# Skybooker

21º

Nevan Alberola

# Inhaltverzeichnis

# 01

## Projekt-Konzept

**Entwicklung eines Microservice basierten Systems zur Buchung und Verwaltung von Flügen, mit API-Gateway, JWT-Authentifizierung und Speicherung in MongoDB & SQL Server.**
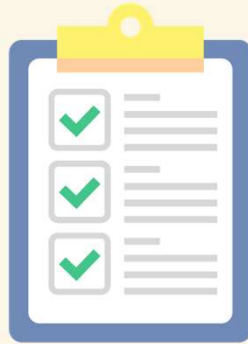
Kernanforderungen

Backend

Datenbank

**Projekt-Konzept**

# Kernanforderungen

- Authentifizierung via JWT.

- API-Gateway zur Weiterleitung.

- CRUD-Operationen für Buchungen.

**Backend**

**Datenbank**

**Projekt-Konzept**

**Kernanforderungen**

**Datenbank**



# Backend

-Microservices mit .NET 8, REST-API, abgesichert mit JWT.

- API-Gateway mit Ocelot.

**01**

**Projekt-Konzept**

**Kernanforderungen**

**Backend**



# Datenbank

-SQL Server für Benutzerdaten,
MongoDB für Buchungsdaten.

# Gantt chart

| Projektarbeit: Skybooker | Verantwortliche | Geplante Stunden | Verarbeite Stunden | 30/04/2025 | 01/05/2025 | 02/05/2025 | 03/05/2025 | 04/05/2025 | 05/05/2025 | 06/05/2025 | 07/05/2025 | 08/05/2025 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Entscheiden** | NA | **2.5** | **3** | | | | | | | | | |
| Wahl der Technologien | NA | 1 | 1.5 | | | | | | | | | |
| NuGet-Pakete testen | NA | 0.5 | 0.5 | | | | | | | | | |
| JWT-Authentifizierung Konzept | NA | 1 | 1 | | | | | | | | | |
| **Realisieren** | NA | **16** | **22.5** | | | | | | | | | |
| AuthService, FlightService, BookService implementieren | NA | 3 | 3.5 | | | | | | | | | |
| SQL Server, MongoDB konfigurieren & anbinden | NA | 6 | 6 | | | | | | | | | |
| Dockerfiles schreiben & Compose-Setup | NA | 4 | 7 | | | | | | | | | |
| Erweiterung auf Level 3: API-Gateway + MessageService-Vorbereitung | NA | 3 | 6 | | | | | | | | | |
| **Kontrolle** | NA | **5** | **6** | | | | | | | | | |
| Postman-Tests der API | NA | 2 | 2.5 | | | | | | | | | |
| Fehlerbehebung & Unitest | NA | 3 | 3.5 | | | | | | | | | |
| **Auswerten** | NA | **10.5** | **12.5** | | | | | | | | | |
| Erstellung der Lerndokumentation | NA | 8 | 9 | | | | | | | | | |
| Präsentationsvorbereitung | NA | 1.5 | 1 | | | | | | | | | |
| **Total Stunden** | | **39** | **47** | | | | | | | | | |

Demo

.....

EXPLORER ···

ngController.cs    {} ocelot.json    {} appsettings.json ...\ApiGateway    Dockerfile ...\BookService    docker-compose.yml ×    ⊟ ···

docker-compose.yml > YAML > {} services > {} apigateway > [ ] ports

OPEN EDITORS

C# Program.cs Services\BookService
C# BookingController.cs Services\BookService\...
{} ocelot.json Gateway\ApiGateway
{} appsettings.json Gateway\ApiGateway
Dockerfile Services\BookService
× docker-compose.yml
Dockerfile Services\AuthService
Dockerfile Services\FlightService
Dockerfile Gateway\ApiGateway

SKYBOOKER
∨ Services
  ∨ AuthService
    > obj
    > Properties
    {} appsettings.Development.json
    {} appsettings.json
    AuthService.csproj
    ⊕ AuthService.http
    Dockerfile
    C# Program.cs
  ∨ BookService
    > bin
    > Controllers
    > Data
    > Migrations
    > Models
    > obj
    > Properties
> SOLUTION EXPLORER

```yaml
 3    services:
24      bookservice:
33        depends_on:
34          - sqlserver
35        networks:
36          - skybooker-network
37
▷ Run Service
38      apigateway:
39        build:
40          context: ./Gateway/ApiGateway
41        container_name: apigateway
42        ports:
43          - "5000:8080"
44        environment:
45          - ASPNETCORE_ENVIRONMENT=Development
46          - ASPNETCORE_URLS=http://+:8080
47        depends_on:
48          - bookservice
49        networks:
50          - skybooker-network
51
```

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS    QUERY RESULTS (PREVIEW)    ⊡ cmd + ∨ ⊟ 🗑 ··· ∧ ×

```
Microsoft Windows [Version 10.0.26100.3915]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Nevan Alberola\source\repos\SkyBooker>
```

# Fazit