

Sprint 6 - Service Agent & Exception Handling

1. Implementieren Sie einen Webservice Agent zum Umwandeln von Adressen auf GPS Koordinaten.
Nutzen Sie für die Encodierung externe Webservices von OpenStreetMap, Bing oder Google.

Der Agent liegt in drei eigenen Projekten:

- <Firma>.<Projekt>.ServiceAgents
- <Firma>.<Projekt>.ServiceAgents.Interfaces
- <Firma>.<Projekt>.ServiceAgents.Tests

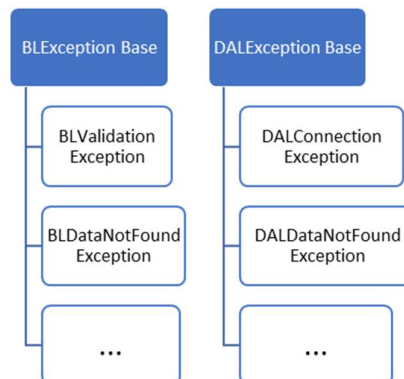
Der Agent könnte z.B. ungefähr so implementiert sein...

```
public interface IGeoEncodingAgent
{
    GeoCoordinates EncodeAddress(Address a);
}

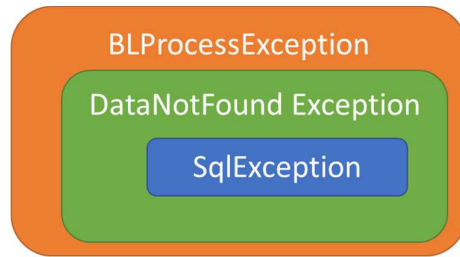
public class BingEncodingAgent : IGeoEncodingAgent...
```

2. Führen Sie **Exception Handling** in allen Layers ein.
Exceptions sollen auf jedem Layer gefangen, behandelt (mit Logging Framework geloggt) und ggf. weiter geworfen werden.
Beim Weiterwerfen in den nächst höheren Layer, sollen die Exceptions in Layer-spezifische **Custom-Exceptions** verpackt (gewrapped) werden, je nach Fehlerart.
Diese leiten von einer Layer-Basisklasse für alle Exceptions ab.

Zum Beispiel:



Da die Exceptions über die Layers immer wieder gefangen, ge-wrapped und weitergeworfen werden entsteht ein Exception Tree, z.B.:



Am Service Layer soll die Exception in die entsprechende HTTP Fehlermeldung (StatusCode, Swagger konforme Error-Message, ..) umgewandelt werden, die dann vom Webserver zurückgegeben wird.

3. **Reminder:** 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

Nutzen Sie die Zeit nochmals über das Projekt zu schauen, und Übung 1-5 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?

4. 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!)