Clean Architecture

Tomáš Jurásek

Motivation

If you think good architecture is expensive, try bad architecture.

-Brian Foote and Joseph Yoder

The only way to go fast, is to go well.

-Robert C. Martin

.NET COLLEGE

SOLID

1. Single Responsibility Principe (SRP)
2. Open-Closed-Principle (OCP)
3. Liskov Substitution Principle (LSP)
4. Interface Segregation Principle (ISP)
5. Dependency Inversion Principle (DIP)

KISS (Keep It Simple, Stupid!)

1. Short method and class
2. Single responsibility
3. Refactoring
4. Split your problems to smaler

.NET
COLLEGE

Clean Architecture

1. Core
   - Domain Entities != Database Entities
   - UseCases (request, response)
   - Domain Services and Interfaces
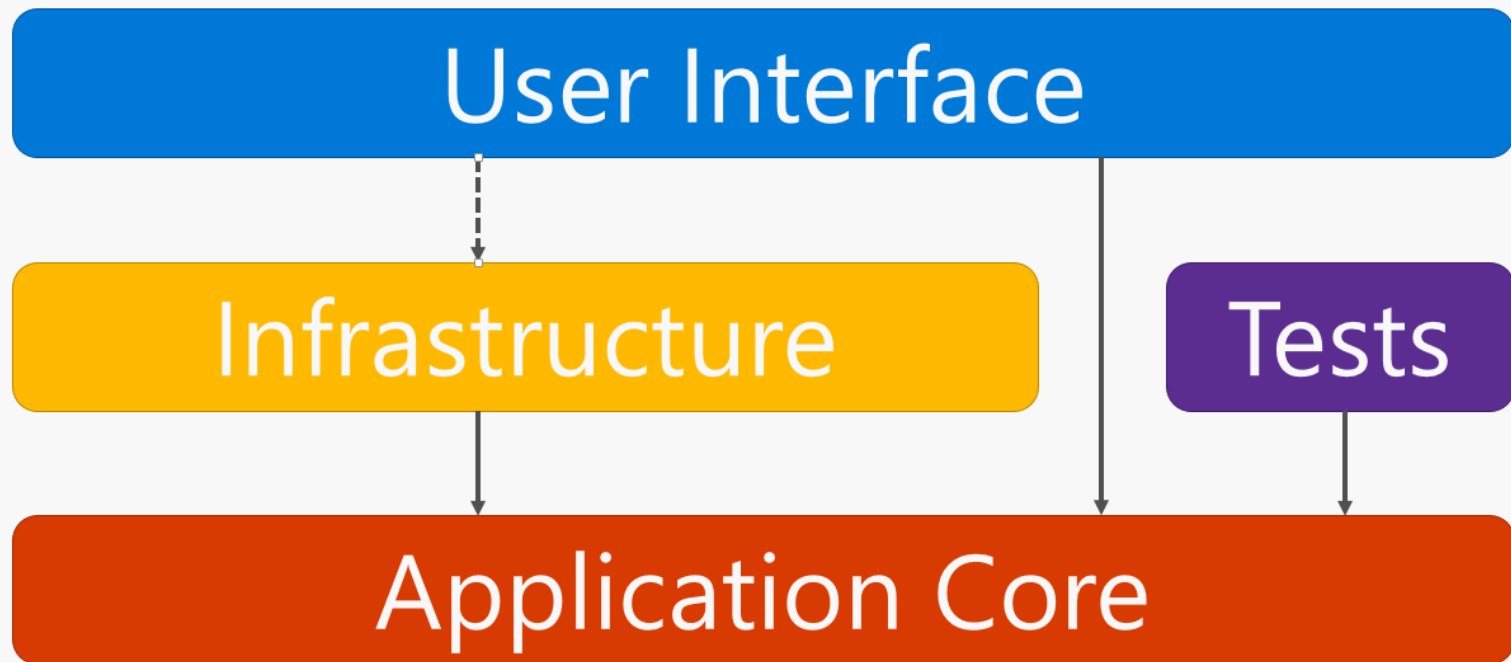2. Infrastructure
   - Third party library services (EF, Redis, SendGrid,IoC,..)
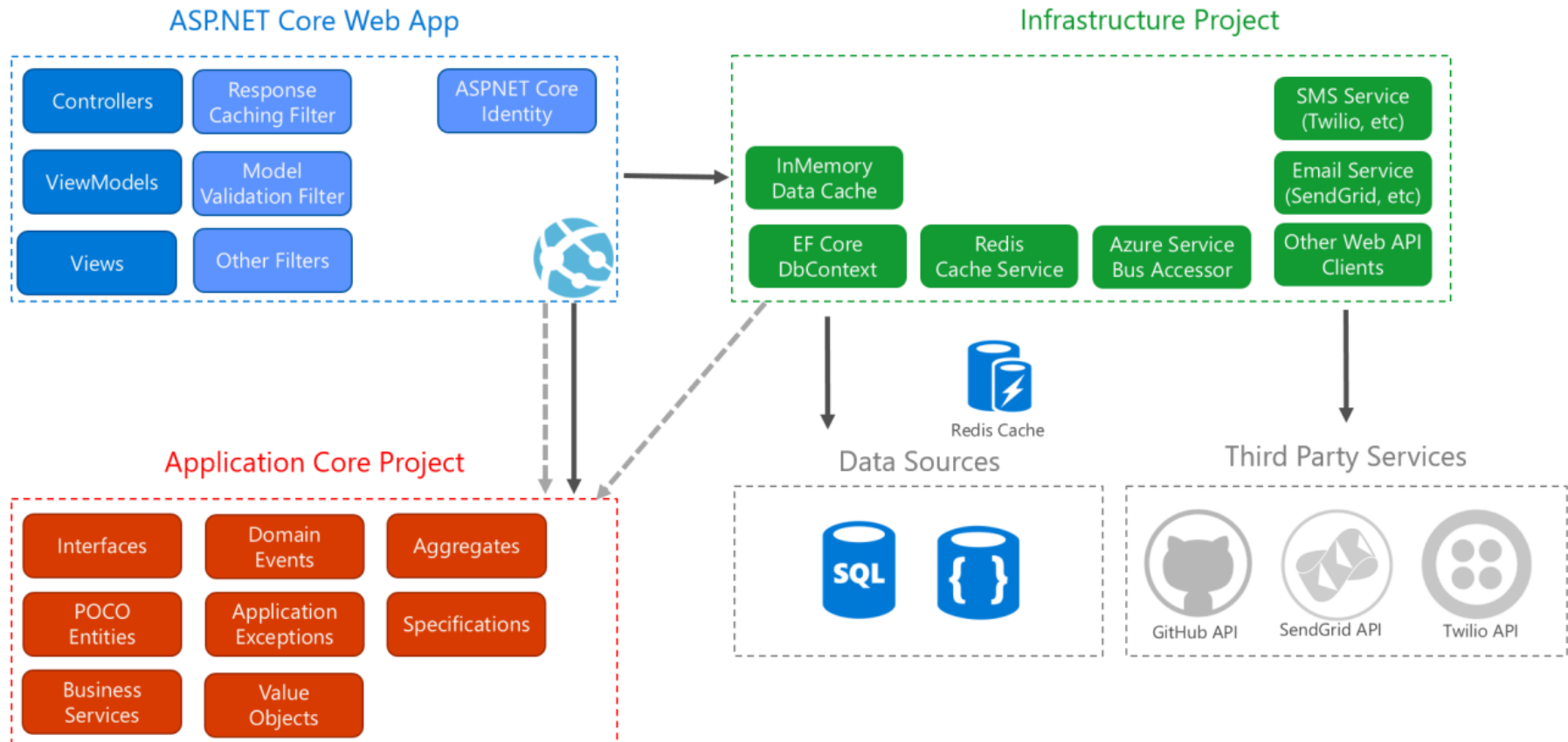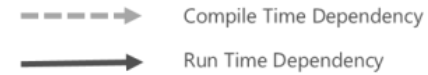3. API
   - Controllers
   - Models
   - Web services (Context handler, web IoC installers,…)

ASP.NET Core Architecture

Compile Time Dependency
Run Time Dependency

ASP.NET Core Web App

Controllers

Response Caching Filter

ASPNET Core Identity

ViewModels

Model Validation Filter

Views

Other Filters

Infrastructure Project

SMS Service (Twilio, etc)

InMemory Data Cache

Email Service (SendGrid, etc)

EF Core DbContext

Redis Cache Service

Azure Service Bus Accessor

Other Web API Clients

Redis Cache

Application Core Project

Interfaces

Domain Events

Aggregates

POCO Entities

Application Exceptions

Specifications

Business Services

Value Objects

Data Sources

SQL

{}

Third Party Services

GitHub API

SendGrid API

Twilio API

.NET COLLEGE

Demo

Links and Books

- https://www.amazon.com/Clean-Architecture-Craftsmans-Software-Structure/dp/0134494164

- http://voho.eu/wiki/princip-solid/

- http://www.robertdresler.cz/2016/10/jednoducha-slozitost-nebo-slozita.html

- https://www.augi.cz/programovani/moderni-programovani-v-c/

- https://github.com/tomasjurasek/CleanArchitecture

.NET
COLLEGE