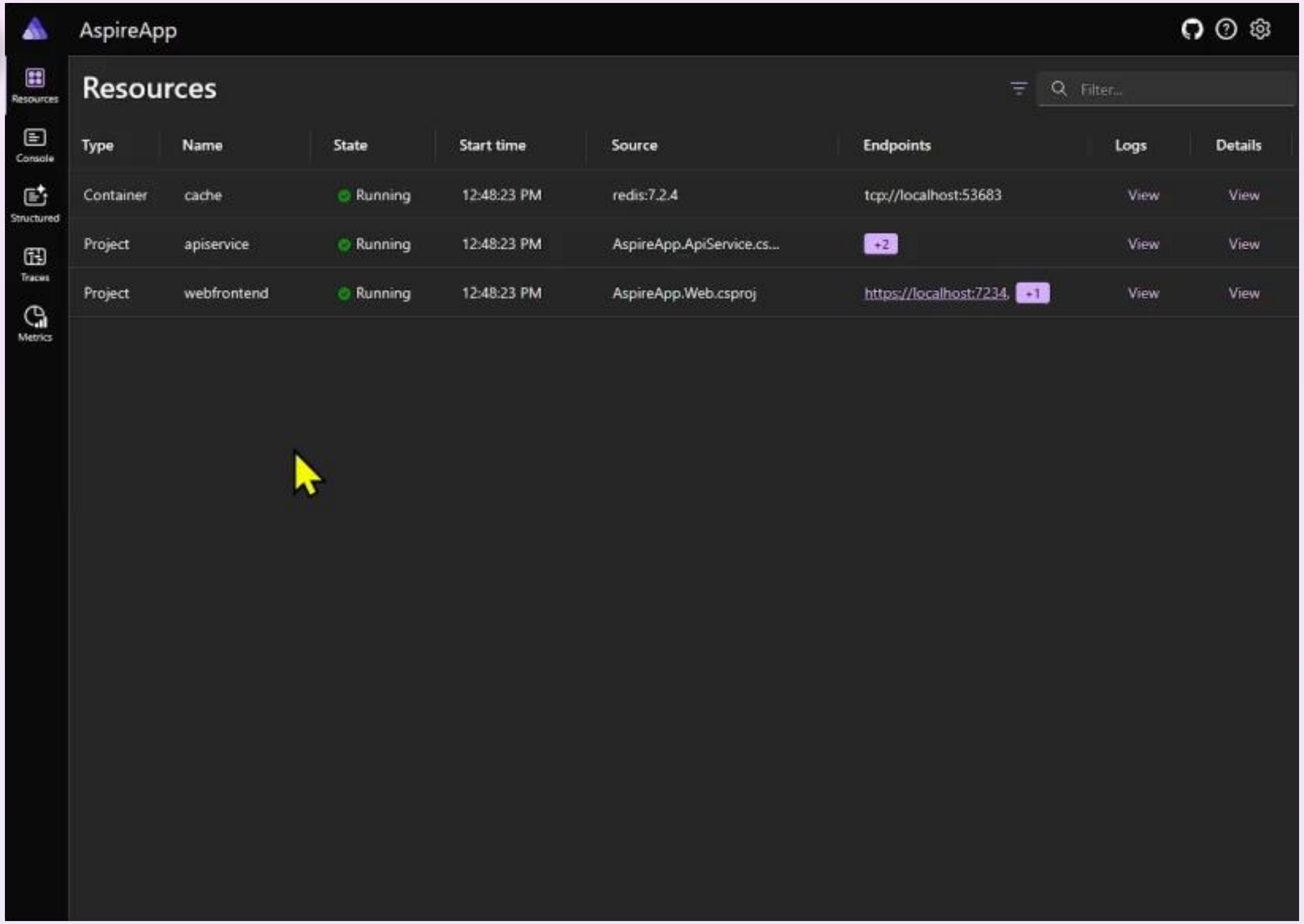
Integrations

Deployment

.NET Aspire

Developer Dashboard

- Structured Logs
- Metrics
- Distributed Traces
- Dependencies



Resources

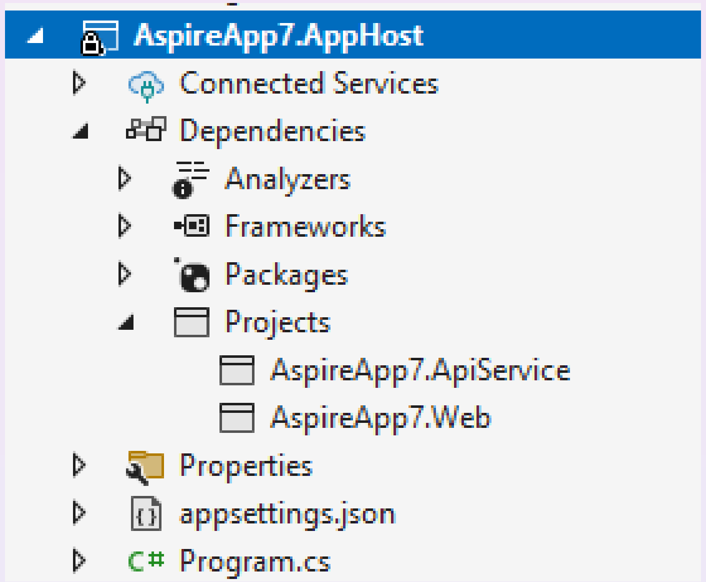
Type	Name	State	Start time	Source	Endpoints	Logs	Details
Container	cache	● Running	12:48:23 PM	redis:7.2.4	tcp://localhost:53683	View	View
Project	apiservice	● Running	12:48:23 PM	AspireApp.ApiService.cs...	+2	View	View
Project	webfrontend	● Running	12:48:23 PM	AspireApp.Web.csproj	https://localhost:7234, +1	View	View

Integrating the Developer Dashboard

Standalone

```
docker run --rm -it `
    -p 18888:18888 `
    -p 4317:18889 -d `
    --name aspire-dashboard `
    mcr.microsoft.com/dotnet/aspire-dashboard:9.0.0
```

.NET Aspire App Host



```
◢ AspireApp7.AppHost
    ▷ Connected Services
    ◢ Dependencies
        ▷ Analyzers
        ▷ Frameworks
        ▷ Packages
        ◢ Projects
            AspireApp7.ApiService
            AspireApp7.Web
    ▷ Properties
    ▷ appsettings.json
    ▷ C# Program.cs
```

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

Smart Defaults

Developer Dashboard

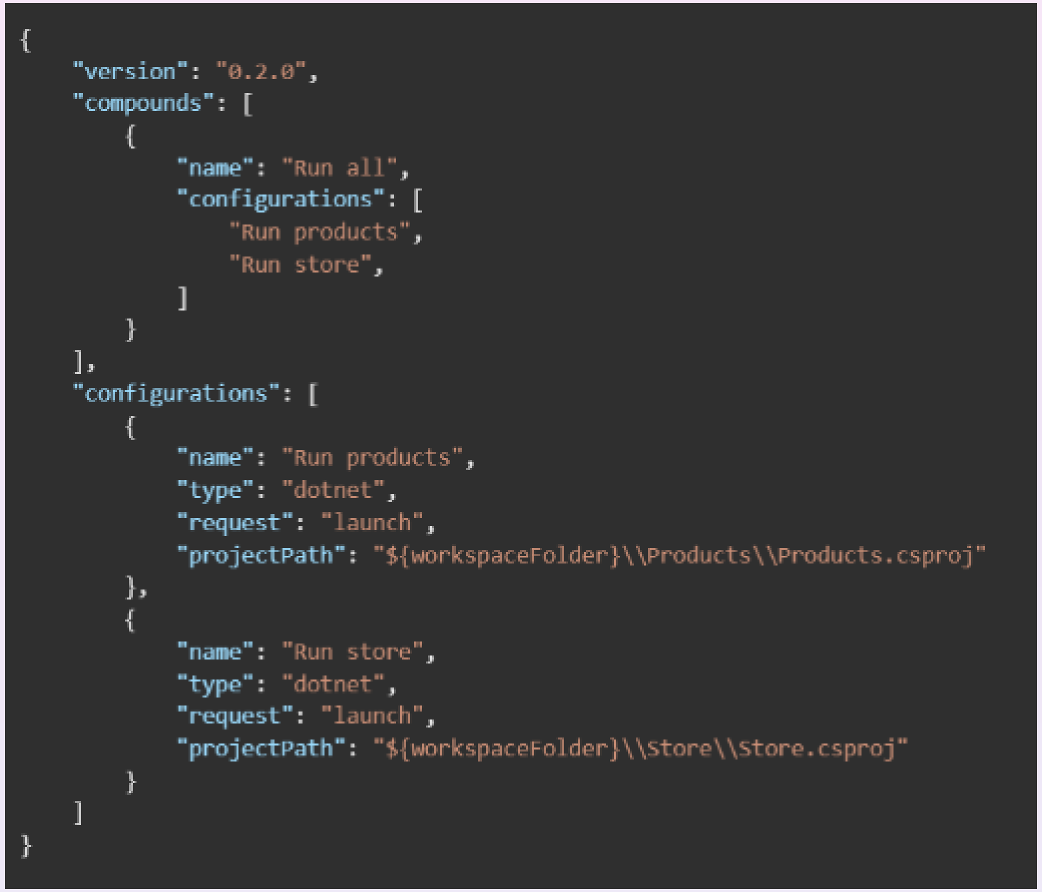
Orchestration

Service Discovery

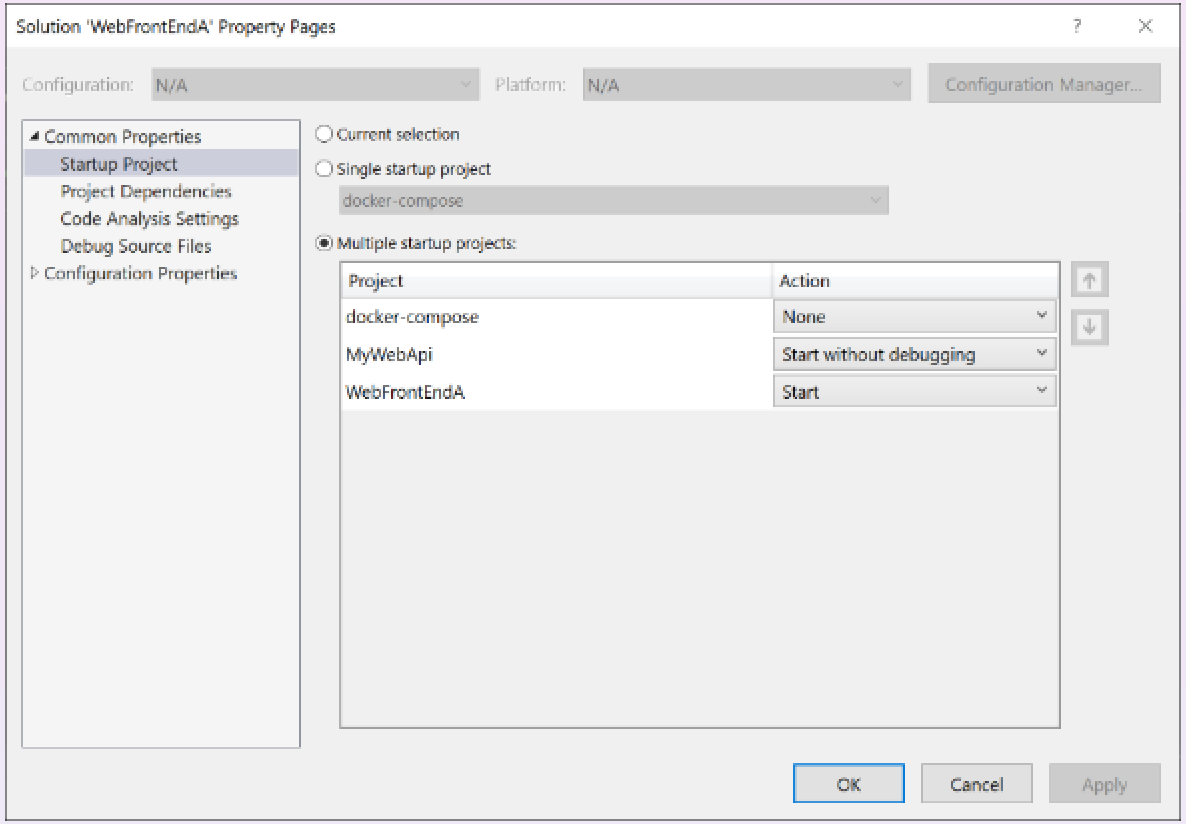
Integrations

Deployment

Orchestration Before



```
{
    "version": "0.2.0",
    "compounds": [
        {
            "name": "Run all",
            "configurations": [
                "Run products",
                "Run store",
            ]
        }
    ],
    "configurations": [
        {
            "name": "Run products",
            "type": "dotnet",
            "request": "launch",
            "projectPath": "${workspaceFolder}\\Products\\Products.csproj"
        },
        {
            "name": "Run store",
            "type": "dotnet",
            "request": "launch",
            "projectPath": "${workspaceFolder}\\Store\\Store.csproj"
        }
    ]
}
```



Solution 'WebFrontEndA' Property Pages

Configuration: N/A

Platform: N/A

○ Current selection

○ Single startup project

docker-compose

● Multiple startup projects:

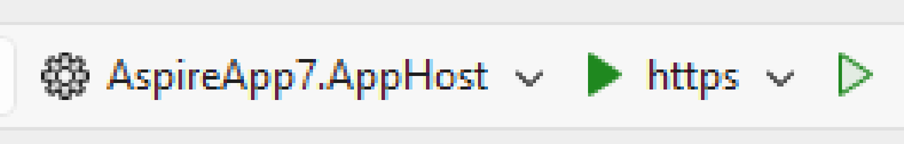
Project	Action
docker-compose	None
MyWebApi	Start without debugging
WebFrontEndA	Start

.NET Aspire

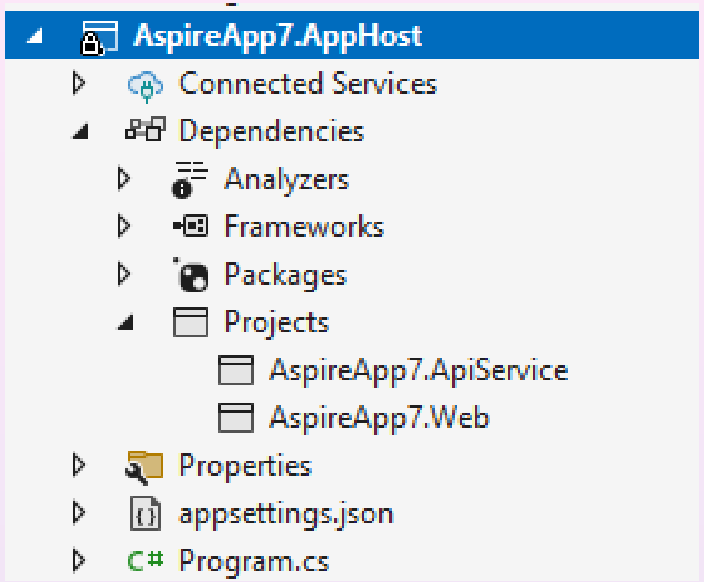
Orchestration

```
var builder = DistributedApplication.CreateBuilder(args);

builder.AddProject<Projects.ApiService>("apiservice");

builder.AddProject<Projects.Web>("webfrontend");

builder.Build().Run();
```

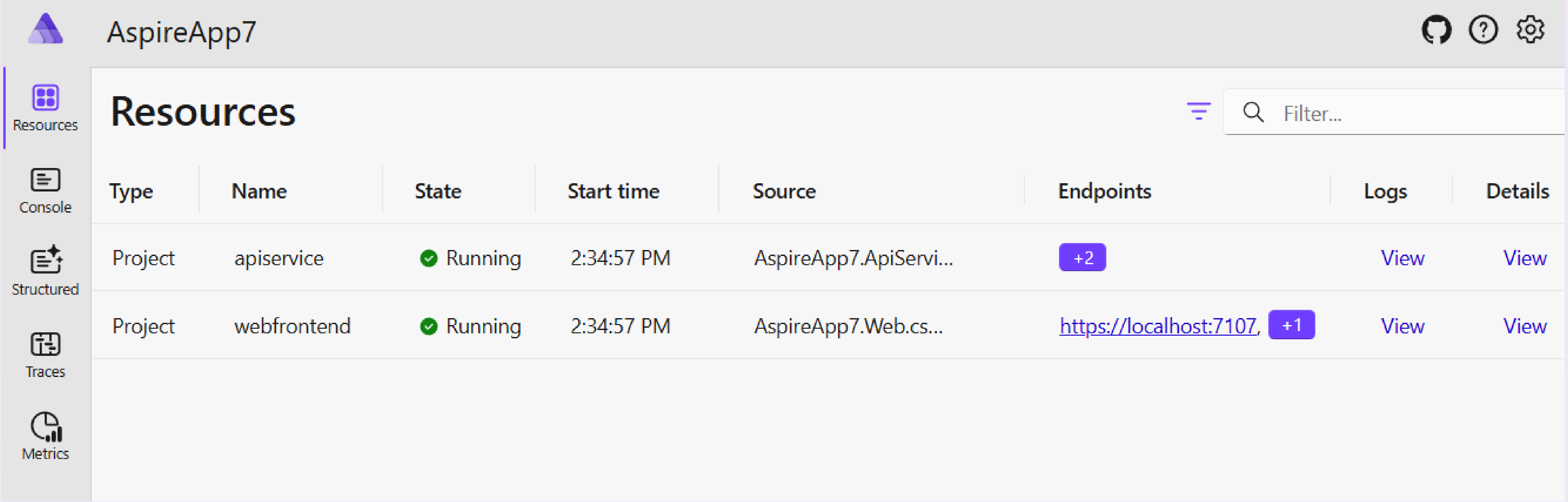
After with .NET Aspire



AspireApp7.AppHost ⌄ ▶ https ⌄ ▷



- AspireApp7.AppHost
  - Connected Services
  - Dependencies
    - Analyzers
    - Frameworks
    - Packages
    - Projects
      - AspireApp7.ApiService
      - AspireApp7.Web
  - Properties
  - appsettings.json
  - Program.cs



AspireApp7

Resources

Type	Name	State	Start time	Source	Endpoints	Logs	Details
Project	apiservice	● Running	2:34:57 PM	AspireApp7.ApiServi...	+2	View	View
Project	webfrontend	● Running	2:34:57 PM	AspireApp7.Web.cs...	https://localhost:7107, +1	View	View

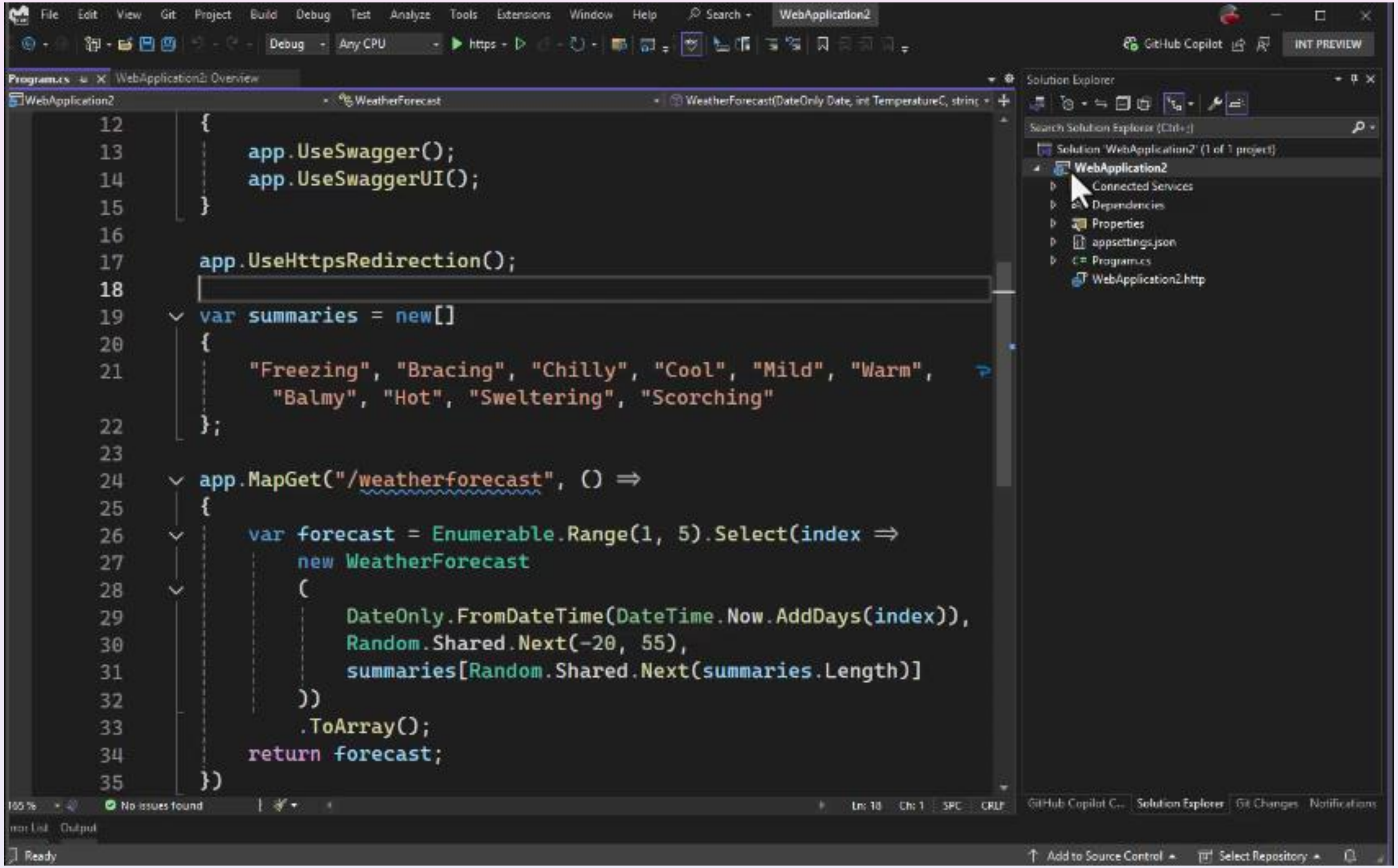
DEMO

.NET Aspire - Orchestration

Q&A

.NET Aspire - Orchestration

Visual Studio Tooling for .NET Aspire



```
    {
        app.UseSwagger();
        app.UseSwaggerUI();
    }

app.UseHttpsRedirection();

var summaries = new[]
{
    "Freezing", "Bracing", "Chilly", "Cool", "Mild", "Warm",
      "Balmy", "Hot", "Sweltering", "Scorching"
};

app.MapGet("/weatherforecast", () =>
{
    var forecast = Enumerable.Range(1, 5).Select(index =>
        new WeatherForecast
        (
            DateOnly.FromDateTime(DateTime.Now.AddDays(index)),
            Random.Shared.Next(-20, 55),
            summaries[Random.Shared.Next(summaries.Length)]
        ))
        .ToArray();
    return forecast;
})
```

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

Smart Defaults

Developer Dashboard

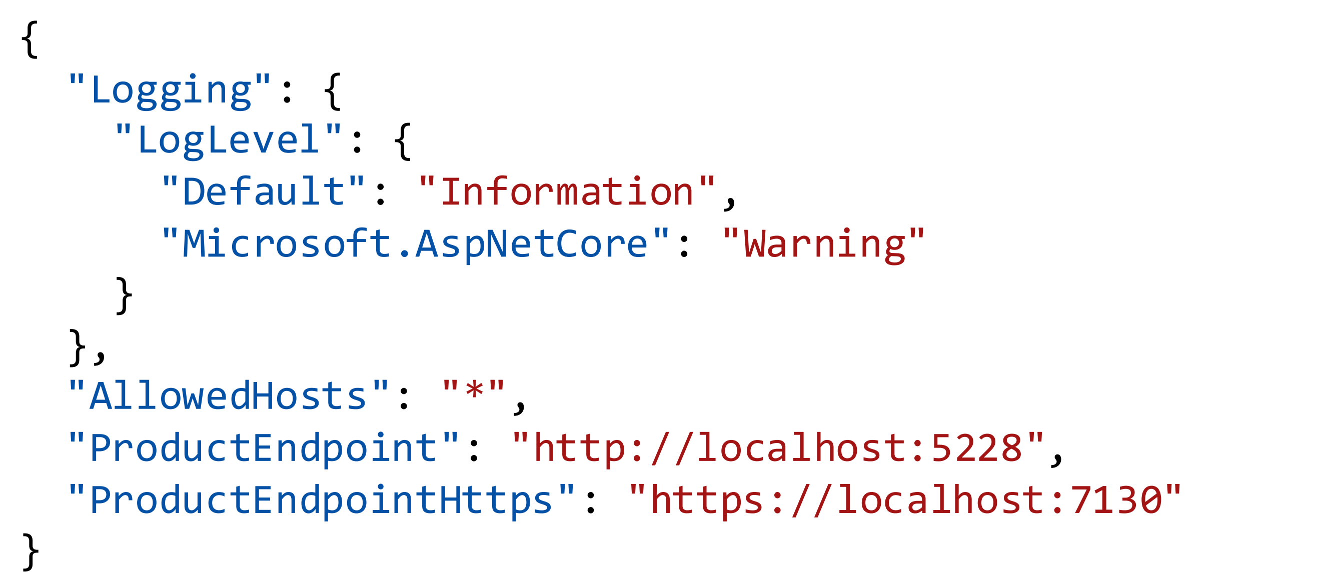
Orchestration

Service Discovery

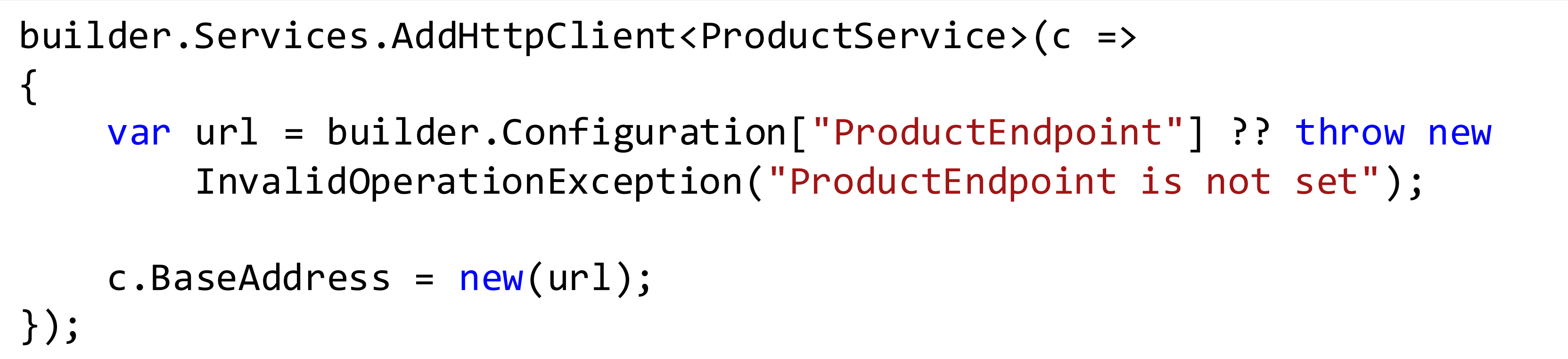
Integrations

Deployment

Service Discovery Before



```json
{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*",
  "ProductEndpoint": "http://localhost:5228",
  "ProductEndpointHttps": "https://localhost:7130"
}
```



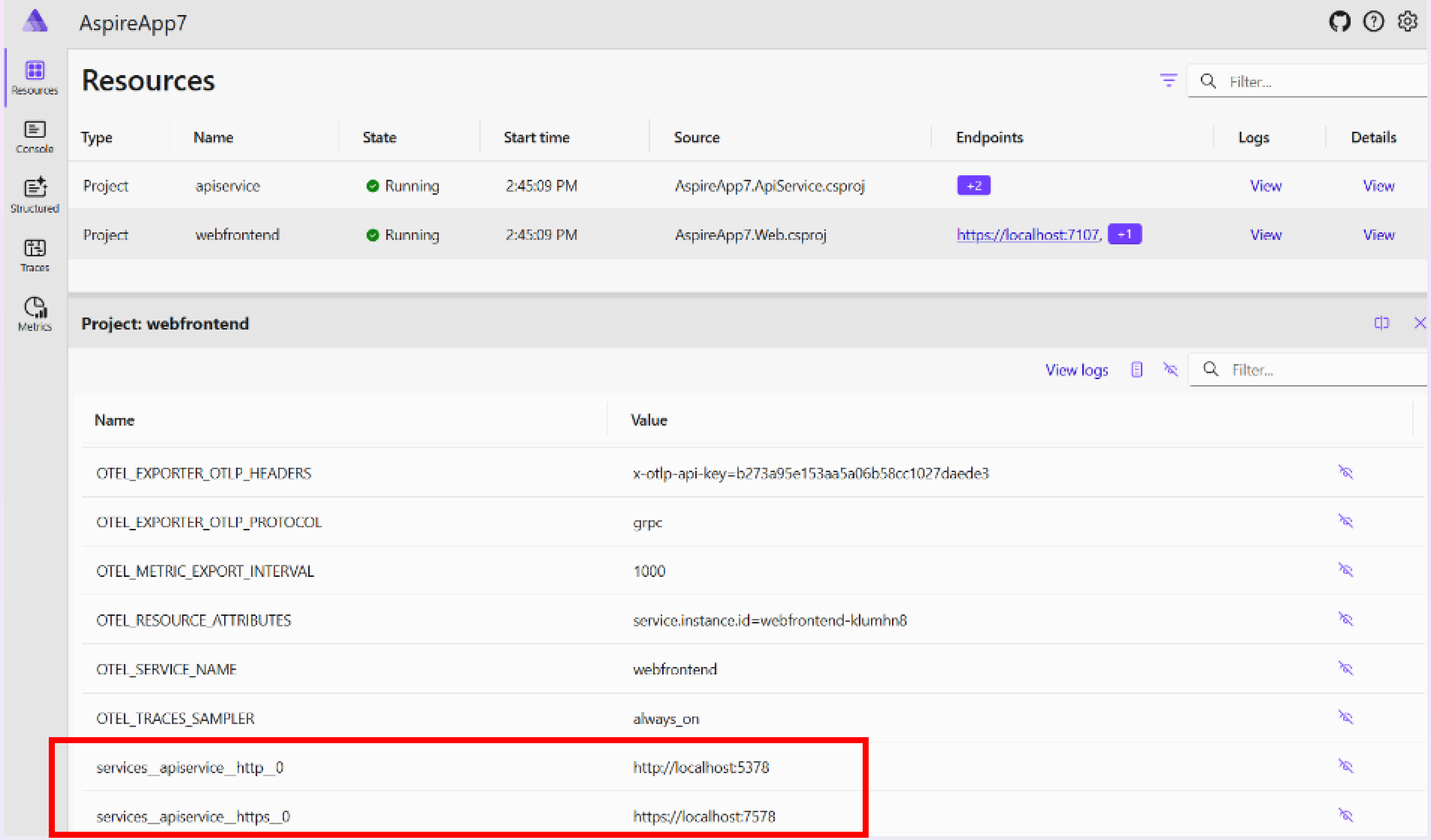
```csharp
builder.Services.AddHttpClient<ProductService>(c =>
{
    var url = builder.Configuration["ProductEndpoint"] ?? throw new
        InvalidOperationException("ProductEndpoint is not set");

    c.BaseAddress = new(url);
});
```

Service Discovery

```csharp
var builder = DistributedApplication.CreateBuilder(args);


var apiService = builder.AddProject<Projects.ApiService>("apiservice");

builder.AddProject<Projects.Web>("webfrontend")
       .WithReference(apiService);

builder.Build().Run();
```

```csharp
builder.Services.AddHttpClient<WeatherApiClient>(client =>
    {
        client.BaseAddress = new("https+http://apiservice");
    });
```

There are still connection strings 😀



Resources

Type	Name	State	Start time	Source	Endpoints	Logs	Details
Project	apiservice	● Running	2:45:09 PM	AspireApp7.ApiService.csproj	+2	View	View
Project	webfrontend	● Running	2:45:09 PM	AspireApp7.Web.csproj	https://localhost:7107, +1	View	View

Project: webfrontend

Name	Value
OTEL_EXPORTER_OTLP_HEADERS	x-otlp-api-key=b273a95e153aa5a06b58cc1027daede3
OTEL_EXPORTER_OTLP_PROTOCOL	grpc
OTEL_METRIC_EXPORT_INTERVAL	1000
OTEL_RESOURCE_ATTRIBUTES	service.instance.id=webfrontend-klumhn8
OTEL_SERVICE_NAME	webfrontend
OTEL_TRACES_SAMPLER	always_on
services__apiservice__http__0	http://localhost:5378
services__apiservice__https__0	https://localhost:7578

DEMO

.NET Aspire - Service Discovery

Q&A

.NET Aspire - Service Discovery

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

Smart Defaults

Developer Dashboard

Orchestration

Service Discovery

Integrations

Deployment

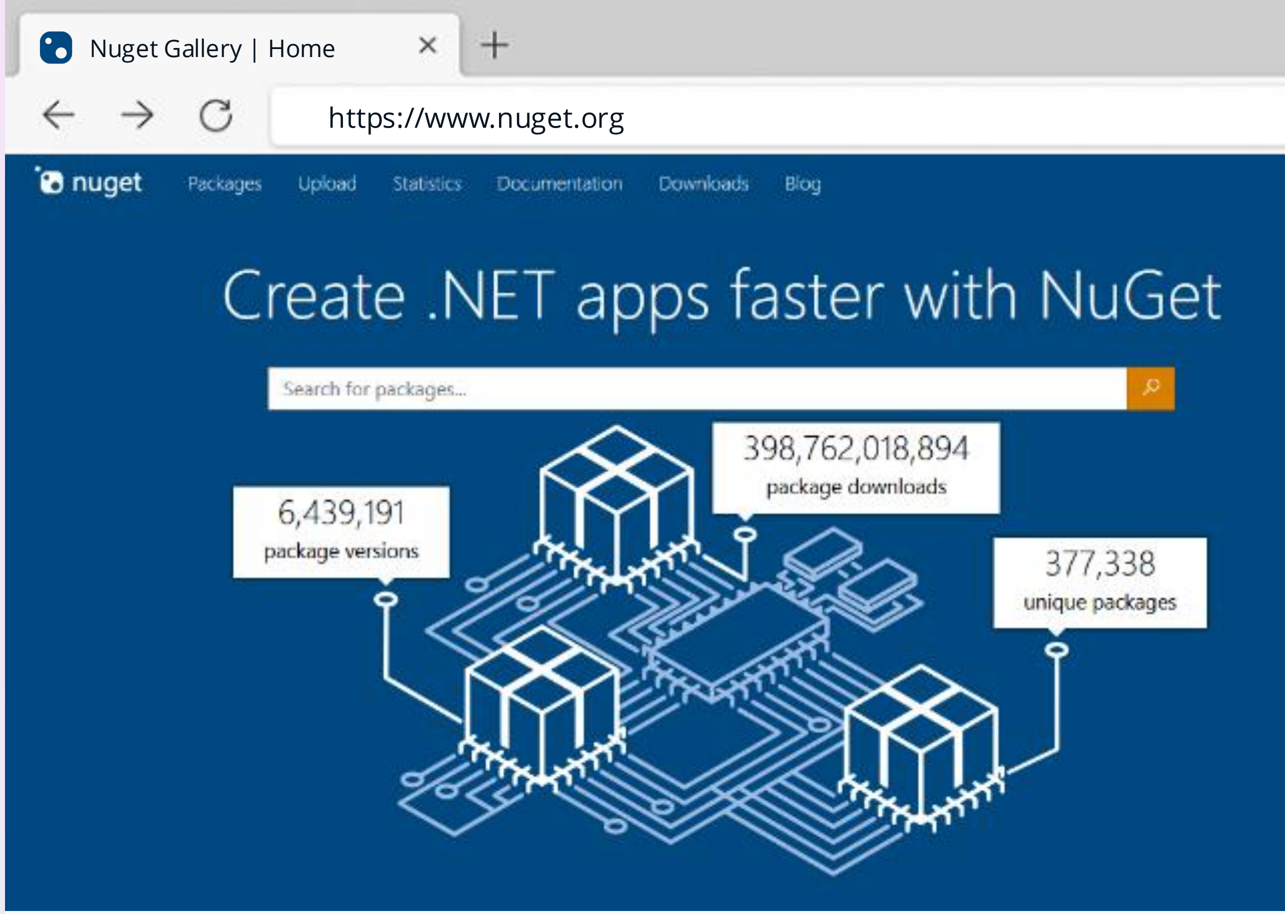
.NET Aspire

Integrations

redis

RabbitMQ

SQL



Create .NET apps faster with NuGet

6,439,191
package versions

398,762,018,894
package downloads

377,338
unique packages

.NET Aspire

Hosting Integration

```
var builder = DistributedApplication.CreateBuilder(args);

var postgres = builder.AddPostgres("postgres").WithPgAdmin();
var db = postgres.AddDatabase("db");

var cache = builder.AddRedis("cache");

var apiService = builder.AddProject<Projects.ApiService>("apiservice")
                        .WithReference(db);

builder.AddProject<Projects.Web>("webfrontend")
       .WithReference(apiService)
       .WithReference(cache);

builder.Build().Run();
```

Automatically will pull down container image and start it in Docker or Podman!

.NET Aspire

Client Integration

```
var builder = WebApplication.CreateBuilder(args);

builder.AddServiceDefaults();
builder.AddRedisOutputCache("cache");

builder.Services.AddRazorComponents()
    .AddInteractiveServerComponents();

var app = builder.Build();

app.UseOutputCache();

app.MapRazorComponents<App>()
    .AddInteractiveServerRenderMode();

app.MapDefaultEndpoints();

app.Run();
```

Configures service with retries, corresponding health checks, logging, and telemetry!!!!

DEMO

.NET Aspire - Integrations

Q&A

.NET Aspire - Integrations

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

Smart Defaults

Developer Dashboard

Orchestration

Service Discovery

Integrations

Deployment

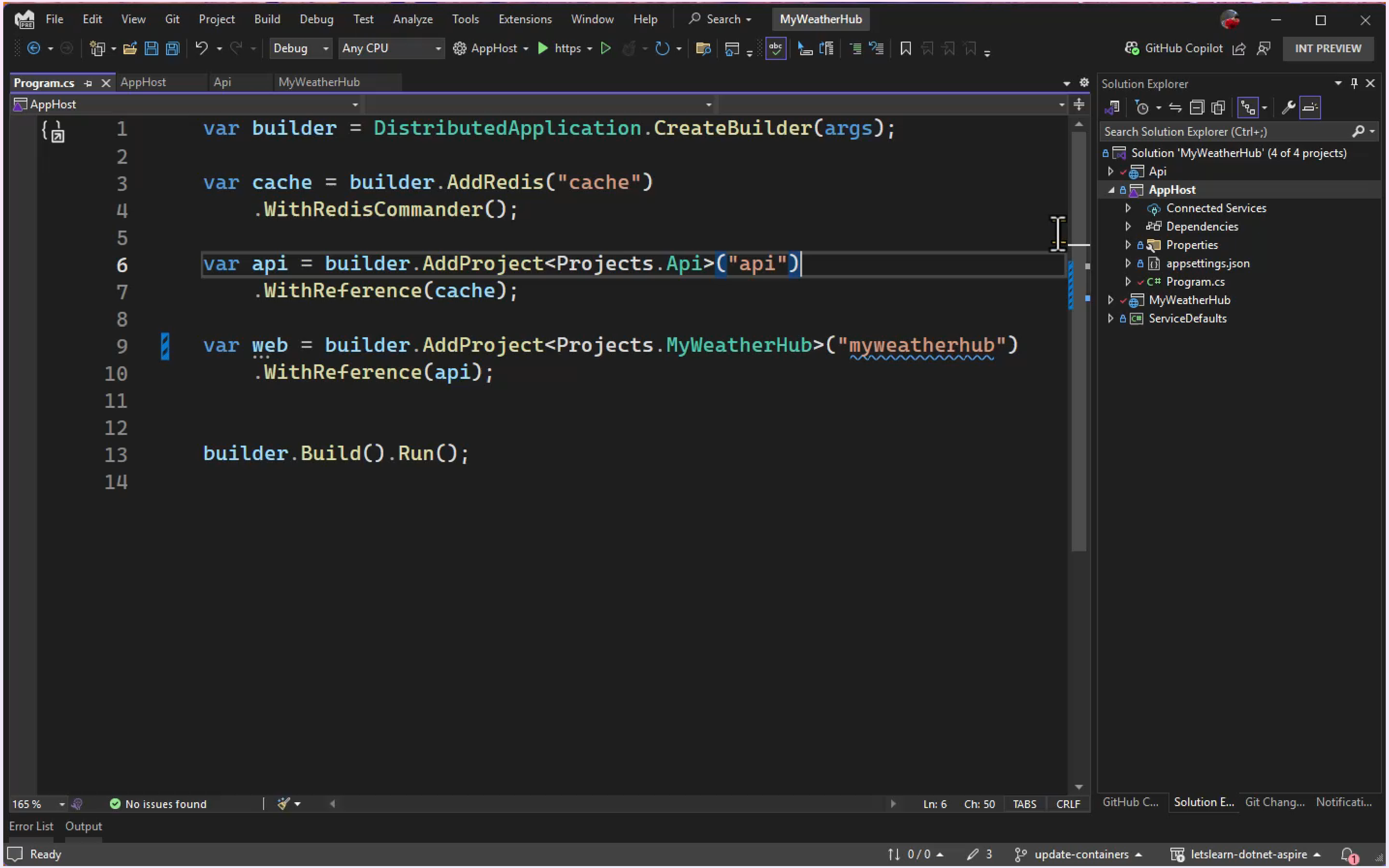
Flexible Integrations & Deployment

How You Do
It Today!

AWS CDK

Azure Dev CLI

Visual Studio



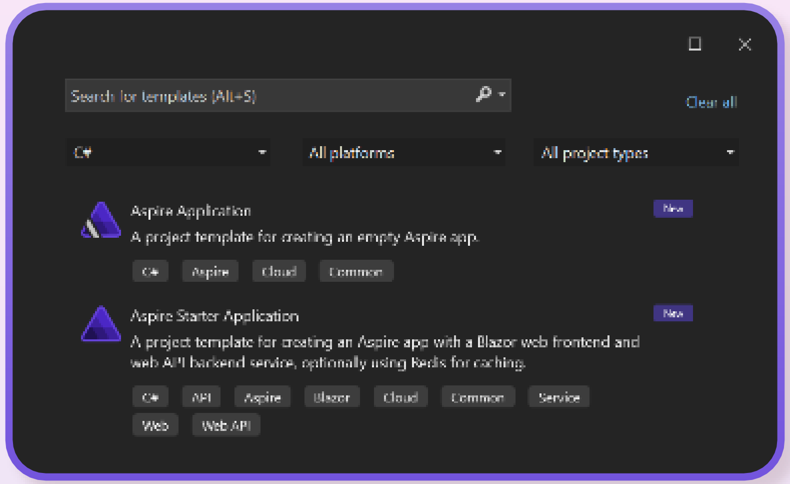
```
var builder = DistributedApplication.CreateBuilder(args);

var cache = builder.AddRedis("cache")
    .WithRedisCommander();

var api = builder.AddProject<Projects.Api>("api")
    .WithReference(cache);

var web = builder.AddProject<Projects.MyWeatherHub>("myweatherhub")
    .WithReference(api);


builder.Build().Run();
```

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

- Smart Defaults
- Developer Dashboard
- Orchestration
- Service Discovery
- Integrations
- Deployment

.NET Aspire



Search for templates (Alt+S)

Clear all

C# | All platforms | All project types

Aspire Application — New

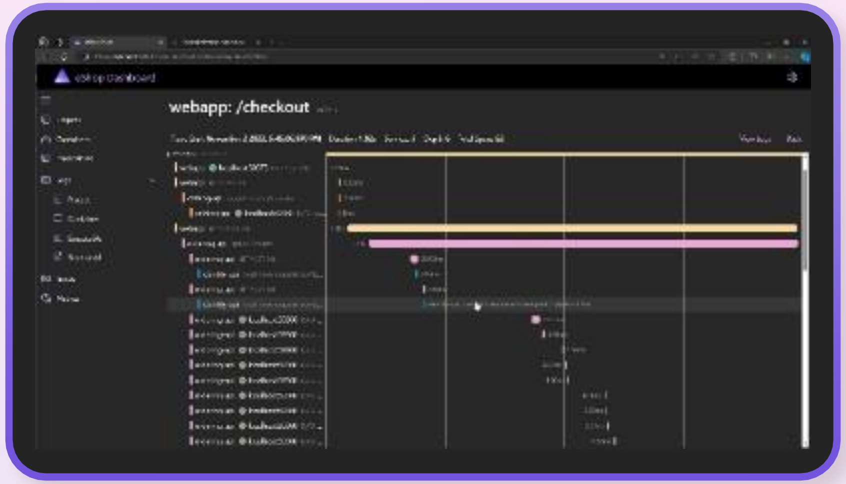
A project template for creating an empty Aspire app.

C# · Aspire · Cloud · Common

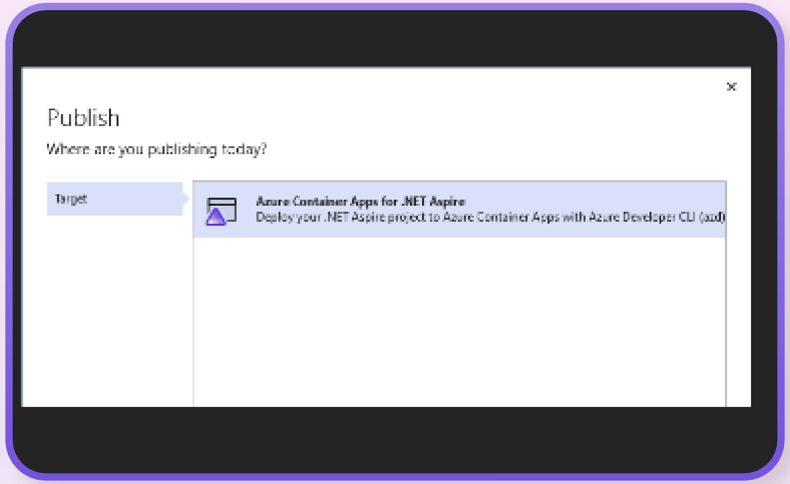
Aspire Starter Application — New

A project template for creating an Aspire app with a Blazor web frontend and web API backend service, optionally using Redis for caching.

C# · API · Aspire · Blazor · Cloud · Common · Service · Web · Web API



webapp: /checkout



Publish

Where are you publishing today?

Target

Azure Container Apps for .NET Aspire
Deploy your .NET Aspire project to Azure Container Apps with Azure Developer CLI (azd)

Easy to get started

SDK Workload

Templates

Integrations

Easy to build

Service discovery

Developer dashboard

Logs, metrics, distributed traces

Easy to deploy

Single command run

App topology in C#

Cloud deployment

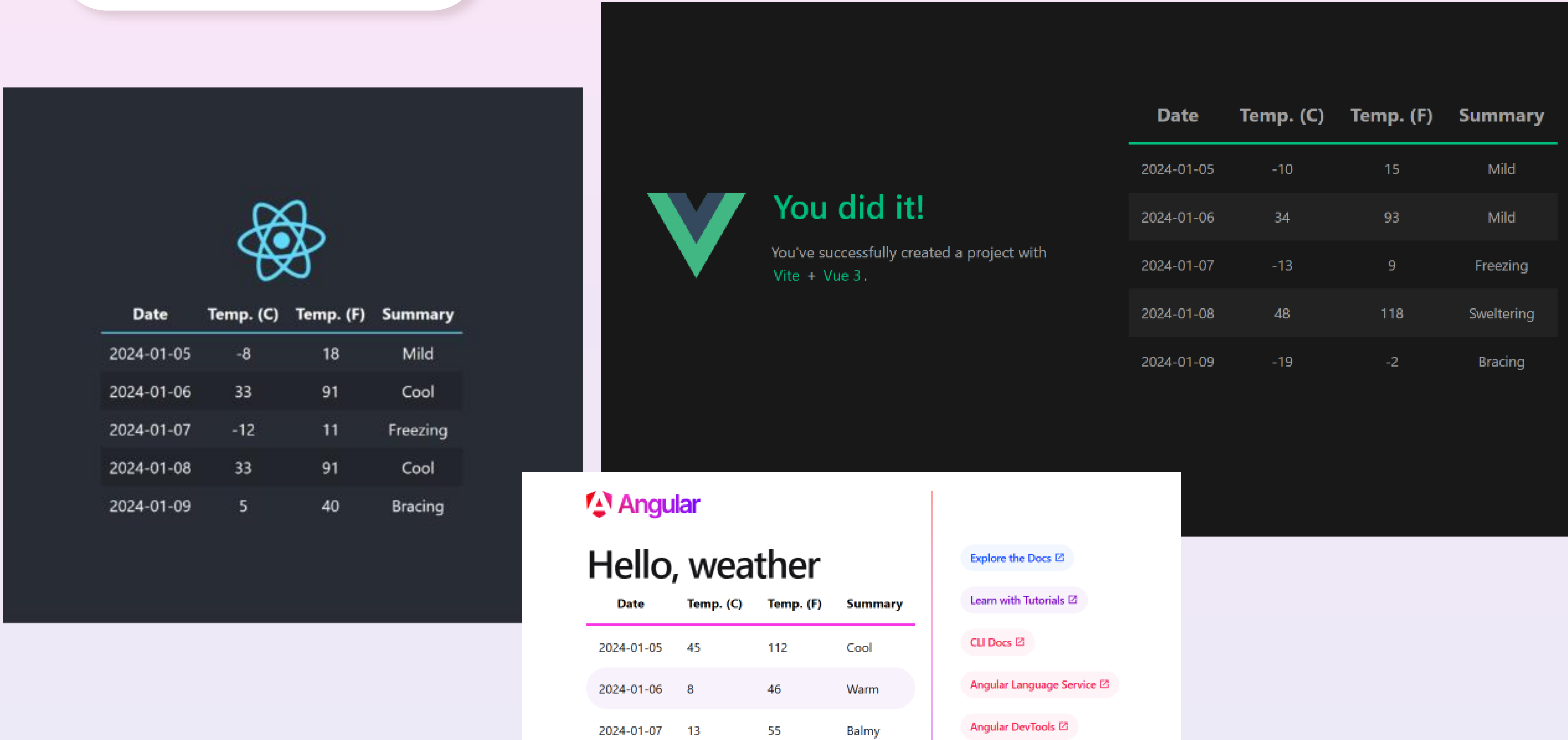
DEMO

.NET Aspire – Starter Project

Q&A

.NET Aspire – Starter Project

Add to any app



Date	Temp. (C)	Temp. (F)	Summary
2024-01-05	-8	18	Mild
2024-01-06	33	91	Cool
2024-01-07	-12	11	Freezing
2024-01-08	33	91	Cool
2024-01-09	5	40	Bracing

You did it!

You've successfully created a project with Vite + Vue 3.

Date	Temp. (C)	Temp. (F)	Summary
2024-01-05	-10	15	Mild
2024-01-06	34	93	Mild
2024-01-07	-13	9	Freezing
2024-01-08	48	118	Sweltering
2024-01-09	-19	-2	Bracing

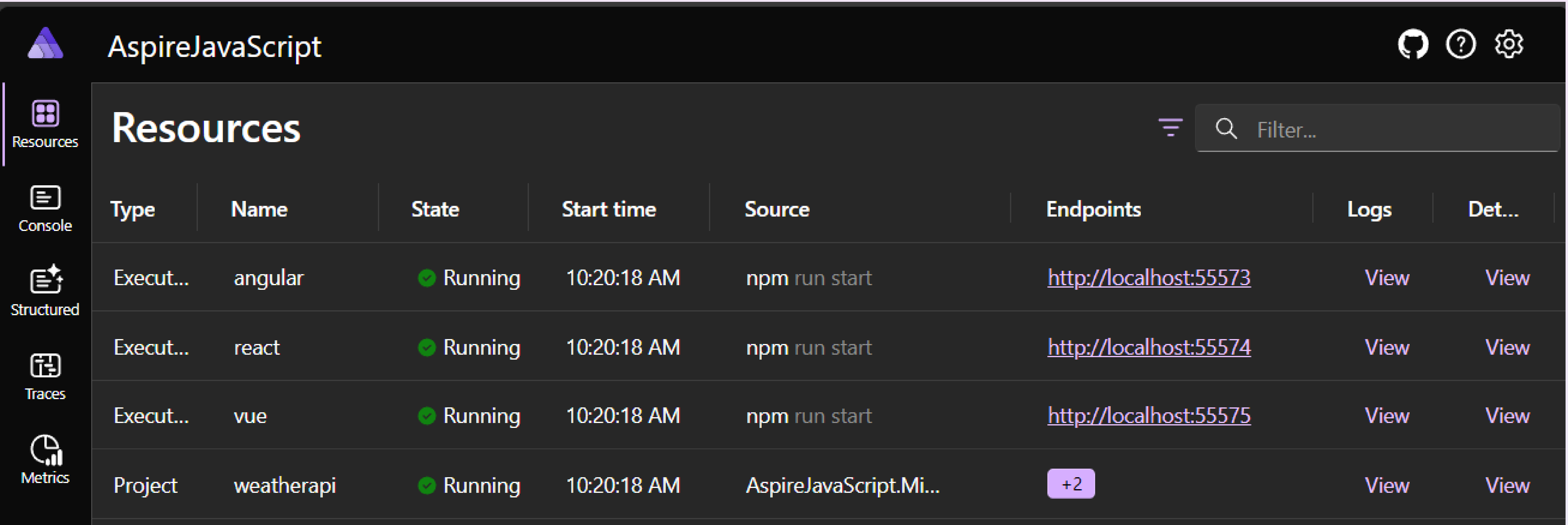
Angular

Hello, weather

Date	Temp. (C)	Temp. (F)	Summary
2024-01-05	45	112	Cool
2024-01-06	8	46	Warm
2024-01-07	13	55	Balmy

.NET Aspire for Everyone!

```
builder.AddNpmApp("angular",
                  "../AspireJavaScript.Angular")
    .WithReference(apiService)
    .WithHttpEndpoint(env: "PORT")
    .WithExternalHttpEndpoints()
    .PublishAsDockerFile();
```



Resources

Type	Name	State	Start time	Source	Endpoints	Logs	Det...
Execut...	angular	Running	10:20:18 AM	npm run start	http://localhost:55573	View	View
Execut...	react	Running	10:20:18 AM	npm run start	http://localhost:55574	View	View
Execut...	vue	Running	10:20:18 AM	npm run start	http://localhost:55575	View	View
Project	weatherapi	Running	10:20:18 AM	AspireJavaScript.Mi...	+2	View	View

.NET Aspire

A cloud ready stack for building observable, production ready, distributed applications

- Smart Defaults
- Developer Dashboard
- Orchestration
- Service Discovery
- Integrations
- Deployment

.NET Aspire

Learning Resources

aka.ms/letslearn/dotnet/aspire

.NET Aspire Videos

aka.ms/aspire/videos

Documentation

aka.ms/dotnet-aspire

Engage with team on GitHub

github.com/dotnet/aspire

Presenter Name
Company Name

𝕏 /@Account
▶ /@Account