

# Refactoring Cucumbers and Glue



Fabian





# Part I



Refactor the Cucumber

# Starting point

- Cucumber tests that run on the INT environment and should verify that our API is up and running

Funktionalität: OKP Mutation

Szenario: OKP mit Dunkerverarbeitung mutieren

Gegeben sei der Kunde PRIVATKUNDE ist auf der INT Umgebung angemeldet

Wenn der Kunde auswählbare Grundversicherungsmodelle fuer die Bestandespolice 10012819869 abfragt

Dann stehen korrekt formatierte auswählbare Grundversicherungsmodelle zur Auswahl

Dann im erstellten Angebot das Modell, die Franchise oder das Versichertesrisiko angepasst werden

Dann die Grundversicherung Mutation mit Anliegen I01031999 mit Dunkerverarbeitung abgeschlossen wird

# Issues

Funktionalität: OKP Mutation

Szenario: OKP mit Dunkelverarbeitung mutieren

Gegeben sei der Kunde PRIVATKUNDE ist auf der INT Umgebung angemeldet

Wenn der Kunde auswählbare Grundversicherungsmodelle fuer die Bestandespolice 10012819869 abfragt

Dann stehen korrekt formatierte auswählbare Grundversicherungsmodelle zur Auswahl

Dann im erstellten Angebot das Modell, die Franchise oder das Versichertesrisiko angepasst werden

Dann die Grundversicherung Mutation mit Anliegen I01031999 mit Dunkelverarbeitung abgeschlossen wird

Purpose is not clear

- They should not test the domain logic  
-> We already have Cucumber Domain Tests

Half of the Parameters are in Cucumber, the other half in Java

- Either everything in Cucumber or in Java

We started using these Tests for manual testing purposes

- We need something like SwaggerUI

# Maybe something like Swagger UI?

- Generated
- Easy to use for manual Testing

The screenshot displays the Swagger UI interface for an API. It features a list of endpoints under the 'Users' and 'Projects' sections. The 'Projects' section is expanded, showing a GET endpoint at '/api/v1/projects' with the description 'List all projects'. Below the endpoint details, there is a blue 'Execute' button. A red arrow points to this button. The 'Responses' section shows a 200 status code with a response content type of 'application/json'. An 'Example Value' is provided in a dark box, showing a JSON object with a 'contributors' array and other fields like 'createdAt', 'updatedAt', 'id', 'name', and 'creator'.

```
{
  "contributors": [
    {
      "createdAt": 0,
      "updatedAt": 0,
      "id": "string",
      "username": "string",
      "firstName": "string",
      "lastName": "string",
      "email": "string",
      "password": "string",
      "role": "annotator"
    }
  ],
  "createdAt": 0,
  "updatedAt": 0,
  "id": "string",
  "name": "string",
  "creator": "string",
  "role": "string"
}
```

# Or SoapUI?

- Can generate A Swagger-like UI
- Another tool
- Get WSDL, Token etc Is already solved in Java (BSFW)

The screenshot displays the SoapUI 5.4.0 application window. The main area shows a SOAP request and its corresponding response. The request is a SOAP envelope with a header and a body containing a calculateAuswahlbareGrundversicherung element. The response is a SOAP envelope with a header and a body containing a fault element.

**Request 1**

```
<?xml version='1.0' encoding='UTF-8'>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header/>
  <soap:Body>
    <ns:calculateAuswahlbareGrundversicherung
      xmlns:ns="http://www.oasis-open.org/ws-sx/ws-securitypolicy/200702"
      <ns:policyNr="123"/>
    </ns:calculateAuswahlbareGrundversicherung>
  </soap:Body>
</soap:Envelope>
```

**Response**

```
<?xml version='1.0' encoding='UTF-8'>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <soap:Fault>
      <faultcode>soap:Server</faultcode>
      <faultstring>These policy alternatives can not be
        (http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702
        [http://www.oasis-open.org/ws-sx/ws-securitypolicy/200702
        [http://docs.oasis-open.org/ws-sx/ws-securitypolicy/200702
      </soap:Fault>
    </soap:Body>
  </soap:Envelope>
```

**Request Properties**

Property	Value
Name	Request 1
Description	
Message Size	458
Encoding	UTF-8
Endpoint	https://caas-services-int.css.ch/bs-angebot-kv-...
Timeout	
Bind Address	
Follow Redirects	true
Username	

response time: 318ms (457 bytes)

SoapUI log http log jetty log error log wsrm log memory log

# New Cucumber approach

```
Szenario: OKP mit Dunkelverarbeitung mutieren
  Gegeben sei der Kunde PRIVATKUNDE ist auf der INT Umgebung angemeldet
  Wenn calculateAuswaehlbareGrundversicherungen aufgerufen wird:
    | PoliceNr |
    | 10012819869 |
  Wenn modifyGrundversicherung aufgerufen wird:
    | AngebotNr | Anbieter | ProduktNummer | VersichertesRisiko | FranchiseStufe |
    | aus Variablen | CSS | 1 | RISIKO_KRANKHEIT_UNFALL | 25 |
  Wenn executeGrundversicherungMutationAbschluss aufgerufen wird:
    | AngebotNr | AnliegenNr |
    | aus Variablen | I01031999 |
```

- Runs automatically in the CICD pipeline
- Can be used for manual Testing
- No new tool



# Part II



Refactor the Glue



# Smelling Glue

```
19 public class ExecuteGrundversicherungMutationAbschlussStepDef {
20
21     private Variablen variablen;
22
23     @Before
24     public void setUp() {
25         Testsystem testsystem = Testsystem.getInstance();
26         this.variablen = testsystem.variablen; // Feature Envoy on Testsystem
27     }
28
29     @SneakyThrows
30     @Wenn("executeGrundversicherungMutationAbschluss.aufgerufen.wird:")
31     public void executeGrundversicherungMutationAbschlussAufgerufenWird(BSExecuteGrundversicherungMutationAbschlussIn in) {
32         final Environment environment = variablen.get(Variablen.ENVIRONMENT, Environment.class); // Feature Envoy on variablen
33         final String accessToken = variablen.get(Variablen.ACCESS_TOKEN, String.class); // Duplicate Code in each Step
34         final BSKvAccessorFactory factory = new BSKvAccessorFactory();
35         factory.setOidcToken(accessToken); // Feature Envoy on factory
36         factory.setEndpointUrl(environment.endpointUrl);
37         factory.createBSAngebotKvLogic().executeGrundversicherungMutationAbschluss(in); // Message Chain
38     }
39
40     @DataTableType
41     public BSExecuteGrundversicherungMutationAbschlussIn testDataTableRow(Map<String, String> row) {
42         final BSExecuteGrundversicherungMutationAbschlussIn in = new BSExecuteGrundversicherungMutationAbschlussInImpl();
43         final String angebotNr = variablen.get(ANGEBOT_NR, String.class);
44         in.setAngebotNr("aus Variablen".equals(row.get("AngebotNr")) ? angebotNr : row.get("AngebotNr")); // Duplicate Code
45         in.setAnliegenNr(row.get("AnliegenNr"));
46         return in;
47     }
48 }
49
```

# After

```
17 public class ExecuteGrundversicherungMutationAbschlussStepDef {
18
19     private TestFixture testFixture;
20
21     @Before
22     public void setUp() {
23         this.testFixture = TestFixture.getInstance();
24     }
25
26     @SneakyThrows
27     @Wenn("executeGrundversicherungMutationAbschluss aufgerufen wird:")
28     public void executegrundversicherungmutationabschlussAufgerufenWird(BSExecuteGrundversicherungMutationAbschlussIn in) {
29         // getBSAngebotKvLogic can not run before Environment is set in in another stepdef
30         testFixture.getBSAngebotKvLogic().executeGrundversicherungMutationAbschluss(in);
31     }
32
33     @DataTableType
34     public BSExecuteGrundversicherungMutationAbschlussIn testDataTableRow(Map<String, String> map) {
35         final Parameters parameters = new Parameters(map);
36         final BSExecuteGrundversicherungMutationAbschlussIn in = new BSExecuteGrundversicherungMutationAbschlussInImpl();
37         in.setAngebotNr(parameters.getVariableOrParameter(ANGEBOT_NR, testFixture));
38         in.setAnliegenNr(parameters.get("AnliegenNr"));
39         return in;
40     }
41
42 }
```

# Conclusion



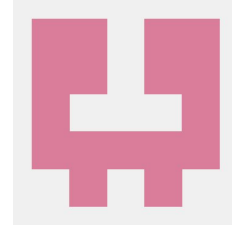
Mindset of refactoring also applies to other aspects of the project

A little bit of refactoring already improves the code massively (80:20)



**Questions?**

**Thank you for  
your time!**



**Find me  
on  
Github**

[github.com/unSinn](https://github.com/unSinn)