



Werkgroep GAP		31
6 oktober 2009	Not.	2

Aanwezig: Tommy, Bart, Broes, Johan

Verontschuldigd: Peter

Business method parameters: objecten vs ID's

Een wijziging van Broes bracht de volgende discussie aan het licht: Gebruiken we in de business methods objecten of ID's als parameters:

Objecten:

<https://develop.chiro.be/trac/cg2/browser/trunk/Solution/Cg2.Workers/GelieerdePersonenManager.cs?rev=563#L308>

ID's:

<https://develop.chiro.be/trac/cg2/browser/trunk/Solution/Cg2.Workers/GelieerdePersonenManager.cs?rev=564#L308>

Conclusie:

De servicelaag werkt met ID's, de businesslaag met parameters. Bijv. voor het koppelen van gelieerde personen aan een categorie:

- Service ontvangt ID's van gelieerde personen en categorie
- Service spreekt business aan om personen en categorie op te halen
- Service spreekt business aan om te koppelen (check autorisatie en goedleggen pointers)
- Service spreekt business aan om te bewaren

Op die manier is iedereen tevreden ;-), en zijn de business methods makkelijker te hergebruiken.

Vordering implementatie

Zie takenlijst: op <https://develop.chiro.be/trac/cg2/report/8>

Programmeerdag 17/10

- Enkel zaterdag
- Kipdorp
- 9.00 – 19.00 uur
- Middageten & avondeten wordt voorzien

Organisaties vs personen

Als we 'organisaties' toevoegen aan ons datamodel, dan zorgt dat voor wel wat extra problemen. Deze komen bijvoorbeeld tot uiting wanneer je etiketten wil afdrukken.

Je wil een etiket kunnen maken voor

- Het gemeentebestuur p/a Richard
- Het gemeentebestuur p/a/ op adres van de gemeente
- ...

We gaan het voorlopig pragmatisch oplossen: we geven iedere persoon een extra adreslijn, waar dan nog 'tav de parochieraad' gezet kan worden.

Varia

Resharper

Johan mailt rechtstreeks naar resharper met de vraag of we educational licensing kunnen krijgen voor de Chiro.

IOC

De dependency injection bij het creëren van workers moet steeds door de IOC-container gebeuren. Ook in tests. Container.Register kan waarschijnlijk gebruikt worden om mocks te gebruiken met unity. Eventueel documentatie opzoeken over de combinatie Unity/Moq. Johan volgt dit op.

IList vs IEnumerable

Als er niets aan een collectie toegevoegd moet kunnen worden, gebruik je beter IEnumerable ipv IList.

Settings in testprojecten

In tests is het niet de gewoonte om Properties.Settings te gebruiken. Gebruik gewone constanten in de plaats.

Opkuis in tests

Op te zoeken: [TestTearDown]

Equals en GetHashCode voor entity's

Als GetHashCode geoverridet wordt voor entity's, dan geeft dat soms problemen geven bij het deserialiseren in DetachObjectGraph. Toch zouden we graag onze eigen Identity/Equality willen implementeren, zodat we bijv. 'distinct' kunnen gebruiken in linq query's. Johan zoekt dit verder uit.

Equals vs operator==

De == operator geeft niet standaard hetzelfde resultaat als Equals. Misschien moeten we dat wel zo implementeren. Nog uit te vinden.

Communicatievorm: enum -> klasse

Wijziging werd doorgevoerd.

Extension Methods op IBasisEntiteit

IBasisEntiteit heeft 2 extension methods, die niet geweldig veel voorstellen, en worden gebruikt in properties van de orm-classes. Die extension methods hebben weinig toegevoegde waarde; de implementatie kan rechtstreeks in betreffende properties gezet worden.

Nieuwe Kracht (Steven Lemmens)

Nog niets van gehoord.

Update: Tommy gaat op 14/10 bij hem langs.

Volgende bijeenkomsten

Programmeerdag 17/10

Programmeervergadering 20/10.