

Sprint 5 - Dependency Injection, Logging, Refactoring

1. Stellen Sie Ihre gesamte Applikation auf **Dependency Injection** um.
Nutzen Sie dabei die ASP.NET Core eigenen Mechanismen (<https://docs.microsoft.com/en-us/aspnet/core/fundamentals/dependency-injection>) und **Constructor Injection**.

Constructor werden nur mehr in Unit Tests selbst aufgerufen, sonst automatisch über DI. Alle Objekte werden über DI injected (außer evt. Hilfsklassen ala Validators, ...).

Stellen Sie sicher, dass auf allen Schichten Abhängigkeiten als Interfaces über Constructor Injection übergeben bekommen (siehe IoC / DI / Constructor Injection Video).
Passen Sie die Unit Tests entsprechend an!
2. Führen Sie **Logging** in allen Layers ein (geht jetzt mit DI ziemlich einfach!).
Loggen Sie mit dem eingebauten **ASP.NET Logging Framework** unter Beachtung unterschiedlicher Levels (Error, Warning, Info, Debug, ..).
Exception Handling kommt erst später dazu, also reicht es derzeit aus Info / Debug zu verwenden, um entsprechende Fortschrittsmeldungen auszugeben.
3. **Konfigurieren** Sie den DB Connection String von Entity Framework ab jetzt über **Asp.Net Core Configuration** (appsettings.json, ..) (geht jetzt mit DI ziemlich einfach!).
In der CD Pipeline / Release soll der ConnectionString dann mit dem Azure SQL DB ConnectionString überschreiben.
4. Stellen Sie nochmals sicher, dass die Code Coverage im BUILD Ergebnis aufscheint!
Bei der **Code Coverage (>= 70%)** können Sie folgende Typen ignorieren
 - Alle Entities / Models / DTOs / ...
 - Validators
 - Startup.cs, Program.cs
 - Automapper Profiles
 - Custom Attributes, Custom Filters
5. Nutzen Sie die Zeit nochmals über das Projekt zu schauen, und Übung 1-4 nachzuziehen!
Refactoren Sie wo nötig existierenden Code, um zum Gesamtstand des Projektes zu passen.
Z.B. sind die Interfaces, Constructor Injection, Logging, Unit Tests, .. durchgängig umgesetzt?
6. Abgabe:
Der MAIN Branch ihres Azure DevOps Git Repositories sollte immer den Abgabestand des Source Codes enthalten. Arbeiten Sie auf einem anderen Branch und mergen Sie auf MAIN, sobald Sie abgeben / releasen.
Die Abgabe auf Moodle einmal pro Gruppe nicht vergessen! (Gruppen müssen auch auf Moodle eingetragen sein!)