



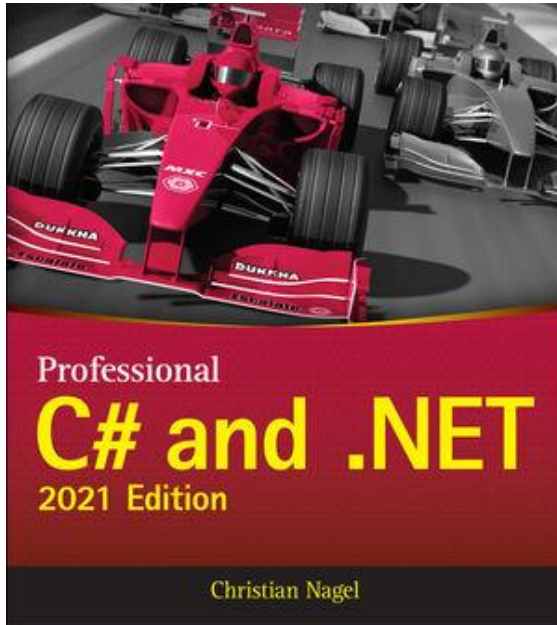
# Full Stack Development with ASP.NET Core, Microsoft Azure and .NET MAUI

Christian Nagel

Sebastian Szvetecz

<https://www.cinnovation.com>

# About us



- **Christian Nagel**

- Software architect
- Software developer
- Trainer
- Book author

- **Sebastian Szvetecz**

- Software developer
- Trainer



# Full-stack development with .NET

- A tour around
  - Backend with EF Core, ASP.NET Core
  - Docker
  - Azure Container Apps, Azure Cosmos DB
  - Frontend with .NET MAUI
  - MVVM
  - CI/CD
- Let's play!

# A challenge!

- In just a few hours!
- We hope to succeed!
- We depend on many different services
- There's a backup plan in case something goes wrong...
- Keep your fingers crossed!





# Questions

---

- If you have questions ask!
- Maybe not an answer right away → Whiteboard
- The more questions you have, we might be too slow with coding – this is ok!!!
- We are here for you – ask questions!

# What do you need?

---

GitHub Account

---

Visual Studio 2022

---

Workloads ASP.NET, Azure, .NET Multi-Platform

---

Docker Desktop

---

Microsoft Azure subscription

---

Macinthecloud (if you use iOS)





# Master Mind!

- Rules
- Start a game
- Set a move

# Part 1 – EF Core



# EF Core

- Object-relational mapper (O/RM)
- Enables working with a database using .NET objects
- Eliminates (most) data-access code
- Many database engines (MSSQL, PostgreSQL, SQLite, Cosmos, ...)

# Using EF Core

- Define a model
- Context for Dependency Injection
- Use the Azure Cosmos provider

# Creating a Nuget Package

- dotnet pack
- Publish to Azure Artifacts using GitHub Actions
- Versioning with
  - Prefix, suffix, build action
  - Overridden with Version



# Bicep

- DSL – Domain Specific Language
- Declarative syntax to deploy Azure resources
- All Azure resources supported
- Simple syntax
- Integration



# Azure Cosmos DB

---

- Distributed NoSQL and relational database
- APIS
  - NoSQL (DocumentDb)
  - PostgreSQL
  - MongoDB
  - Cassandra
  - Table
  - Apache Gremlin



# What you've seen...

- Azure Cosmos DB
- Creating resources with bicep scripts
- Creating NuGet packages







## Part 2 – Minimal API

# Library for DTOs

- Reuse between client and service
- C# records are of practical use

# Minimal API

- Less code to write compared to controllers
- Method parameters are injected
- .NET 8: AOT Compilation
- OpenAPI Description

# Logging, Configuration

- What's needed by all applications
  - Dependency Injection
  - Logging
  - Configuration
- Since .NET Core we have a new approach
- Easy, fast, provider-based!
- OpenTelemetry – just configure a provider

# Azure Services

- Azure App Configuration
  - Configuration for multiple services
  - Can be used to differ staging/production environments
  - Azure Key Vault for secrets
  - Supports feature flags
- Application Insights and Azure Log Analytics
  - Monitoring
  - Logging



# Docker

- Why? What are the issues?
- It runs on my machine!!
- Images – Containers – Registries - Docker CLI - Dockerfiles
- .NET 7: Creating an image is built-in with **dotnet publish**



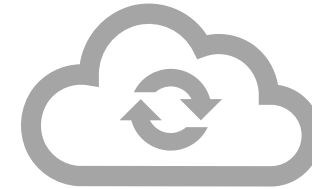


# Azure Container Services



## Azure Container Registry

Registry for your images



## Azure Container Apps

Abstraction of Kubernetes  
Simple Ingress configuration  
Simple Scalability (KEDA)  
Pay per use

# What you've seen

- Minimal API
- Logging, Configuration
- Docker
- Azure App Configuration
- Azure Container Registry
- Azure Container Apps



# Part 3 – .NET MAUI and Client-Libraries



# .NET MAUI

- Platforms
- Successor of Xamarin.Forms



# Client Application Setup

- .NET MAUI Client
  - Windows, Android, iOS
- MVVM Games Library
  - Can be used by all XAML-based client applications
- Games Http Client Library
  - Can be used by all .NET client applications

# Android, iOS

- Pair to Mac
  - Android SDK Manager
  - Device Manager
- 
- dotnet workload search maui



# Sharing a Library

- Multi-Targeting
- Supporting LTS (.NET 6) and STS (.NET 7) Clients

# HttpClient

- Inject the HttpClient
- Use built-in JSON extension methods
- Use streams instead of strings (Large Object Heap)
- System.Text.Json Serializer

# View-Models

- MVVM Pattern
- Usable from all XAML-based clients
- Independence of different platforms with contracts (IDialogService)
- Source Generator with CommunityToolkit.Mvvm for change notifications, commands

# Pages and Views

- XAML (differences to WPF/...)
- Data Templates
- Views
- Converters



# Visual Studio App Center

- UI Tests in the Cloud
- Distribution to testers
- Monitoring

# Take away

- Microsoft Azure Services
- EF Core with Azure Cosmos DB
- ASP.NET Core Minimal API
- Github Actions
- .NET MAUI



Thank you

