Modular Monolith Architecture

MJ Tech

Agenda

00 - Introduction

My background, course overview, objective, and what to expect.

01 - Modular Monoliths: Introduction

What a modular monolith is, and how it’s different from other architectures.

02 - Defining Module Boundaries

Tools and techniques we can use to identify module boundaries.

03 - Module Cross-Cutting Concerns

Importance of cross-cutting concerns for building a modular architecture.

04 - Module Communication

Synchronous and asynchronous communication, tradeoffs.

05 - Authentication & Authorization

External identity providers, token based authentication, and RBAC.

MJ Tech

Agenda

06 - Architecture Enforcement

Three ways how you can enforce the modular monolith architecture.

07 - Reliable Messaging Patterns

How to use the Outbox and Inbox patterns for reliable messaging.

08 - Event-Driven Architecture

Patterns of event-driven architecture and how they fit in a modular monolith.

09 - Testing Modular Monoliths

Importance of testing. Unit testing, integration testing, and CI/CD pipelines.

10 - Microservices: Extracting Modules

Moving to a distributed system and extracting modules from the monolith.

MJ Tech

Agenda

B1 - Modular Monoliths: CQRS With MongoDB

Introducing a NoSQL database into the system, building projections.

My Background

Meet Milan

- I’m a senior software engineer
- Sometimes, a Software Architect
- Microsoft MVP for Developer Technologies
- Building enterprise-grade applications for nearly a decade
- Scaled applications to 100,000+ users
- Enjoy well-built monolith applications

Course Structure

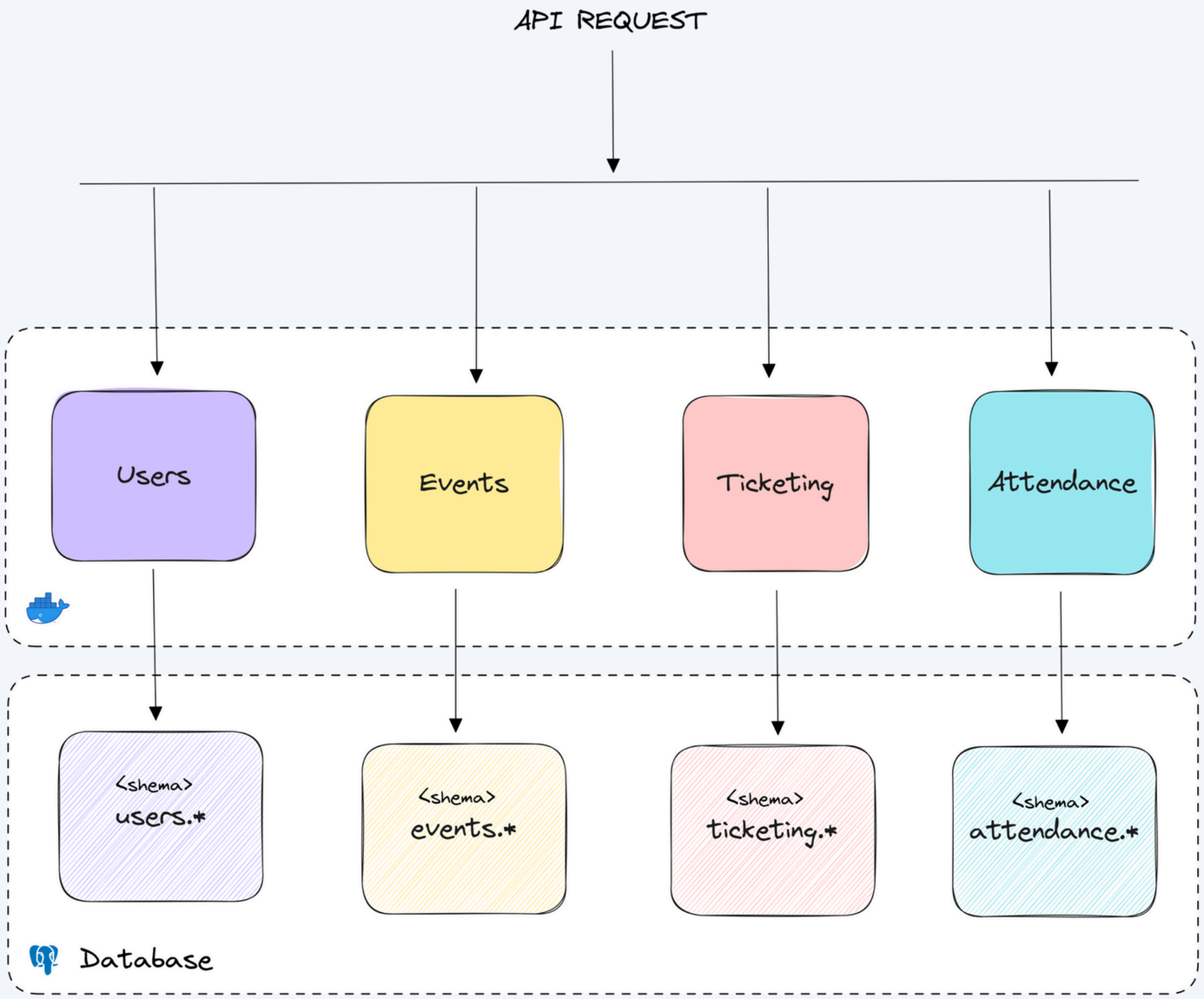
Course Structure

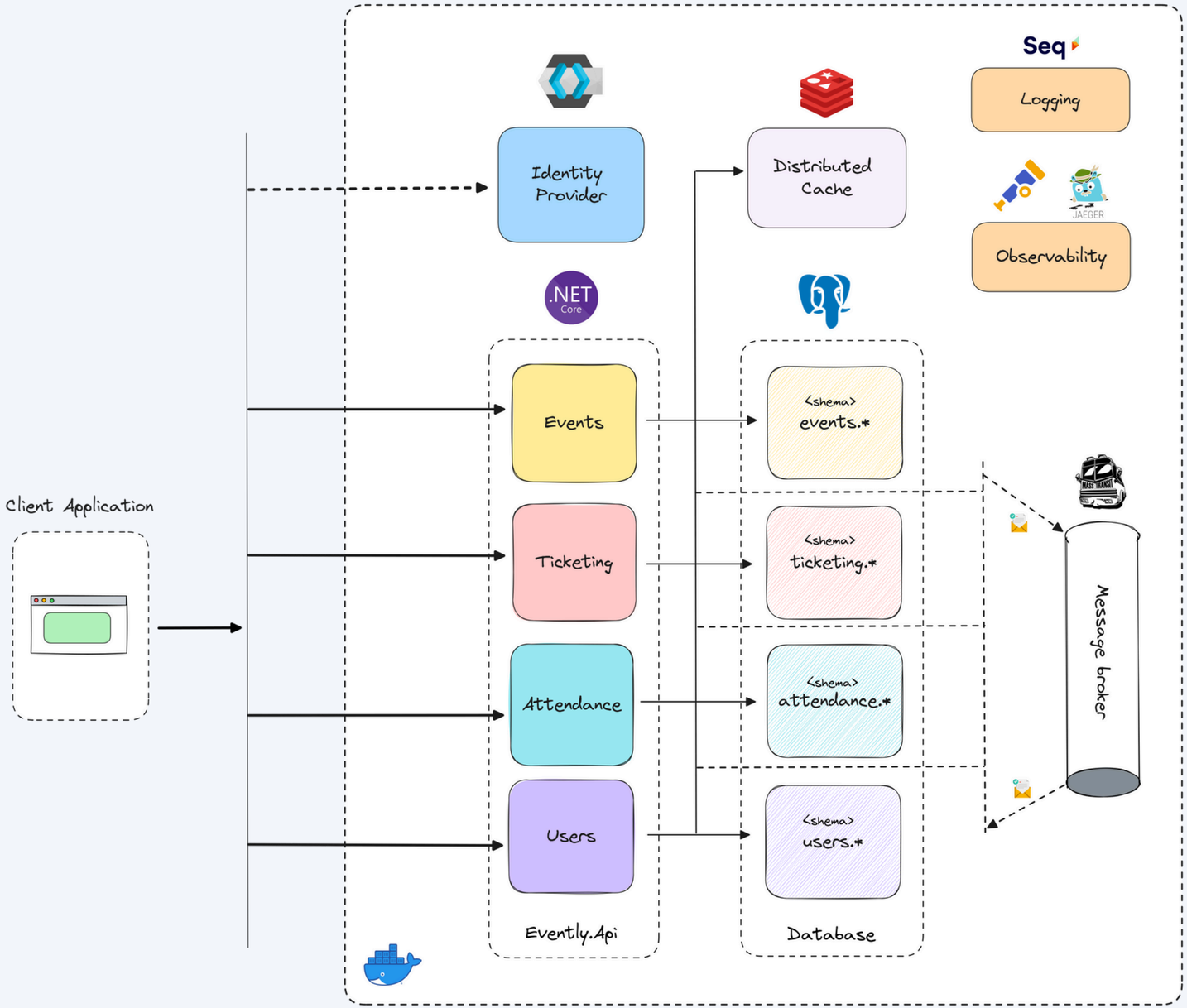
- There are 10 Chapters and a bonus mini-course.
- Each chapter includes 3-9 lessons inside.
- Sometimes lessons are long, so take your time.

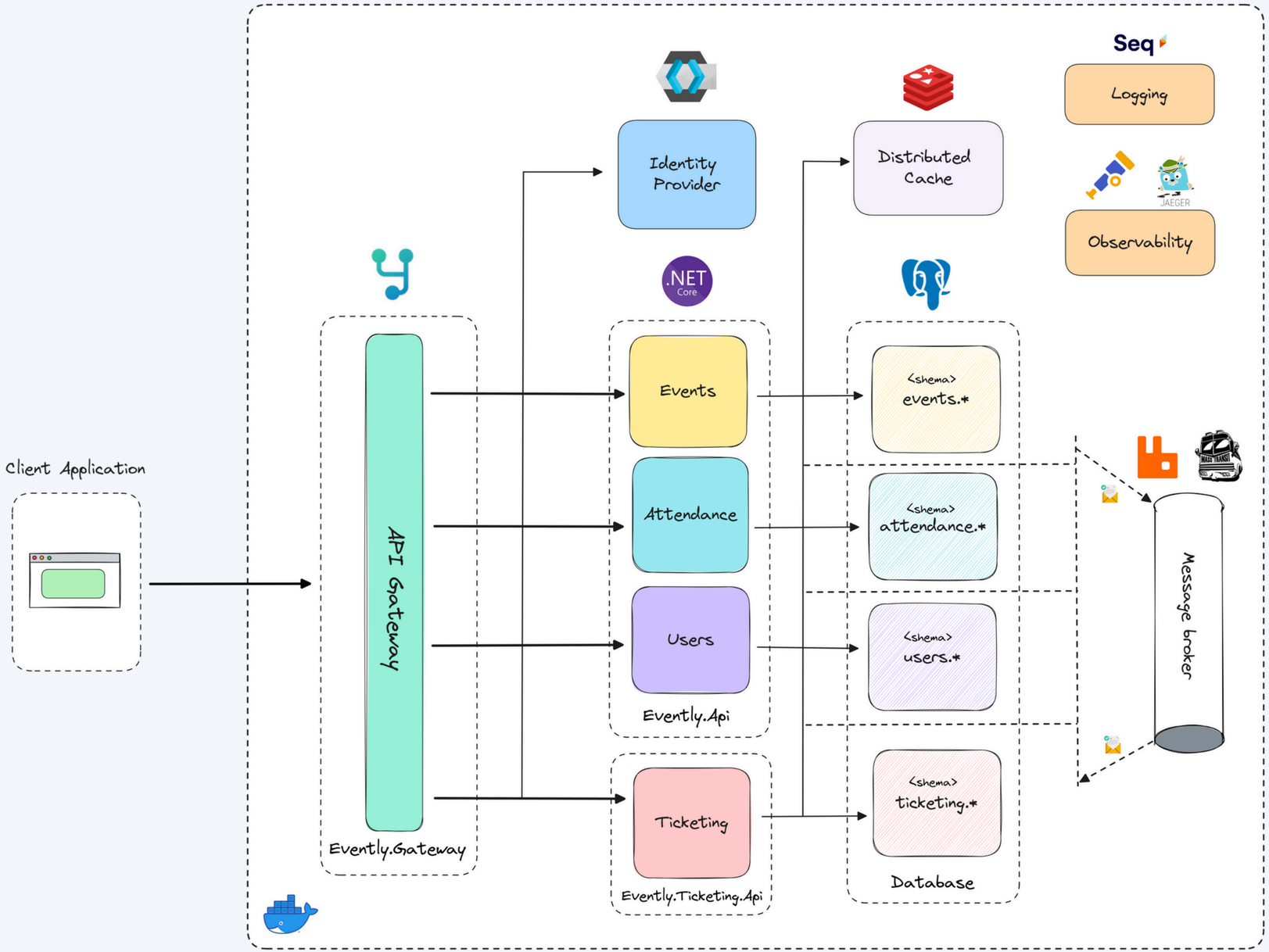
Course Structure

At the beginning of each lesson, I'll:

- Give a basic introduction to the topic.
- Share a simple agenda and the goals.
- Talk about the resources included to help you.
- Say if you should download the updated version of the source code.


API REQUEST
Users
Events
Ticketing
Attendance
<shema>
users.*
<shema>
events.*
<shema>
ticketing.*
<shema>
attendance.*
Database


Seq
Logging
Identity Provider
Distributed Cache
JAEGER
Observability
.NET Core
Events
Ticketing
Attendance
Users
<shema> events.*
<shema> ticketing.*
<shema> attendance.*
<shema> users.*
Message broker
Client Application
Evently.Api
Database


Seq
Logging
Identity Provider
Distributed Cache
JAEGER
Observability
.NET Core
Client Application
API Gateway
Events
Attendance
Users
Evently.Api
<shema> events.*
<shema> attendance.*
<shema> users.*
Message broker
Evently.Gateway
Ticketing
Evently.Ticketing.Api
<shema> ticketing.*
Database

Next:
How To Use This Course

How To Use This Course

Best Approach

- Watch this course from beginning to end.
- Choose to watch the parts that are most relevant to your challenges.

Prerequisites

- Visual Studio (or VS Code)
- .NET 8
- Docker

Technical Tips

- Download the OpenAPI specification or Postman collection.

- Apply database migrations with SQL scripts.

Next:
Modular Monoliths: Introduction

MJ Tech