

CA2 Dokumentation

REST-API description

CityInfo		
Metode	URI	Beskrivelse
<i>Get</i>	<i>api/zipcode</i>	<i>Returnerer alle postnumre.</i>

<u>PersonResource</u>		
<u>get</u>	<u>complete/{id}</u>	<u>Returnerer en person ud fra personens ID.</u>
<u>get</u>	<u>contactinfo/{id}</u>	<u>returnerer en persons kontaktinformation ud fra personens ID.</u>
<u>get</u>	<u>all/complete</u>	<u>Returnerer alle personer.</u>
<u>get</u>	<u>all/contactinfo</u>	<u>Returnerer alt kontaktinformation for alle personer.</u>
<u>get</u>	<u>all/complete/zipcode/{zip}</u>	<u>Returnerer alle personer som bor i et bestemt postnummer.</u>
<u>put</u>	<u>addPerson</u>	<u>tilføjer en person.</u>
<u>post</u>	<u>editPerson</u>	<u>redigerer en persons info.</u>
<u>delete</u>	<u>deletePerson</u>	<u>Sletter en person.</u>

<u>HobbyResource</u>		
<u>get</u>	<u>single/{hobby}</u>	<u>returnerer en person ud fra en hobby.</u>

<u>get</u>	<u>all/{hobby}</u>	<u>returnerer alle personer med en bestemt hobby.</u>
------------	--------------------	---

<u>CompanyResource</u>		
<u>get</u>	<u>all/population/{minPop}</u>	<u>returnerer alle firmaer med flere medarbejdere end "minPop"</u>
<u>get</u>	<u>single/cpr/{cvr}</u>	<u>returnerer et firma ud fra firmaets cvr</u>
<u>get</u>	<u>comp/all</u>	<u>returnerer alle firmaer i databasen.</u>
<u>put</u>	<u>addcompany</u>	<u>tilføjer et firma.</u>
<u>post</u>	<u>editcompany</u>	<u>redigerer et firma.</u>

Her er hvordan vores REST-API ser ud med hensyn til de ressourcer der skulle bruges og deres GET, POST, PUT og DELETE metoder.

Error responses description

I vores projekt er der blevet lavet to mapper, en med exceptionmappere og en, med navnet exception.error, som gør den i stand til at blive udskrevet. Det der sker med vores error responses er at hvis man indsetter forkert info eller data i f.eks. Enden af url'en eller på vores html side får man enten en 404 not found error eller en 500 internal server error. Disse fejl er blevet testet med og skulle virke.

JSon format description

Vi lavede en JSONConverter klasse, som laver al vores information om til JSON, hvilket kan bruges når der skal udskrives noget tekst. Dette kan bruges til frontend javascript og HTML. Når vi har f.eks. En one-to-one annotation, igennem java til JSON, bruges GSON biblioteket.

Strategy

- Plain unit tests

Vi brugte unit tests til at bekræfte funktionalitet af programmet. Se billag 1

- Testing database facade

JUnit testsne har interfacet med facaderne. Se billag 1

- Testing REST-API

Vi havde nogle rest assured tests som virkede, før at de ting servletsne var afhængige af blev ændret.

Hvem gjorde hvad?

- Michael
 - Har lavet frontend.
 - Har lavet HTML og JavaScript siden.
 - Har indskrevet SQL data.
 - Har lavet lidt på person og companyresource.
 - Skrevet på API Dokumentation og inheritance implementation strategy.

Egne tanker

Jeg synes at alle bidragende så meget som de nu kunne i dette projekt, de gjorde hvad de var sat til at gøre med de mini sprints specificeret til dem og fik af opgaver. Selvfølgelig arbejdede nogen mere end andre, på grund af forskellige niveauer og kompetencer, men alle gjorde deres bedste. Det er dét I har lært os at man får studypoints for! Jeg synes derfor alle skal have 5 studypoints da alle har arbejdet hver dag og er mødt op.

Projektet gik dog lidt i kage til sidst, men det er jo af sine fejl at man lære.

- Lasse
 - Har lavet backend.
 - Har lavet facade.
 - Har lavet facadeTest.
 - Har lavet JUnit.
 - Har lavet entities.
 - Har lavet rest ressourcer

Egne tanker

Projektet gik ok indtil torsdag hvor det viste sig at der skulle ændres på nogle relationer i databasen. Det var skuffende at skulle lave en del ting om. Jeg har bla fået det ud af projektet at det er vigtigt med enighed i implementering af tingene, og at holde styr på hvem der skal lave på hvad.

Jeg synes dog stadig at alle skal have 5 studypoints.

- Patrick
 - Har lavet backend.
 - Har lavet rest ressourcer.
 - Har lavet entities.
 - Har lavet error og exception handling.
 - Har lavet JSONconverter.
 - Har lavet persistence.
 - Har lavet facade

Egne tanker

Jeg synes allesammen har fortjent 5 studypoints da de gjorde et godt stykke arbejde. Projektet gik dog galt til sidst med nogle fejl vi ikke lige havde overskud til at kunne rette og at tiden ikke helt var der til at kunne lave alt helt færdigt.

- Christian
 - Har indskrevet sql data.
 - Har lavet company resource.
 - Gitignore
 - Skrevet på API Dokumentation.

Egne tanker

Jeg synes alle gjorde et fint stykke arbejde og fortjener 5 studypoints. Lidt ærgeligt med fejlen til sidst, men det sker.

Inheritance implementation strategy

Vi brugte inheritance strategien Joined Table Strategy. Denne strategi blev brugt til at undgå null værdier og fordi det var meget mere overskueligt at bruge. Med joined Table strategien kunne man også bedst følge modellen givet til os, da mange ting skulle arve fra en super klasse, hvilket i dette tilfælde var InfoEntity.

Precise description of how to test the system

Vi bruger SQL schemaer med navnene "ca" og "ca2test". I disse - indsæt værdier fra builder.sql

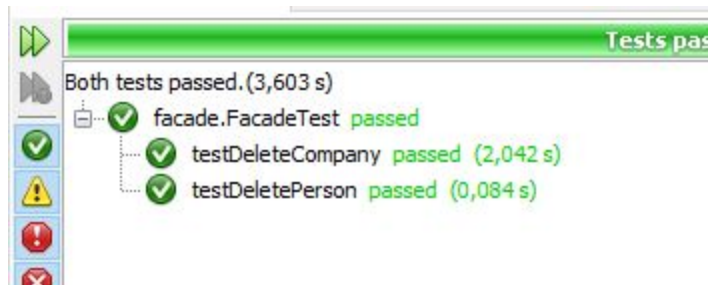
Serveren skulle først køre for at kunne teste rest-api'et, men rest api'et gik i stykker efter en refaktorering af fremmednøgler i databasen, så det kan ikke køres nu (udover testen af det om der overhovedet er forbindelse).

Konklusion

Alt i alt, har vi stort set lavet projektet så godt som vi nu kunne have nået at lave det. Hvis vi havde haft mere tid til dette projekt ville det have nogenlunde fungeret, da der var en refaktorings fejl, som skete mere i den sidste ende af projektet, der ødelagde det. Hvis vi havde haft mere tid ville vi have rettet fejlen og finjusteret projektet mere, så vores frontend og backend kunne have set flottere ud og at det også ville kunne virke. Det var et meget tidskrævende og svært projekt, som nok er meget lærerigt for os, men det kunne godt have været på et lidt nemmere niveau.



Det driller når man kører alle tests på en gang.



Billag 1.

testDeletecompany, deleteperson, persistinfoentity og company blev her kørt for sig og resten udkommenteret - bestod.