



# Spock und AsciiDoc

## Vom Test zur Spezifikation und zurück

DB System GmbH | Ralf D. Müller

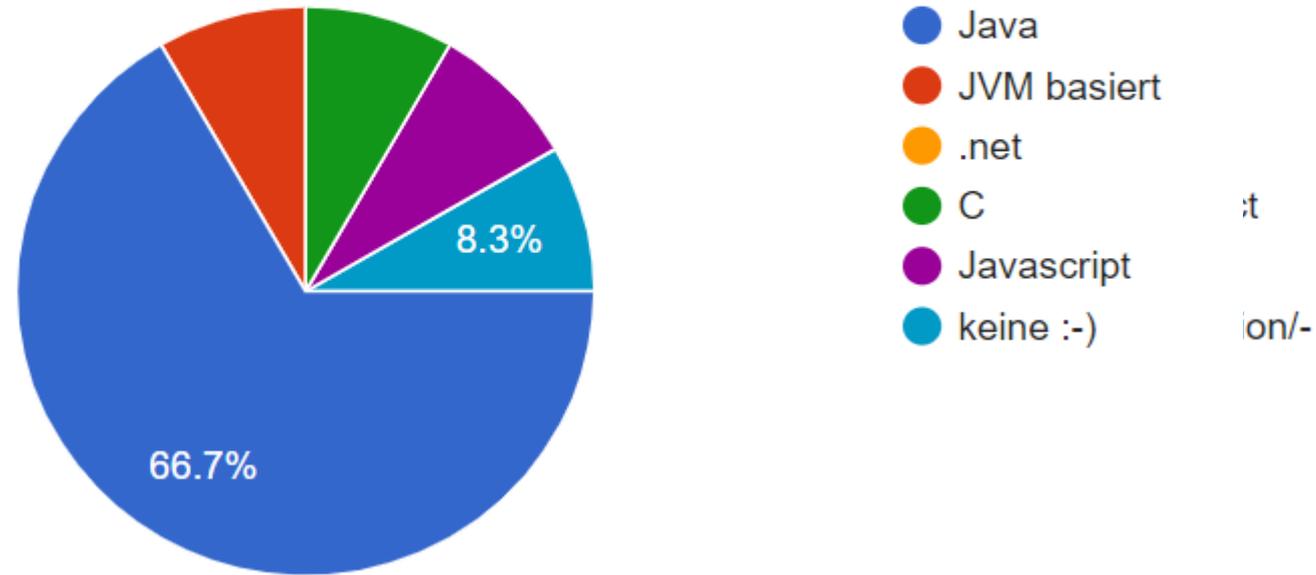
**DB System**

Moving the digital future. Together.



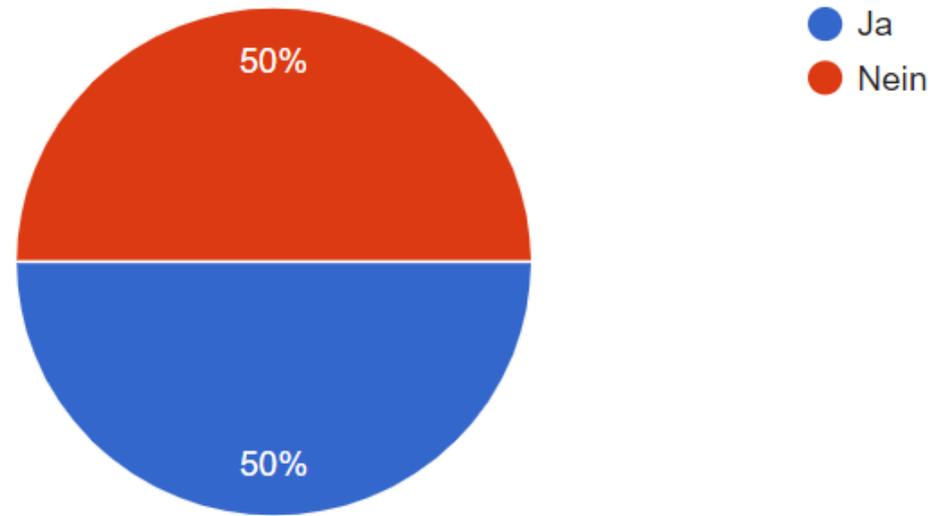
In welcher Rolle arbeitest Du hauptsächlich? (z.B. Entwickler, Tester, ...)  
Mit welcher Programmiersprache arbeitest Du hauptsächlich?

12 responses



Kennst Du Spock schon?

12 responses



**Peter Niederwieser**  
@pniederw



**Luke Daley**  
@ldaley



**Leonard  
Brüning**



```
void testeAdditionJUnit() {  
  
    Calculator calculator = new Calculator()  
  
    BigDecimal result = calculator.plus( 3, 4)  
  
    assertEquals(7, result, 0.1)  
  
}
```



```
void "Rechner kann addieren Spock I" () {  
    given:  
        Calculator calculator = new Calculator()  
    when:  
        BigDecimal result = calculator.plus( 3, 4)  
    then:  
        assertEquals(7, result, 0.1)  
}
```



```
void "Rechner kann addieren Spock II" () {  
  given: "ein Caclulator Objekt"  
    Calculator calculator = new Calculator()  
  when: "zwei Zahlen addiert werden"  
    BigDecimal result = calculator.plus( 3, 4)  
  then: "erhalte ich das richtige Ergebnis"  
    assertEquals(7, result, 0.1)  
}
```



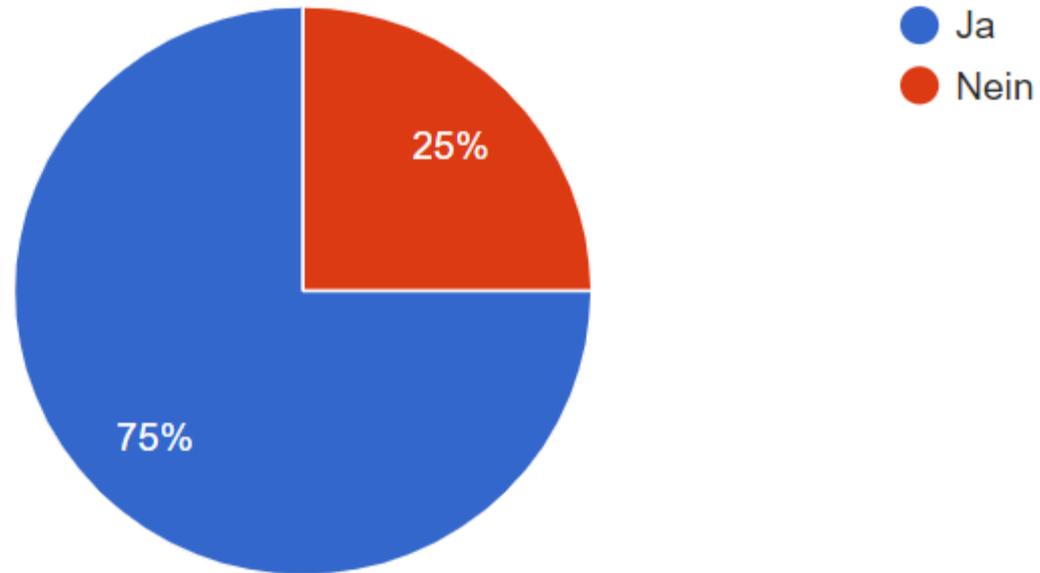
```
void "Rechner kann addieren Spock III" () {  
  given: "ein Caclulator Objekt"  
    Calculator calculator = new Calculator()  
  when: "#a und #b addiert werden"  
    BigDecimal result = calculator.plus( a, b)  
  then: "erhalte ich #expected"  
    assertEquals(expected, result, 0.1)  
  where: "Beispieldaten"  
    a | b || expected  
    3 | 4 || 7  
    4 | 3 || 7  
    -5 | 7 || 2  
    -7 | 5 || -2  
}
```



```
void "Rechner kann addieren Spock IV" () {  
  given: "ein Caclulator Objekt"  
    def calculator = new Calculator()  
  when: "#a und #b addiert werden"  
    def result = calculator.plus( a, b)  
  then: "erhalte ich #expected"  
    result == expected  
  where: "Beispieldaten"  
    a   |   b   || expected  
    3   |   4   ||    7  
    4   |   3   ||    7  
   -5  |   7   ||    2  
   -7  |   5   ||   -2  
}
```

Kennst Du AsciiDoc schon?

12 responses



**Dan Allen** 🙌  
@mojavelinux

= **AsciiDoc**

**\*AsciiDoc\*** ist ein Dokumentationsformat, welches mit Hilfe des Tools **\*Asciidoctor\*** in verschiedene andere Formate wie z.B. HTML gewandelt werden kann.

AsciiDoc hat den Vorteil, dass eine eMail schreiben kann und sich wie Sourcecode versionieren lässt.

Sie auch <https://asciidoctor.org>

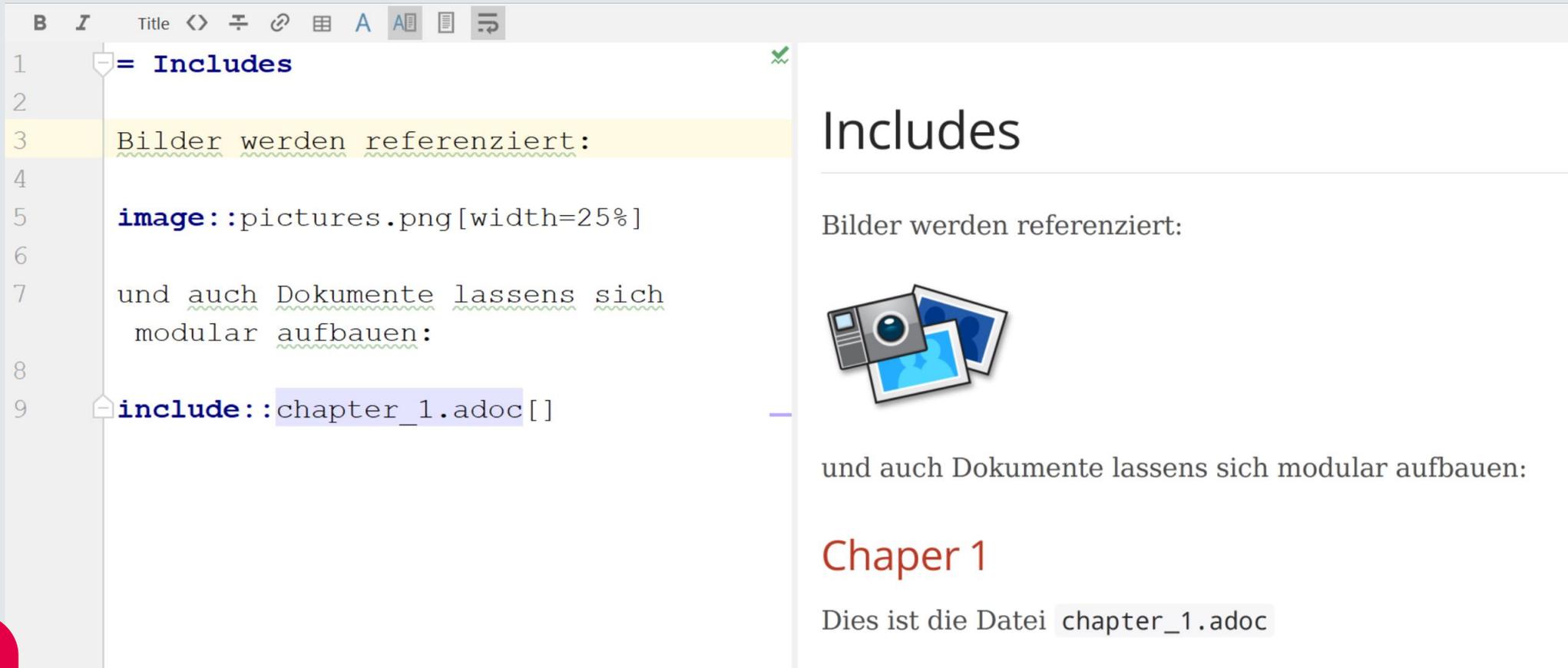
## AsciiDoc

**AsciiDoc** ist ein Dokumentationsformat, welches mit Hilfe des Tools **Asciidoctor** in verschiedene andere Formate wie z.B. HTML gewandelt werden kann.

AsciiDoc hat den Vorteil, dass man es einfach wie eine eMail schreiben kann und sich wie Sourcecode versionieren lässt.

Sie auch [asciidoctor.org](https://asciidoctor.org)





The screenshot shows an AsciiDoc editor interface. On the left, the source code is displayed with line numbers 1 through 9. Line 1 contains the title `= Includes`. Line 3 contains the text `Bilder werden referenziert:`. Line 5 contains the AsciiDoc image macro `image::pictures.png[width=25%]`. Line 7 contains the text `und auch Dokumente lassens sich modular aufbauen:`. Line 9 contains the include macro `include::chapter_1.adoc[]`. On the right, the rendered output is shown. It features the title `Includes`, followed by the text `Bilder werden referenziert:`, an image of a camera and two photos, the text `und auch Dokumente lassens sich modular aufbauen:`, the title `Chaper 1` (note the typo), and the text `Dies ist die Datei chapter_1.adoc`.



**Alexander Schwartz**  
@ahus1de

```
1  = Tabellen
2
3  ...gehen klassisch als Ascii-Art:
4
5  .Ascii-Art-Style
6  |===
7  |Header 1 |Header 2 |Header 3
8  |Column 1 |Column 2 |Column 3
9  |Column 1 |Column 2 |Column 3
10 |===
11
12 ...aber auch wartungsfreundlicher:
13
14 .AsciiDoc-Style
15 |===
16 |Header 1 |Header 2 |Header 3
17
18 |Column 1, row 1
19 |Column 2, row 1
20 |Column 3, row 1
21
22 |Column 1, row 2
23 |Column 2, row 2
24 |Column 3, row 2
25 |===
```



## Tabellen

...gehen klassisch als Ascii-Art:

*Table 1. Ascii-Art-Style*

Header 1	Header 2	Header 3
Column 1	Column 2	Column 3
Column 1	Column 2	Column 3

...aber auch wartungsfreundlicher:

*Table 2. AsciiDoc-Style*

<b>Header 1</b>	<b>Header 2</b>	<b>Header 3</b>
Column 1, row 1	Column 2, row 1	Column 3, row 1
Column 1, row 2	Column 2, row 2	Column 3, row 2



```
1 = Diagramme
2
3 ...können direkt als Text definiert werden:
4
5 [plantuml
6 ----
7 User -> Server
8     Server -> Datenbank
9     Server <-- Datenbank
10 User <-- Server
11 ----
12
13 [plantuml
14 ----
15 [AsciiDoctor] -> [Gradle]: invoked by
16 ----
```

## Diagramme

...können direkt als Text definiert werden:

```
classDiagram
    class User
    class Server
    class Datenbank
    User ..> Server
    Server ..> Datenbank
    Server ..> User
    Datenbank ..> User
```

```
graph LR
    AsciiDoctor[AsciiDoctor] -- invoked by --> Gradle[Gradle]
```







```
void "Rechner kann addieren Spock IV" () {  
  given: "ein Caclulator Objekt"  
    def calculator = new Calculator()  
  when: "#a und #b addiert werden"  
    def result = calculator.plus( a, b)  
  then: "erhalte ich #expected"  
    result == expected  
  where: "Beispieldaten"  
    a   |   b   || expected  
    3   |   4   ||   7  
    4   |   3   ||   7  
   -5  |   7   ||   2  
   -7  |   5   ||  -2  
}
```

# Was liefern die Tests als Ausgabe?



Debugger Console

Tests failed: 2, passed: 3, ignored: 3 of 8 tests – 65 ms

Test Name	Duration	Status
<default package>	65 ms	Failed
AppTest	65 ms	Failed
application has a greeting	31 ms	Passed
greeting contains 'hello'	29 ms	Failed
not implemented yet		Ignored
data driven hello	4 ms	Passed
data driven world	1 ms	Passed
GebTestSpec		Failed
simple screenshot test		Ignored
go method does NOT set the page		Ignored

```
Condition not satisfied:  
  
result == "hallo welt"  
|      |  
|      false  
|      6 differences (50% similarity)  
|      (H)ello w(or)l(d.)  
|      (h)allo w(e-)l(t-)  
Hello world.  
  
<Click to see difference>
```

## Class demo.TaschenrechnerSpec

all > demo > TaschenrechnerSpec

**8** tests  
**0** failures  
**3** ignored  
**0.030s** duration

**100%**  
successful

### Tests

Test	Duration	Result
Rechner kann addieren	0.030s	passed
Rechner kann addieren : data driven[0]	0s	passed
Rechner kann addieren : data driven[1]	0s	passed
Rechner kann addieren : data driven[2]	0s	passed
Rechner kann addieren : data driven[3]	0s	passed
Rechner kann dividieren	-	ignored
Rechner kann multiplizieren	-	ignored
Rechner kann subtrahieren	-	ignored



## Rechner kann addieren

*Given:* neuen Rechner instanziiieren

*When:* 3 + 4 eingegeben wird

*Then:* dann ist das Ergebnis 7

## Rechner kann addieren : data driven [0]

*Given:* neuen Rechner instanziiieren

*When:* 3 + 4 eingegeben wird

*Then:* dann ist das Ergebnis 7

*Where:* ----

*Examples:*

arg1	arg2	erwartetesErgebnis
3	4	7

OK

## Rechner kann subtrahieren

*noch nicht implementiert*

*Expect:* ----



**Renato Athaydes**  
@renatoathaydes

# Was liefern die Tests als Ausgabe?



Given: **neuen Rechner instanziiieren**

```
Rechner rechner = new Rechner()
```

When: **#arg1 + #arg2 eingegeben wird**

```
BigDecimal ergebnis = rechner.plus( arg1, arg2)
```

Then: **dann ist das Ergebnis #erwartetesErgebnis**

```
ergebnis == erwartetesErgebnis
```

Where: -

arg1	arg2	erwartetesErgebnis	
3	4	7	(PASS)
4	3	7	(PASS)
-5	7	2	(PASS)
-7	5	-2	(PASS)



**Tobias Kraft**  
@tokraft

# Was liefern die Tests als Ausgabe?



```
▲ hide passed test
When: die Bing Homepage aufgerufen wird
  to BingHomePage
Then: wird die Bing Homepage angezeigt
  report "Bing Home Page"
  at BingHomePage
When: nach 'calc' gesucht wird
  form.q = 'calc'
  sleep(200)
  searchBtn.click()
Then: erscheint die Suchresultseite
  report "Bing Search Result Page"
  at BingSearchResultPage
  +
```

## Table 2. Screenshots:



Figure 1. Bing Home Page

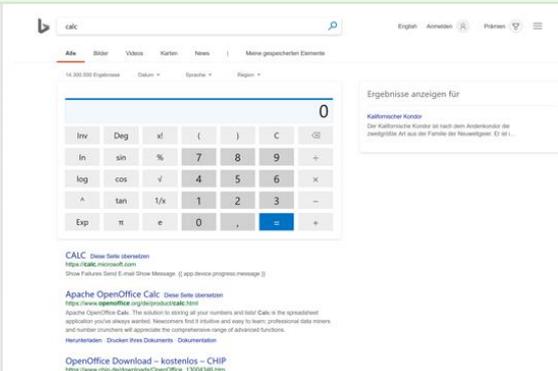


Figure 2. Bing Search Result Page



Tilman Ginzel



Luke Daley  
@ldaley



Marcin Erdmann  
@marcinerdmann





Spock Test Results

file:///C:/Users/ralfd/projects/github/SpockAsciiDocReports/build/docs/html5/spock-reports

## Table of Contents

- 1. Specification run results
  - 1.1. Specifications summary
  - 1.2. Specifications
- 2. Report for de.docs\_as\_co.AsciiDocSpockP
  - 2.1. Summary
  - 2.2. Features
    - 2.2.1. PASS: Parse File
- 3. Report for demo.TaschenrechnerSpec
  - 3.1. Summary
  - 3.2. Features
    - 3.2.1. PASS: Rechner kann addieren Spock I
    - 3.2.2. PASS: Rechner kann addieren Spock II
    - 3.2.3. PASS: Rechner kann addieren Spock III
    - 3.2.4. PASS: Rechner kann addieren Spock IV
    - 3.2.5. PASS: Rechner kann addieren
    - 3.2.6. PASS: Rechner kann addieren : data driven
    - 3.2.7. IGNORED: Rechner kann subtrahieren
    - 3.2.8. IGNORED: Rechner kann

## 4.2. Features

### 4.2.1. PASS: Bing-Rechner aufrufen

▲ hide passed test

**When: die Bing Homepage aufgerufen wird**

```
to BingHomePage
```

**Then: wird die Bing Homepage angezeigt**

```
report "Bing Home Page" at BingHomePage
```

**When: nach 'calc' gesucht wird**

```
form.q = 'calc'
sleep(200)
searchBtn.click()
```

**Then: erscheint die Suchresultseite**

```
report "Bing Search Result Page" at BingSearchResultPage
+
```

*Table 2. Screenshots:*

Figure 1. Bing Home Page

Figure 2. Bing Search Result Page



# Was liefern die Tests als Ausgabe?



```
▲ hide passed test
When: die Bing Homepage aufgerufen wird
  to BingHomePage
Then: wird die Bing Homepage angezeigt
  report "Bing Home Page"
  at BingHomePage
When: nach 'calc' gesucht wird
  form.q = 'calc'
  sleep(200)
  searchBtn.click()
Then: erscheint die Suchresultseite
  report "Bing Search Result Page"
  at BingSearchResultPage
  +
```

Table 2. Screenshots:

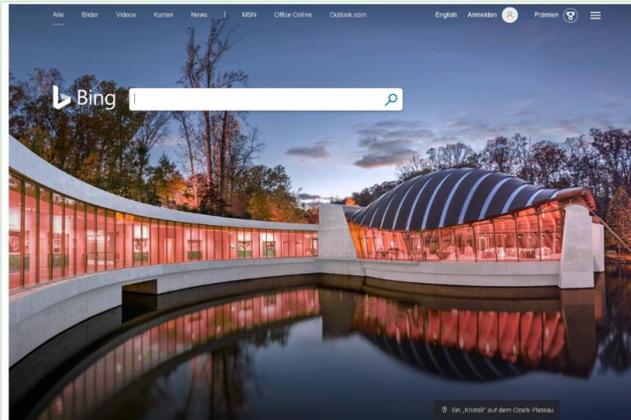


Figure 1. Bing Home Page

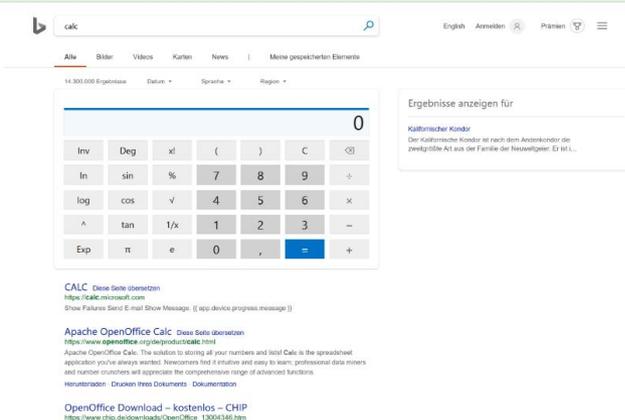


Figure 2. Bing Search Result Page



# Was können wir jetzt damit machen?



2. Report für die dbcc...  
3. Report für die...  
4. Report für die...





# Was können wir jetzt damit machen?



## Spezifikation

Dieses Projekt soll den Benutzer durch zusätzliche Features länger auf der Suchseite halten um höhere Werbeinnahmen zu generieren.

### 1. Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des Suchbegriffs "Rechner" oder bei der Eingabe einer Mathematischen Rechnung einen Taschenrechner als Web-Anwendung angezeigt bekommen, so daß ich nicht die Suche Verlassen muss

#### 1.1. Akzeptanztest

##### 1.1.1. PASS: Bing-Rechner aufrufen

When: die Bing Homepage aufgerufen wird

to BingHomePage

Then: wird die Bing Homepage angezeigt

report "Bing Home Page"

at BingHomePage

When: nach 'calc' gesucht wird

form.q = 'calc'

sleep(200)

searchBtn.click()

Then: erscheint die Suchresultseite

report "Bing Search Result Page"

at BingSearchResultPage

Table 1. Screenshots:



Figure 1. Bing Home Page

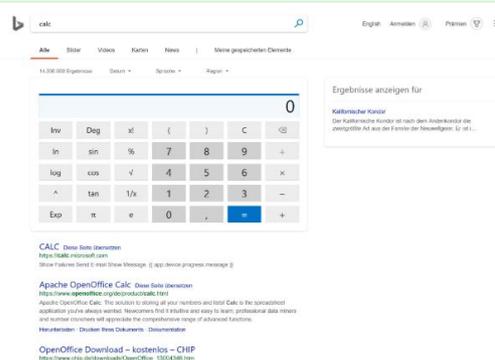


Figure 2. Bing Search Result Page



Anwendung



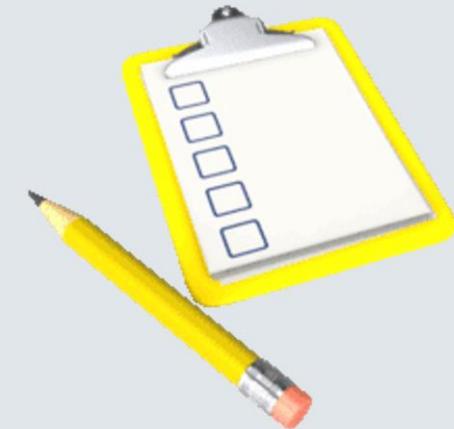
Tests



Test-  
Report



Spec



# ~~Was haben wir erreicht?~~ Was wollen wir erreichen?

Spec



Tests

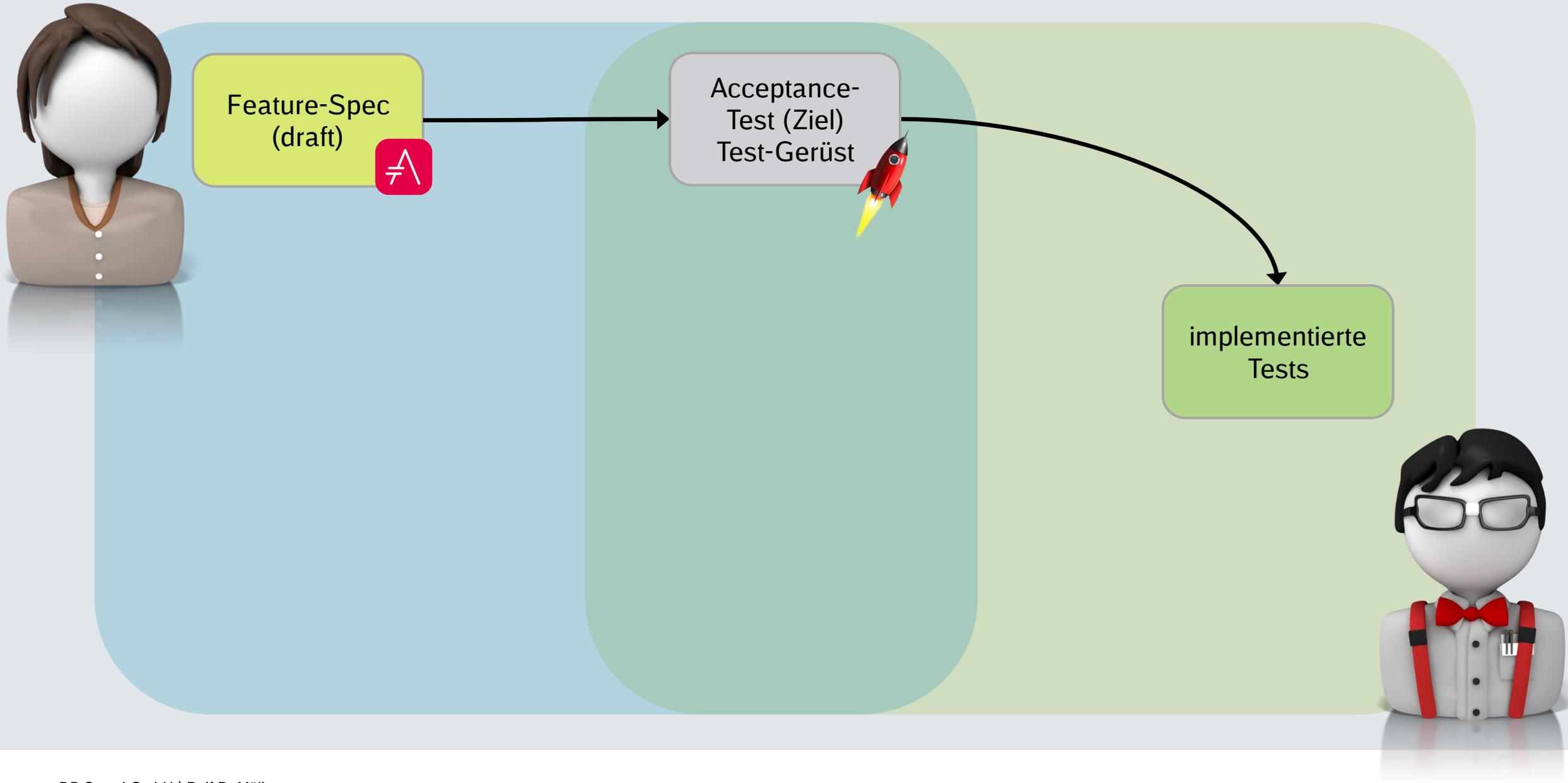


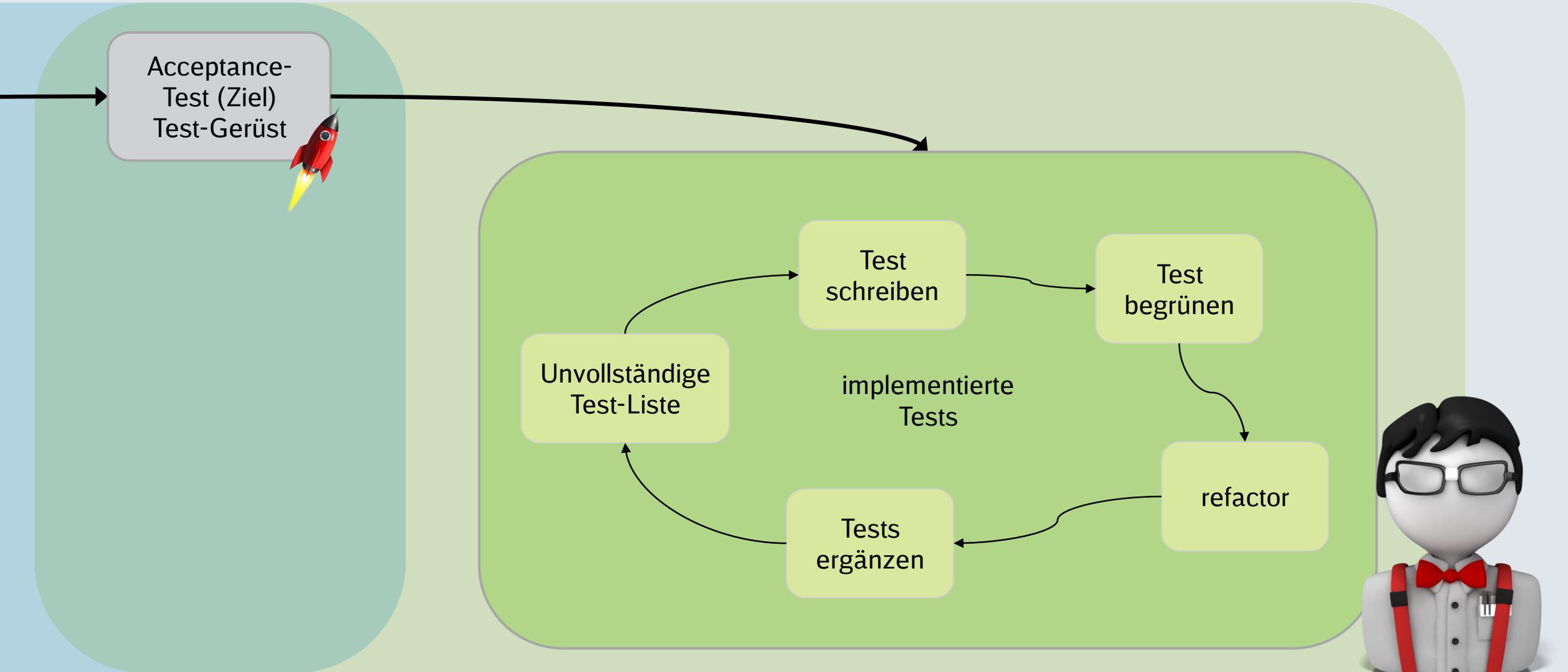
Test-  
Report

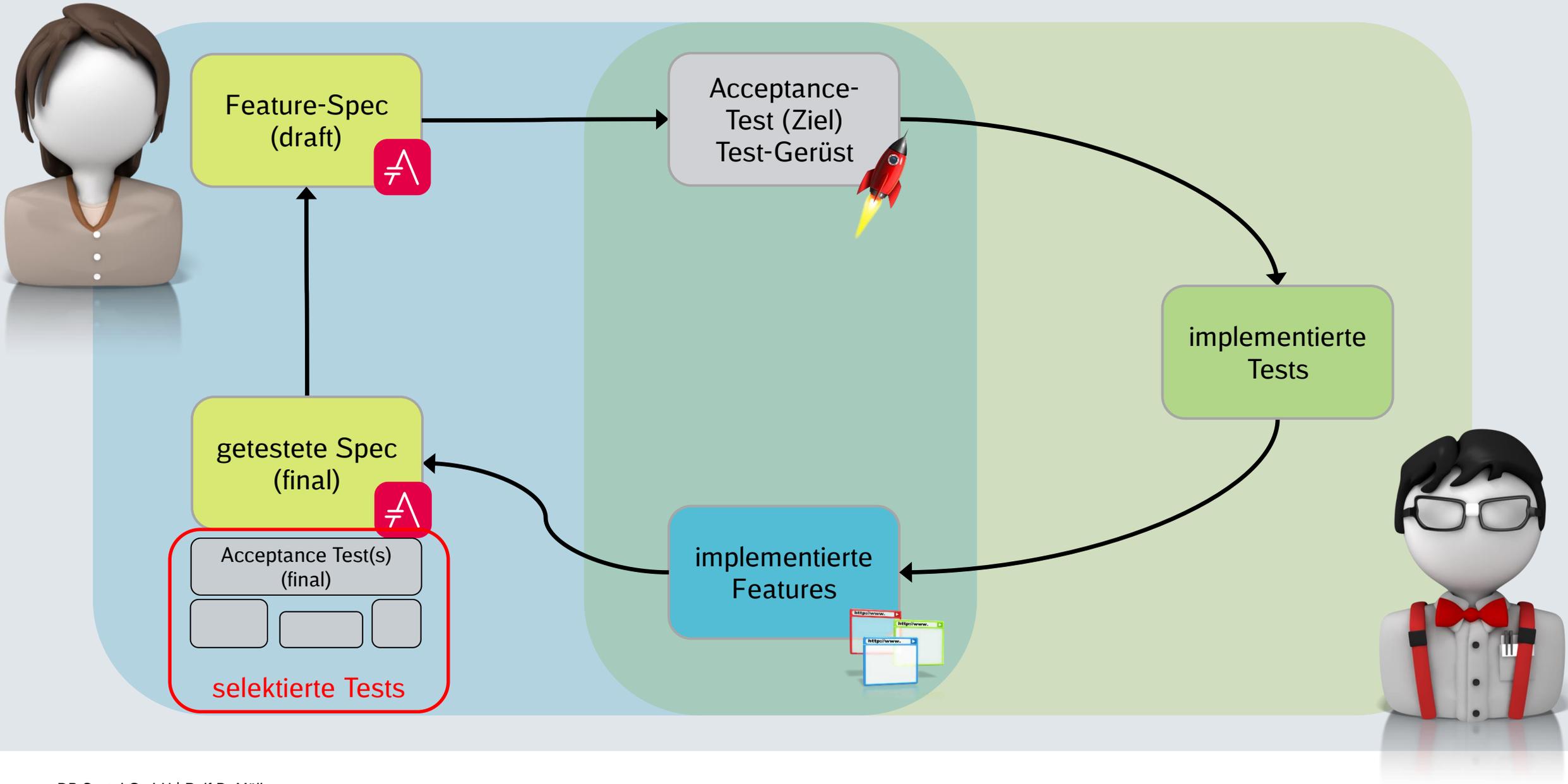


Anwendung









## = Spezifikation

```
include::config.adoc[]
```

Dieses Projekt soll den Benutzer durch zusätzliche Features länger auf der Suchseite halten um höhere Werbeinnahmen zu generieren.

## == Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des Suchbegriffs "Rechner" oder bei der Eingabe einer Mathematischen Rechnung einen Taschenrechner als Web-Anwendung angezeigt bekommen, so daß ich nicht die Suche Verlassen muss

## === Acceptancetests

### ==== User calls Calculator

```
[cols="1,10"]
```

```
|===
```

```
| Given | User is on the main page  
| When  | User enters 'calc' as search-term  
| Then  | the search result displays a calculator
```

```
|===
```

# Spezifikation

Dieses Projekt soll den Benutzer durch zusätzliche Features länger auf der Suchseite halten um höhere Werbeinnahmen zu generieren.

## 1. Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des Suchbegriffs "Rechner" oder bei der Eingabe einer Mathematischen Rechnung einen Taschenrechner als Web-Anwendung angezeigt bekommen, so daß ich nicht die Suche Verlassen muss

### 1.1. Acceptancetests

#### 1.1.1. User calls Calculator

Given	User is on the main page
When	User enters 'calc' as search-term and submits the form
Then	the search result displays a calculator

## = Spezifikation

```
include::config.adoc[]
```

Dieses Projekt soll den Benutzer durch zusätzliche Features länger auf der Suchseite halten um höhere Werbeinnahmen zu generieren.

## == Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des Suchbegriffs "Rechner" oder bei der Eingabe einer Mathematischen Rechnung einen Taschenrechner als Web-Anwendung angezeigt bekommen, so daß ich nicht die Suche Verlassen muss

## === Acceptancetests

### ==== User calls Calculator

```
[cols="1,10"]
```

```
|===
```

```
| Given | User is on the main page
```

```
| When  | User enters 'calc' as search-term and submits the form
```

```
| Then  | the search result displays a calculator
```

```
|===
```

## = Spezifikation

```
include::config.adoc[]
```

Dieses Projekt soll den Benutzer durch zusätzliche Features länger auf der Suchseite halten um höhere Werbeinnahmen zu generieren.

## == Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des Suchbegriffs "Rechner" oder bei der Eingabe einer Mathematischen Rechnung einen Taschenrechner als Web-Anwendung angezeigt bekommen so daß ich nicht die Suche verlasse

## === Acceptancetests

### ==== User calls Calculator

```
[cols="1,10"]
```

```
|===
```

```
| Given | User is on the main page  
| When  | User enters 'calc' as search-term and submits the form  
| Then  | the search result displays a calculator
```

```
|===
```

```
@Ignore("not implemented")  
def "User calls Calculator"() {  
  given: "User is on the main page"  
  when: "User enters 'calc' as search-term and submits the form"  
  then: "the search result displays a calculator"  
}
```

## 2. Report for acceptance.Specification2bSpec

This test uses the calculator feature of Google to showcase the Geb-Reports

### 2.1. Summary

Total Runs	Success Rate	Total time	Failures	Errors	Skipped
0	100,0%	0,251 seconds	0	0	2

[acceptance/Specifaction2bSpec.groovy](#)

### 2.2. Features

#### 2.2.1. IGNORED: User calls Calculator

not implemented

Given: User is on the main page

When: User enters 'calc' as search-term and submits the form

Then: the search result displays a calculator

## = Spezifikation

```
include::config.adoc[]
```

Dieses Projekt soll den Benutzer durch zu länger auf der Suchseite halten um höhere generieren.

## == Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des S oder bei der Eingabe einer Mathematischen Taschenrechner als Web-Anwendung angezeig so daß ich nicht die Suche Verlassen muss

## === Acceptancetests

```
include::{pathToSpec}acceptance.Spec  
tag=User_calls_Calculator]
```

# Spezifikation

Dieses Projekt soll den Benutzer durch zusätzliche Features länger auf der Suchseite halten um höhere Werbeinnahmen zu generieren.

## 1. Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des Suchbegriffs "Rechner" oder bei der Eingabe einer Mathematischen Rechnung einen Taschenrechner als Web-Anwendung angezeigt bekommen, so daß ich nicht die Suche Verlassen muss.

### 1.1. Acceptancetests

#### 1.1.1. IGNORED: User calls Calculator

not implemented

Given: **User is on the main page**

When: **User enters 'calc' as search-term and submits the form**

Then: **the search result displays a calculator**

## Spezifikation

Dieses Projekt soll den Benutzer durch zusätzliche Features länger auf der Suchseite halten um höhere Werbeinnahmen zu generieren.

### 1. Story: Taschenrechner

Als Benutzer möchte ich bei Eingabe des Suchbegriffs "Rechner" oder bei der Eingabe einer Mathematischen Rechnung einen Taschenrechner als Web-Anwendung angezeigt bekommen, so daß ich nicht die Suche Verlassen muss

#### 1.1. Akzeptanztest

##### 1.1.1. PASS: Bing-Rechner aufrufen

When: die Bing Homepage aufgerufen wird

to BingHomePage

Then: wird die Bing Homepage angezeigt

```
report "Bing Home Page"
at BingHomePage
```

When: nach 'calc' gesucht wird

```
form.q = 'calc'
sleep(200)
searchBtn.click()
```

Then: erscheint die Suchresultseite

```
report "Bing Search Result Page"
at BingSearchResultPage
```

Table 1. Screenshots:



Figure 1. Bing Home Page

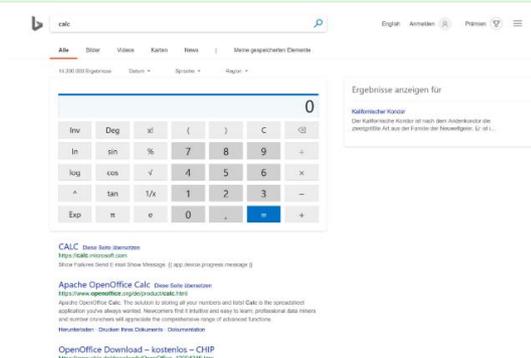


Figure 2. Bing Search Result Page





15:00

15:45



**Dokumentation in der IDE  
schreiben? Klar, mit  
AsciiDoc!**

Track 2

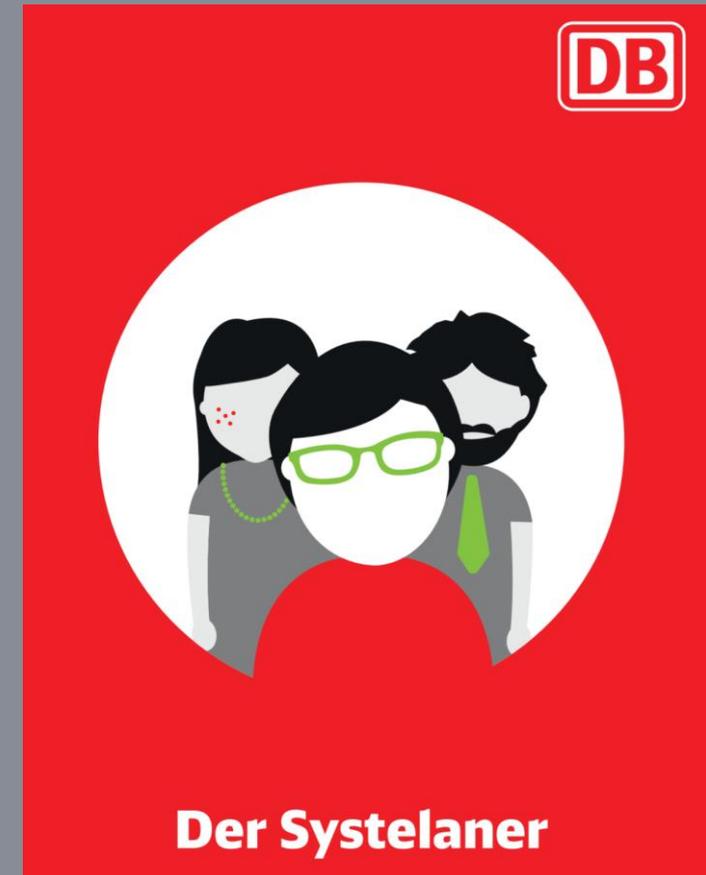


Vielen Dank für Ihre Aufmerksamkeit!



<https://rdmueller.github.io/feedback>

Ralf.D.Mueller@DeutscheBahn.com



[www.dbsystel.de](http://www.dbsystel.de)