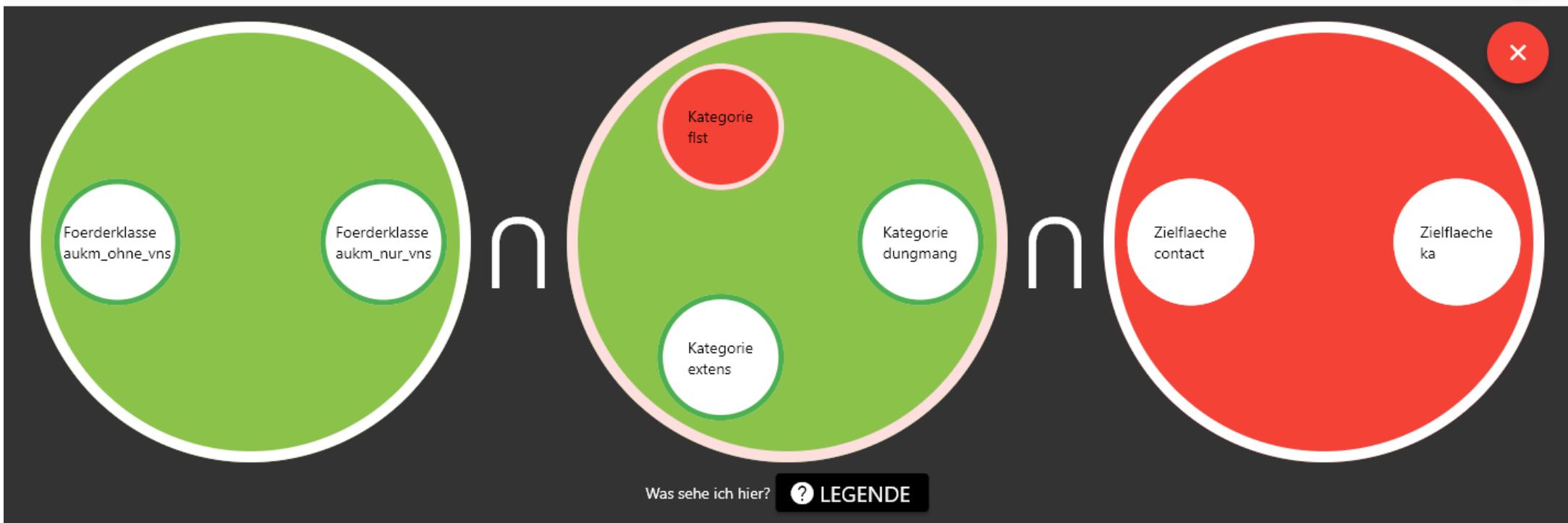


[← Zieleinheit](#)

- keine Angabe/Vorgabe (i)
- ha (i)
- bitte um Unterstützung (i)
- m³ (z.B. Gülle) (✓)





Entwicklung einer Formularanwendung  
mit Kompatibilitätsvalidierung der  
Einfach- und Mehrfachauswahl-Eingabefelder

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung 

**Identifikatoren**

Maßnahmentitel

**Maßnahmencharakteristika**

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz 

Kategorie 

Zielsetzung

Zielfläche 

Zieleinheit 

Hauptzielsetzung Land 

Nebenziele 

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie

Zielsetzung

Zielfläche

Zieleinheit

Hauptzielsetzung Land

Nebenziele

- ← Kategorie DEBUG
- Anbau Zwischenfrucht/Untersaat
  - Anlage/Pflege Struktur
  - Düngemanagement
  - Extensivierung
  - Flächenstilllegung/Brache
  - Nutzungsumwandlung
  - Förderung bestimmter Rassen / Sorten / Kulturen
  - bitte um Unterstützung
- ✓

## Maßnahmen Detail

Status  
in Bearbeitung

### Identifikatoren

Maßnahmentitel

### Maßnahmencharakteristika

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie

### Zielsetzung

Zielfläche

Zieleinheit

Hauptzielsetzung Land

Nebenziele

## Kategorie

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit

Hauptzielsetzung Land

Nebenziele

- ← Kategorie DEBUG
- Anbau Zwischenfrucht/Untersaat
  - Anlage/Pflege Struktur
  - Düngemanagement
  - Extensivierung
  - Flächenstilllegung/Brache
  - Nutzungsumwandlung
  - Förderung bestimmter Rassen / Sorten / Kulturen
  - bitte um Unterstützung
- ✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit

Hauptzielsetzung Land

Nebenziele

- ← Kategorie DEBUG
- Anbau Zwischenfrucht/Untersaat
  - Anlage/Pflege Struktur
  - Düngemanagement
  - Extensivierung
  - Flächenstilllegung/Brache
  - Nutzungsumwandlung
  - Förderung bestimmter Rassen / Sorten / Kulturen
  - bitte um Unterstützung
- ✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit

Hauptzielsetzung Land

Nebenziele

← Zieleinheit DEBUG

- keine Angabe/Vorgabe
- m<sup>3</sup> (z.B. Gülle)
- Kopf/Stück (z.B. Tiere oder Bäume)
- GV/GVE
- RGV
- ha
- bitte um Unterstützung

✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit

Hauptzielsetzung Land

Nebenziele

← Zieleinheit DEBUG

keine Angabe/Vorgabe

m<sup>3</sup> (z.B. Gülle)

Kopf/Stück (z.B. Tiere oder Bäume)

GV/GVE

RGV

ha

bitte um Unterstützung

✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit  
 m<sup>3</sup> (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

← Zieleinheit DEBUG

- keine Angabe/Vorgabe
- m<sup>3</sup> (z.B. Gülle)
- Kopf/Stück (z.B. Tiere oder Bäume)
- GV/GVE
- RGV
- ha
- bitte um Unterstützung

✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit  
 m<sup>3</sup> (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

← Zieleinheit DEBUG

- keine Angabe/Vorgabe
- m<sup>3</sup> (z.B. Gülle)
- Kopf/Stück (z.B. Tiere oder Bäume)
- GV/GVE
- RGV
- ha
- bitte um Unterstützung

✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit  
 $m^3$  (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

- ← Kategorie DEBUG
- Anbau Zwischenfrucht/Untersaat
  - Anlage/Pflege Struktur
  - Düngemanagement
  - Extensivierung
  - Flächenstilllegung/Brache
  - Nutzungsumwandlung
  - Förderung bestimmter Rassen / Sorten / Kulturen
  - bitte um Unterstützung
- ✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit  
 $m^3$  (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

- ← Kategorie DEBUG
- Anbau Zwischenfrucht/Untersaat
  - Anlage/Pflege Struktur
  - Düngemanagement
  - Extensivierung
  - Flächenstilllegung/Brache
  - Nutzungsumwandlung
  - Förderung bestimmter Rassen / Sorten / Kulturen
  - bitte um Unterstützung
- ✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Flächenstilllegung/Brache

Zielsetzung

Zielfläche

Zieleinheit  
 $m^3$  (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

- ← Kategorie DEBUG
- Anbau Zwischenfrucht/Untersaat
  - Anlage/Pflege Struktur
  - Düngemanagement
  - Extensivierung
  - Flächenstilllegung/Brache
  - Nutzungsumwandlung
  - Förderung bestimmter Rassen / Sorten / Kulturen
  - bitte um Unterstützung
- ✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Flächenstilllegung/Brache

Zielsetzung

Zielfläche

Zieleinheit  
 $m^3$  (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

- ← Kategorie DEBUG
- Anbau Zwischenfrucht/Untersaat
  - Anlage/Pflege Struktur
  - Düngemanagement
  - Extensivierung
  - Flächenstilllegung/Brache
  - Nutzungsumwandlung
  - Förderung bestimmter Rassen / Sorten / Kulturen
  - bitte um Unterstützung
- ✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Flächenstilllegung/Brache

Zielsetzung

Zielfläche

Zieleinheit  
**m<sup>3</sup> (z.B. Gülle)**

Hauptzielsetzung Land

Nebenziele

← Zieleinheit DEBUG

keine Angabe/Vorgabe

**m<sup>3</sup> (z.B. Gülle)**

Kopf/Stück (z.B. Tiere oder Bäume)

GV/GVE

RGV

ha

bitte um Unterstützung

✓

### Maßnahmen Detail

Status  
in Bearbeitung

#### Identifikatoren

Maßnahmentitel

#### Maßnahmencharakteristika

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Flächenstilllegung/Brache

#### Zielsetzung

Zielfläche

Zieleinheit  
 m<sup>3</sup> (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

### Zieleinheit

- keine Angabe/Vorgabe
- Kopf/Stück (z.B. Tiere oder Bäume)
- GV/GVE
- RGV
- ha
- bitte um Unterstützung
- m<sup>3</sup> (z.B. Gülle)

✓

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Flächenstilllegung/Brache

Zielsetzung

Zielfläche

Zieleinheit

Hauptzielsetzung Land

Nebenziele

← Zieleinheit DEBUG

keine Angabe/Vorgabe

Kopf/Stück (z.B. Tiere oder Bäume)

GV/GVE

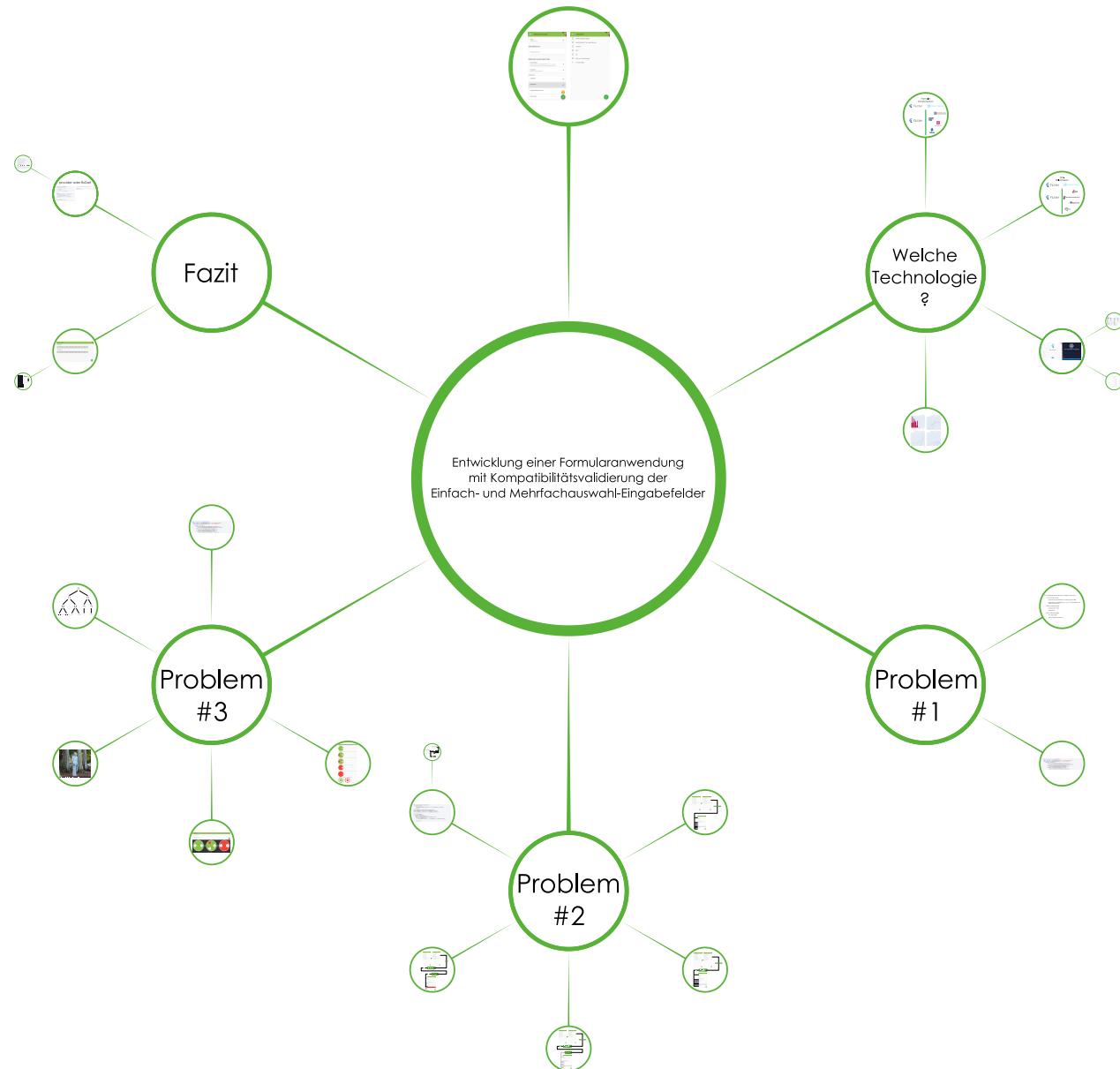
RGV

ha

bitte um Unterstützung

✗ m<sup>3</sup> (z.B. Gülle)

✓



# Welche Technologie ?



# Problem #1

m

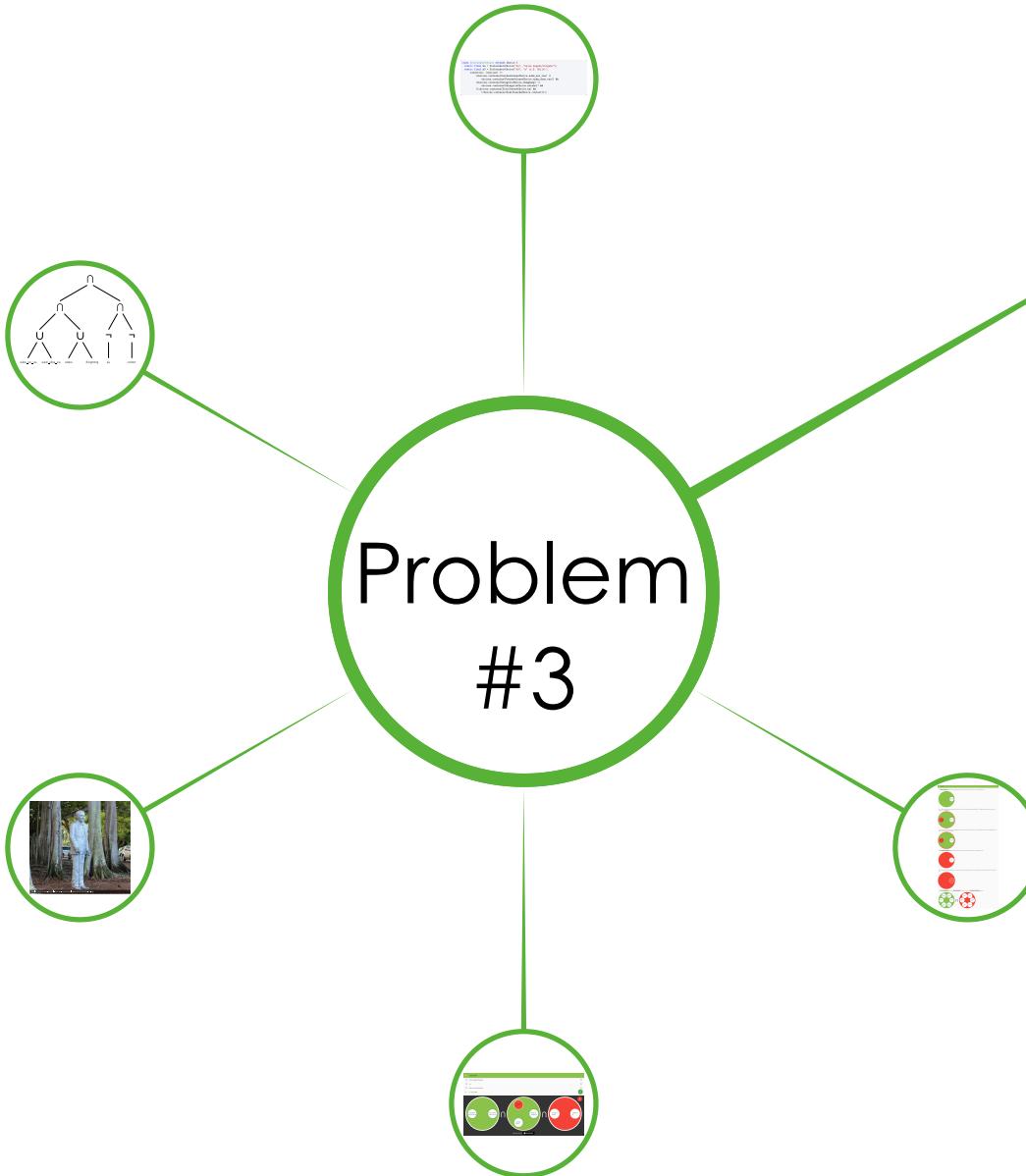


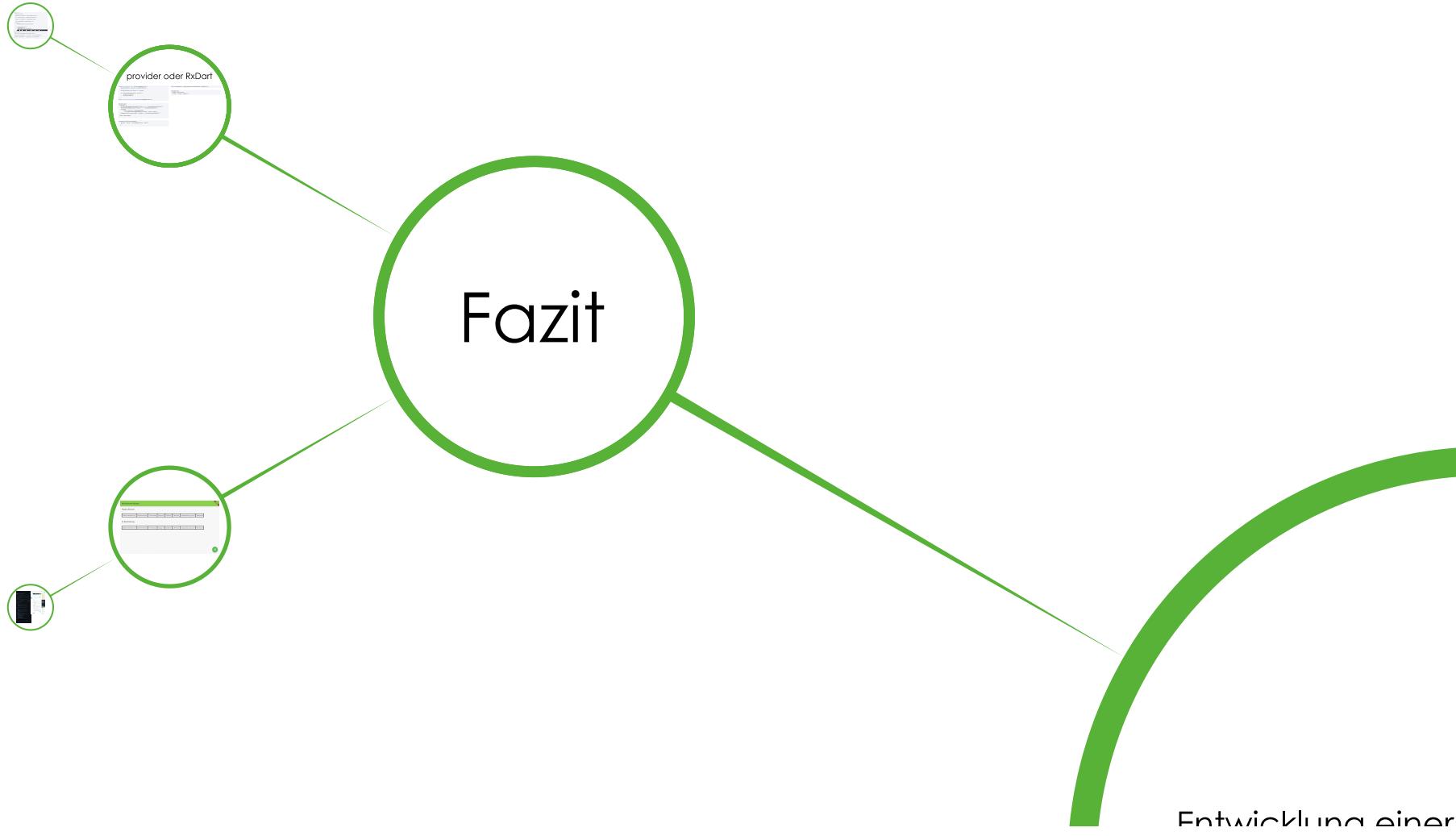
## Problem #2



EINHÄNGLING

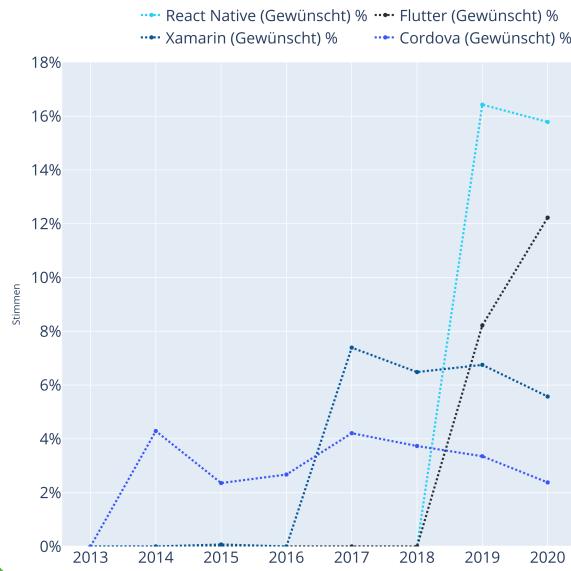
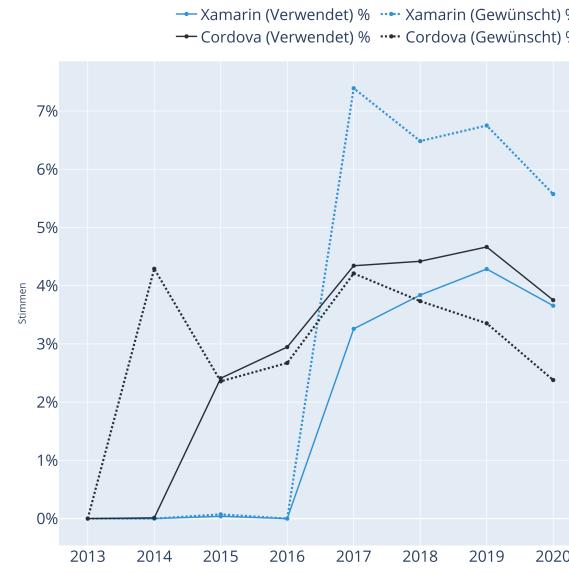
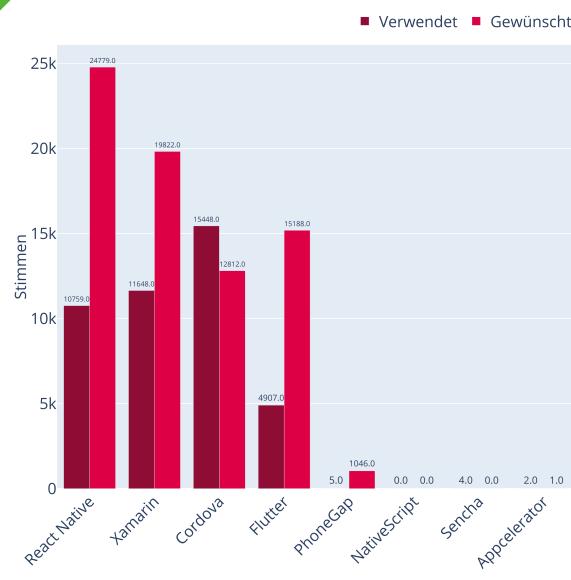
# Problem #3

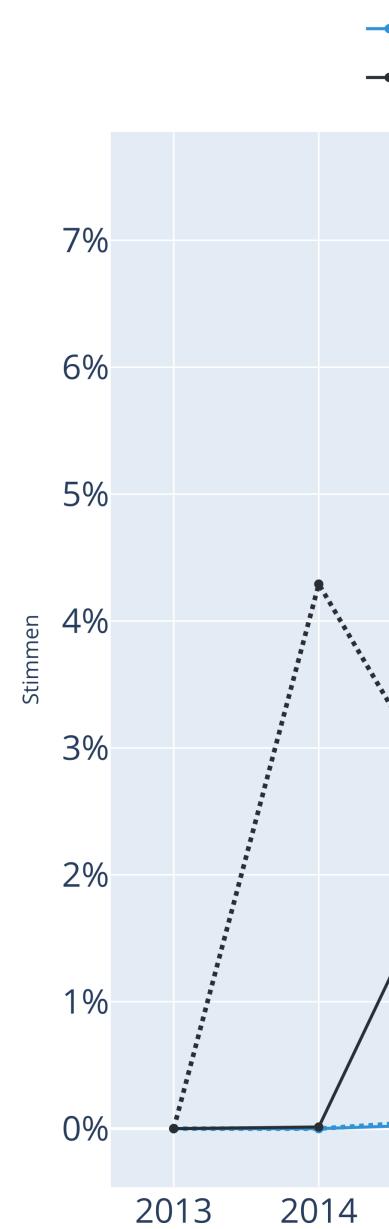
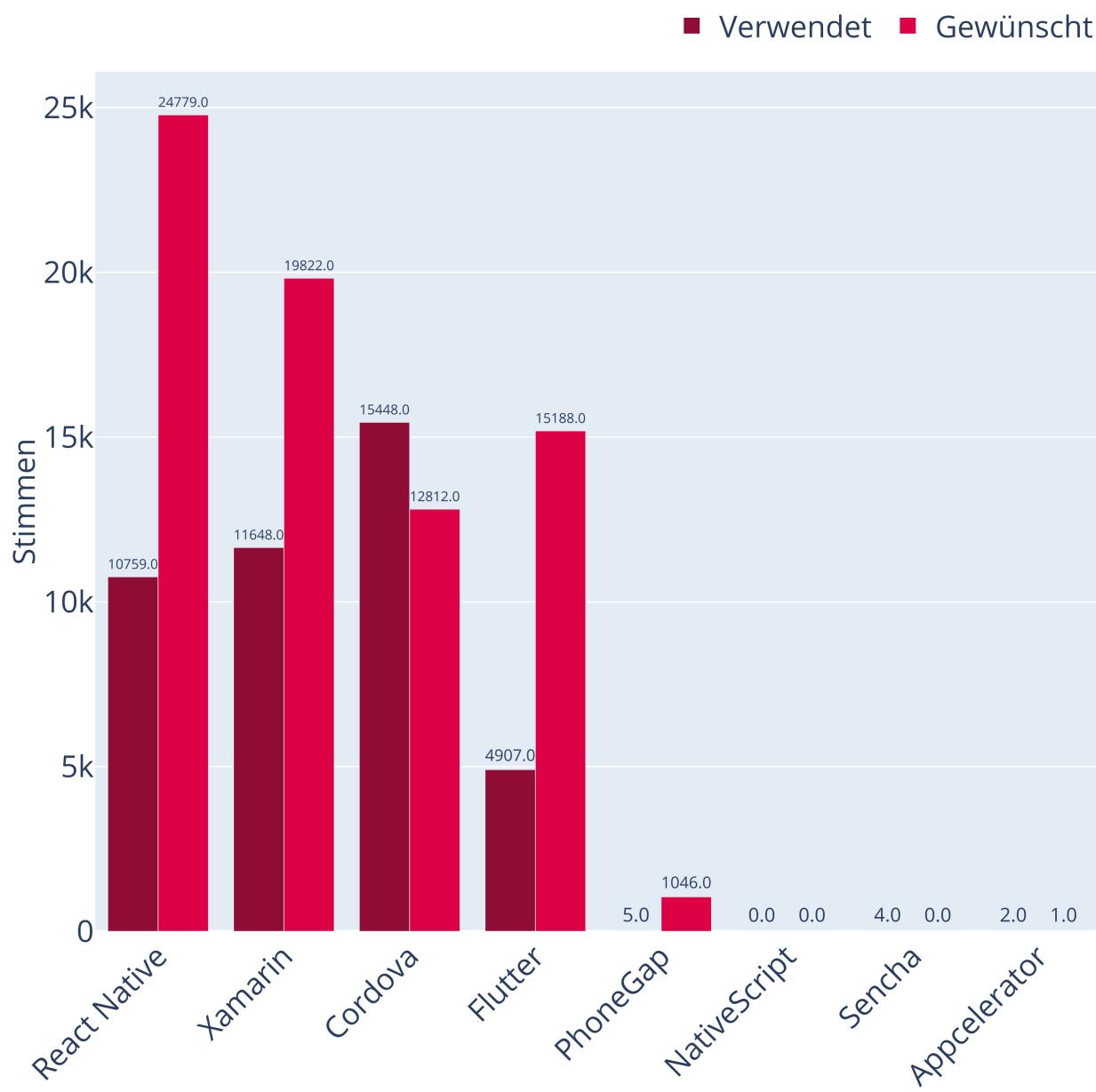




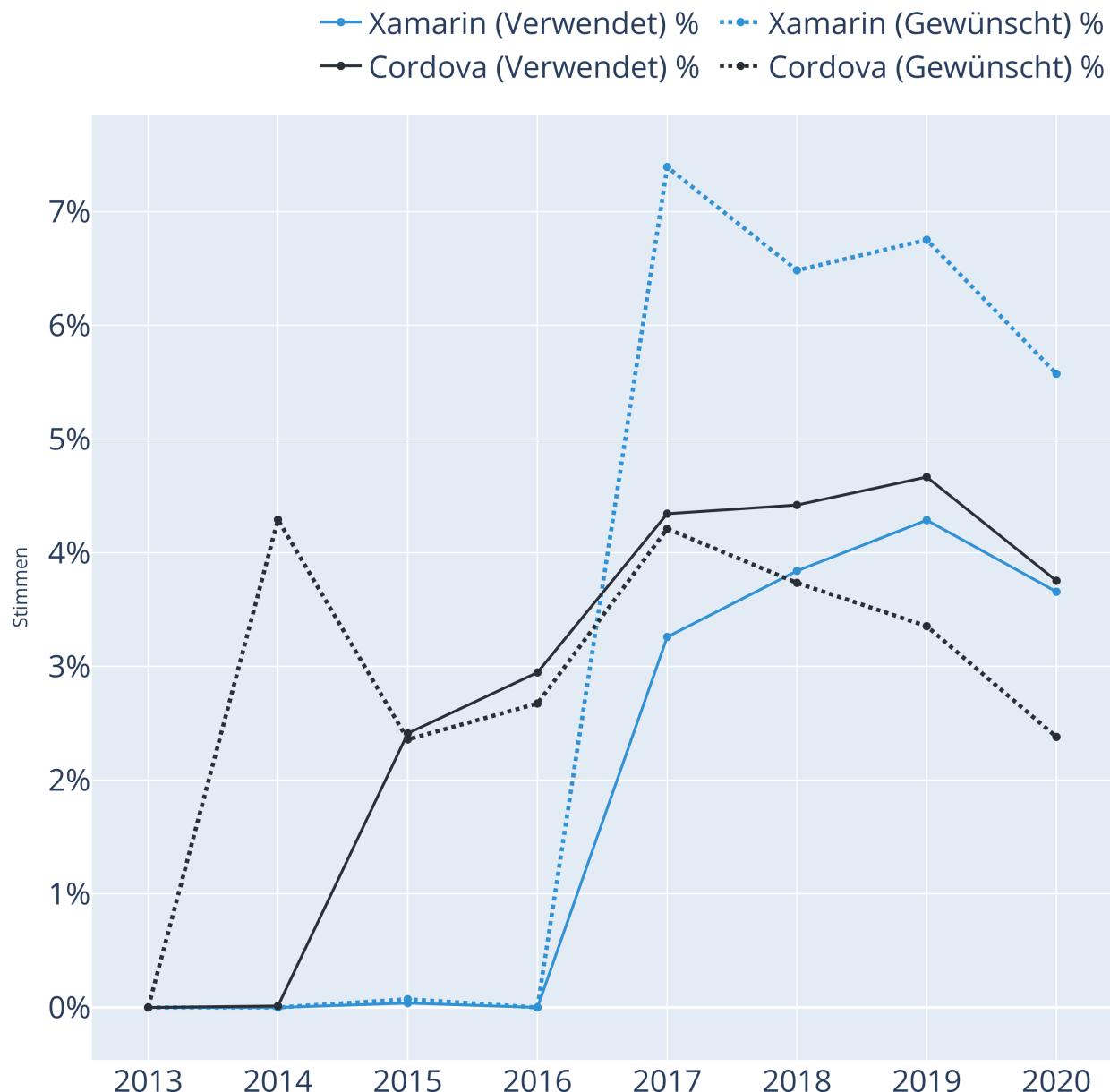
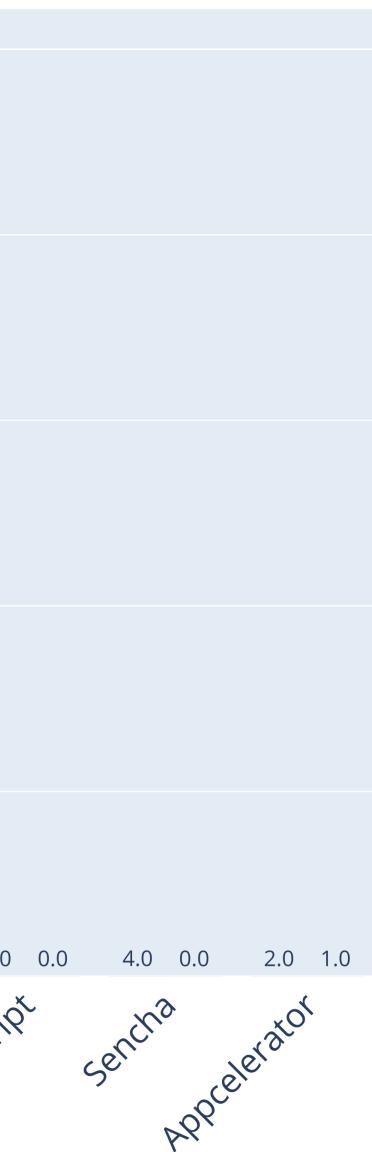
# Welche Technologie ?

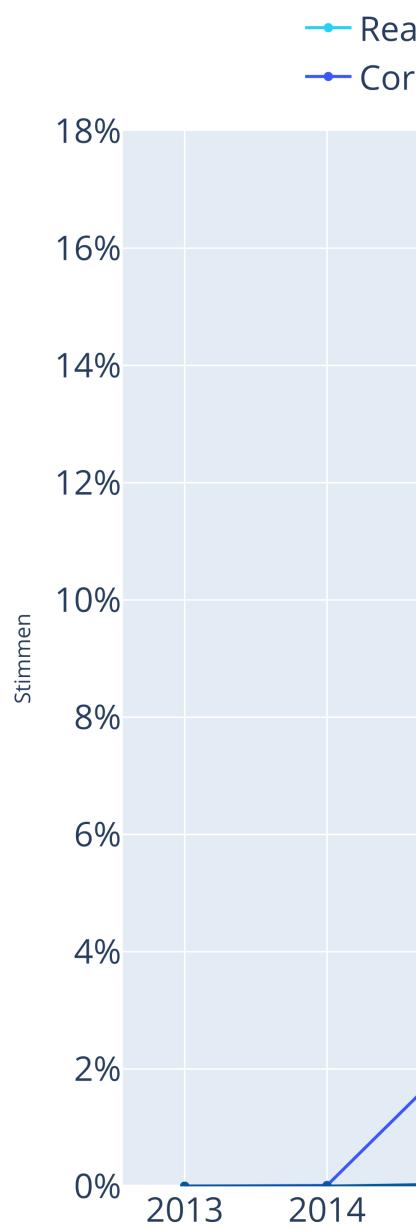
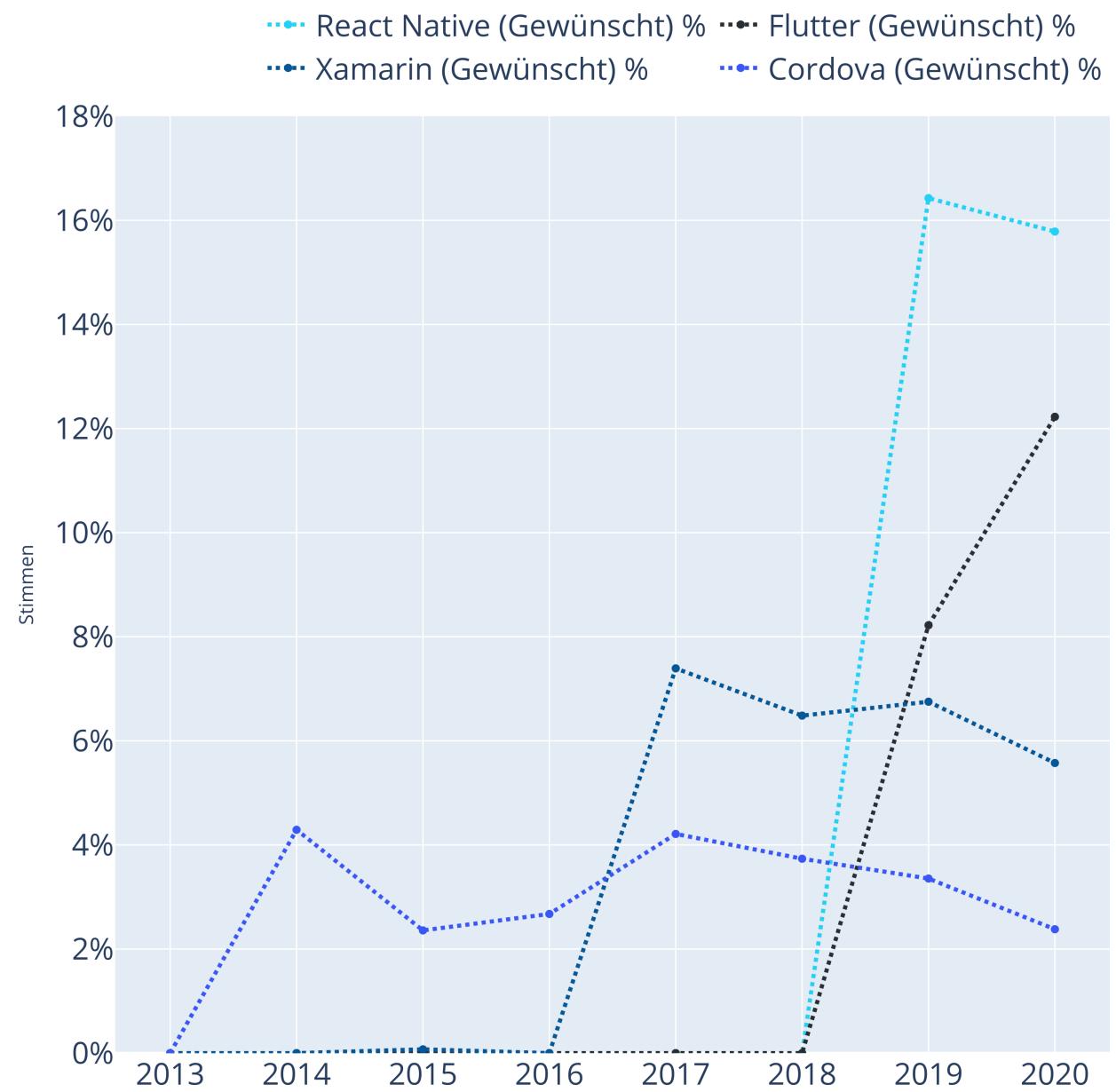




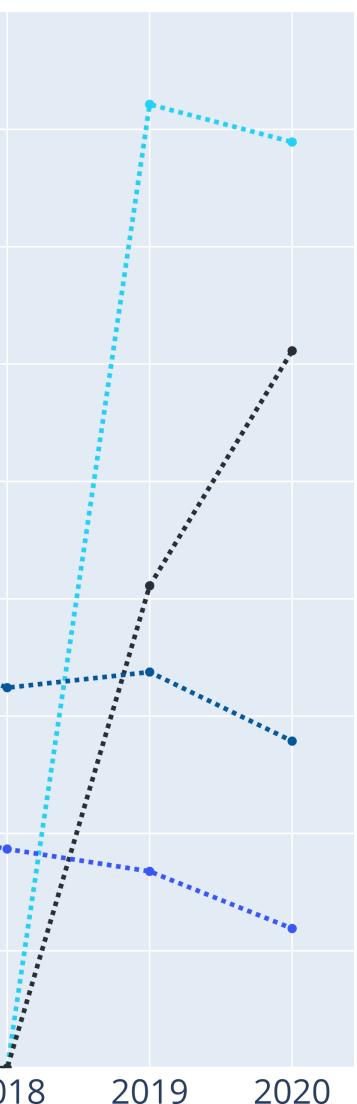


wendet ■ Gewünscht





Flutter (Gewünscht) %  
Cordova (Gewünscht) %

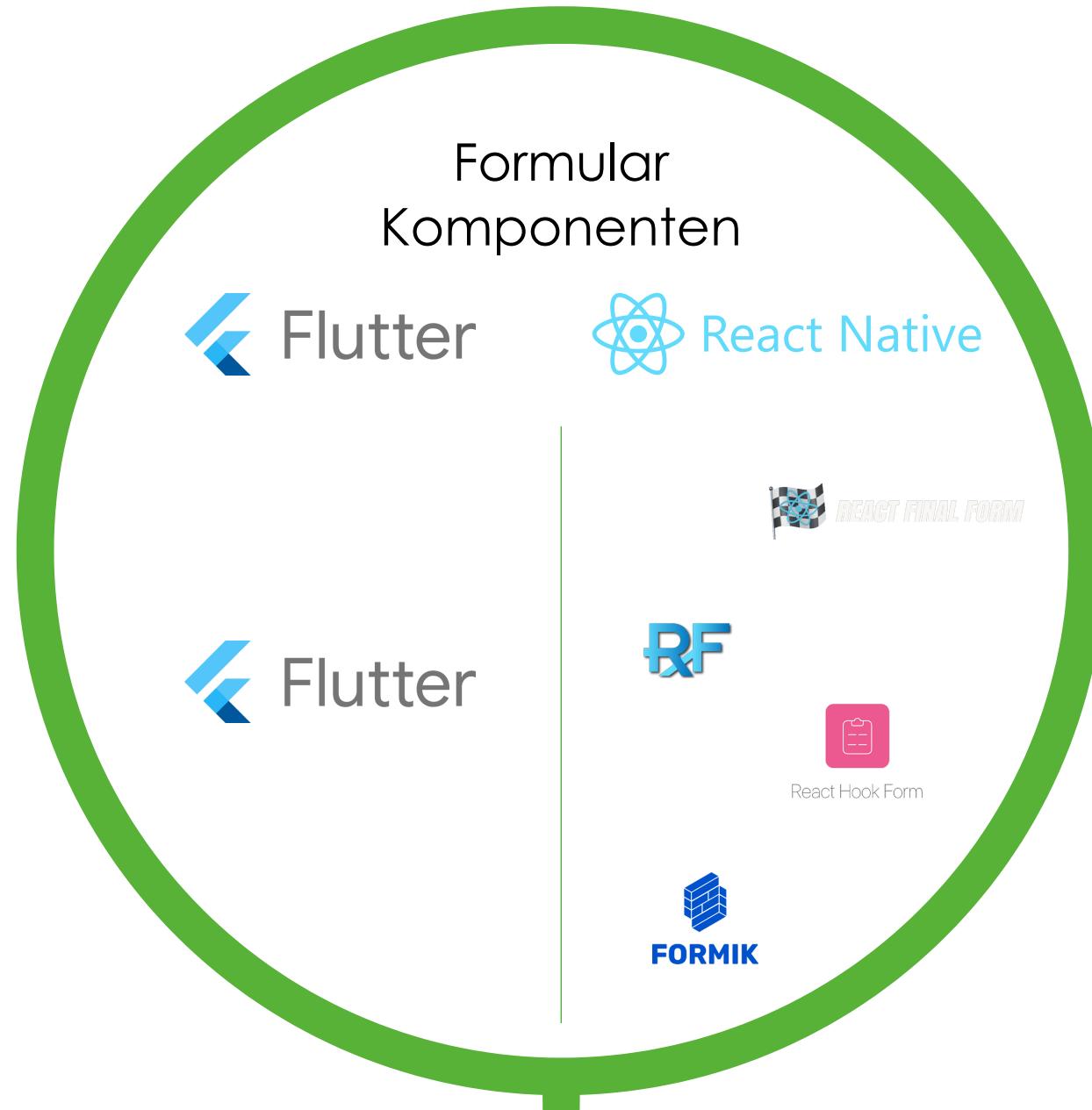


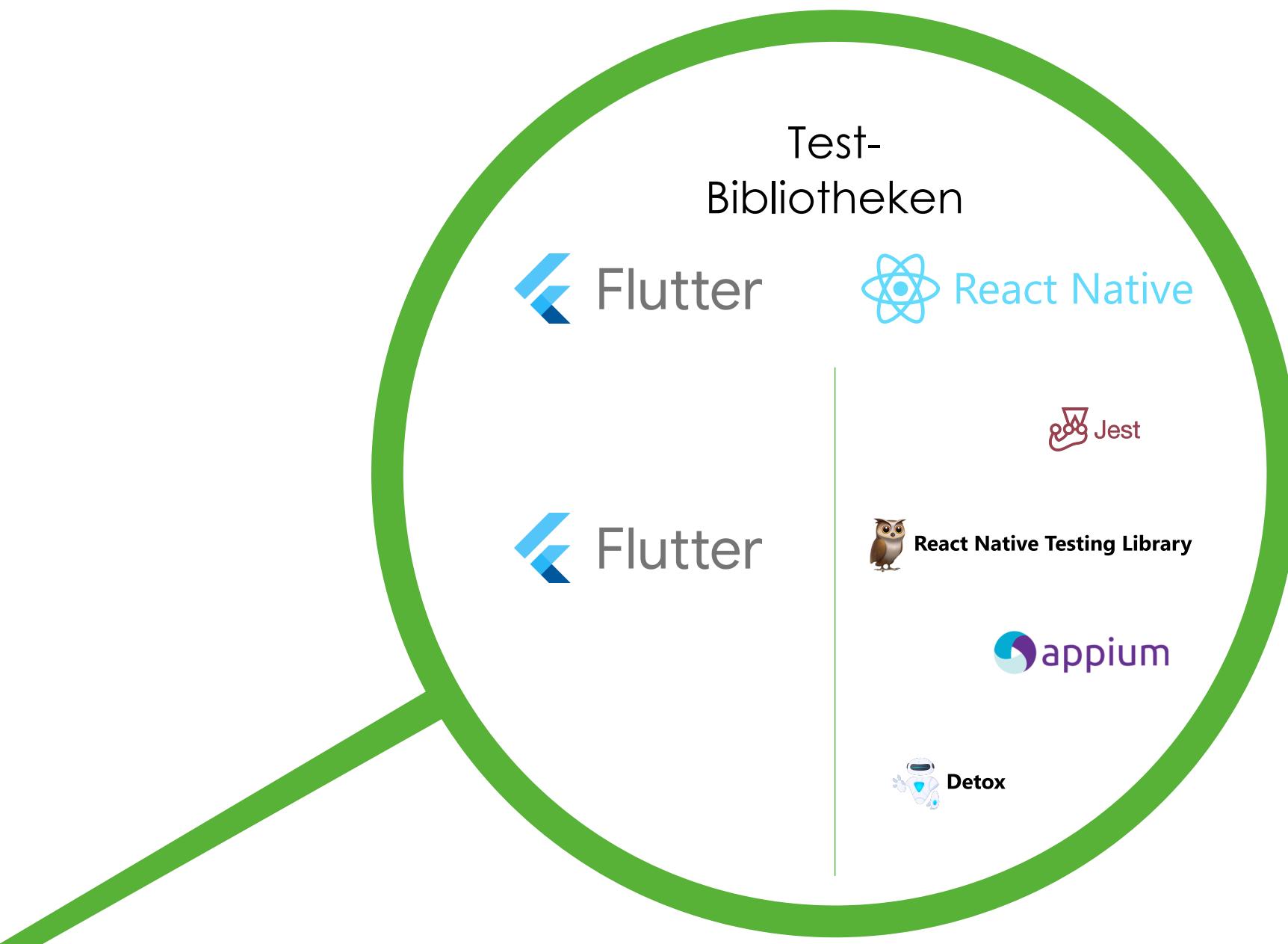
React Native (Verwendet) %    Flutter (Verwendet) %  
Cordova (Verwendet) %    Xamarin (Verwendet) %

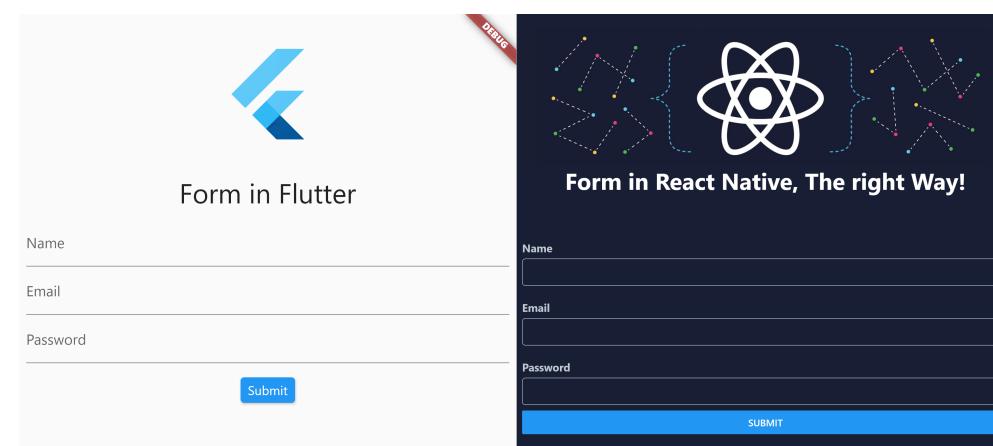


# Welche Technologie ?

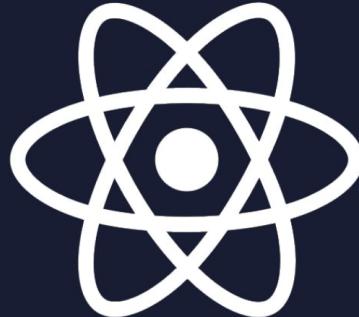








DEBUG



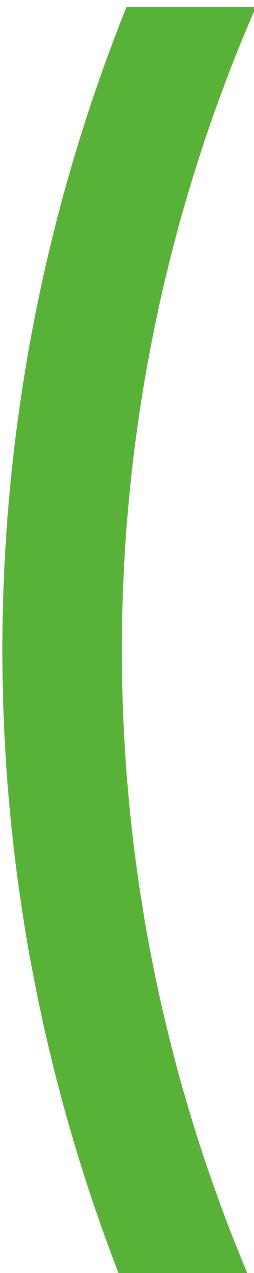
**Form in React Native, The right Way!**

Name

Email

Password

SUBMIT



DEBUG

# Form in Flutter

Name

---

Email

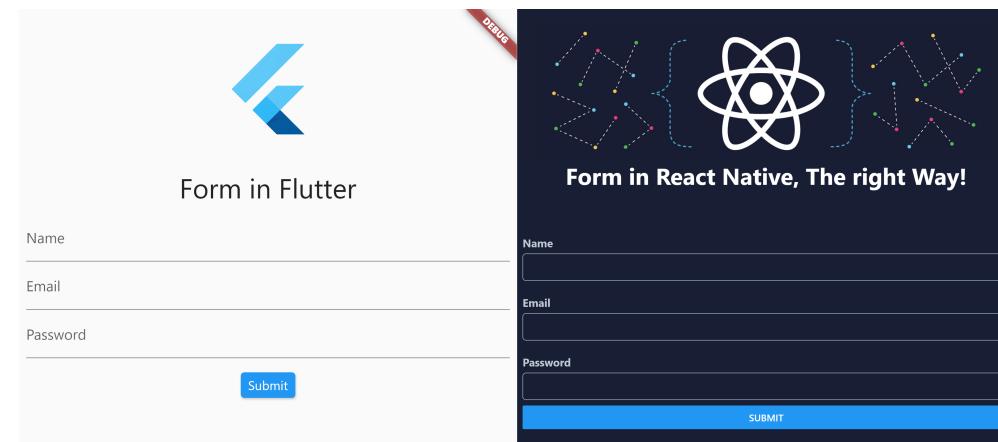
---

Password

---

Submit







# React Native

```
1 export default {
2   name: { required: true, message: 'Name is required' },
3   email: {
4     required: true, message: 'Email is required',
5     pattern: '^[_a-zA-Z0-9]+([._][_a-zA-Z0-9]+)*@[a-zA-Z0-9.-]+\\.[a-zA-Z]{2,3}$',
6     value: (([0-9]{1,3}).([0-9]{1,3}).([0-9]{1,3})|([0-9]{1,3})\\.)(((za-zA-Z0-9){1,3})$),
7     message: 'Invalid Email Format',
8   },
9 }, password: {
10   required: true, message: 'Password is required',
11 },
12 };
13 
```

5 interface ValidationMap {  
6 [key: string]: ValidationOptions;  
7 }  
8  
9 interface ErrorMap {  
10 [key: string]: FieldError | undefined;  
11 }  
12  
13 interface Props {  
14 children: JSX.Element | JSX.Element[];  
15 register: ValidationMap;  
16 errors: ErrorMap;  
17 validation?: ValidationOptions;  
18 }> void;  
19 errors: ErrorMap;  
20 validation: ValidationMap;  
21 setValue: (name: string, value: string, validate?: boolean) => void;  
22 }  
23  
24 export default ({  
25 register,  
26 errors,  
27 setValue,  
28 validation,  
29 children,  
30 }: Props) => {  
31 const Inputs = React.useRef<Array<TextInput>>([]);  
32  
33 React.useEffect(() => {  
34 (Array.isArray(children) ? [...children] : [children]).forEach((child) => {  
35 if (child.props.name)  
36 register({ name: child.props.name }, validation[child.props.name]);  
37 });
38 }, [register]);  
39  
40 return (  
41 <>  
42 {Array.isArray(children) ? [...children] : [children]}.map(  
43 (child, id) => {  
44 return child.props.name  
45 ? React.createElement(child.type, {  
46 ...  
47 ...child.props,  
48 ref: (e: TextInput) => {  
49 Inputs.current[id] = e;  
50 },  
51 onChangeText: (v: string) =>  
52 setValue(child.props.name, v, true),  
53 onSubmitEditing: () => {  
54 Inputs.current[id + 1]  
55 ? Inputs.current[id + 1].focus()  
56 : Inputs.current[id].blur();  
57 },  
58 onBlur: () => triggerValidation(child.props.name),  
59 blurOnSubmit: false,  
60 name: child.props.name,  
61 error: errors[child.props.name],  
62 },  
63 )  
64 : child;  
65 })  
66 </>  
67 );
68 );
69 };  
70

```
14
15 const styles = StyleSheet.create({
16   container: {
17     justifyContent: 'center',
18     flex:1,
19   },
20   paragraph: {
21     margin: 24,
22     marginTop: 0,
23     fontSize: 34,
24     fontWeight: 'bold',
25     textAlign: 'center',
26     color: '#FFF'
27   },
28   logo: {
29     width: '100%',
30     height: 200
31   }
32 });
```



Flutter	Programmcode	main.dart	60
		validation.dart	15
		<b>Summe</b>	<b>75</b>
React Native	Programmcode	App.tsx	15
		Hero.tsx	9
		Input.tsx	17
		Form.tsx	52
		validation.tsx	12
		<b>Summe</b>	<b>112</b>
	Stylesheets	App.tsx	14
		Hero.tsx	17
		Input.tsx	24
		<b>Summe</b>	<b>55</b>
<b>Summe</b>		<b>167</b>	



# Welche Technologie ?



anwendung

# Problem

## #1

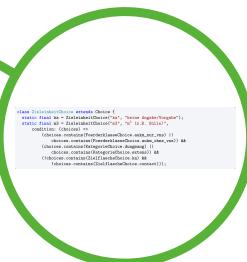
Um im Auswahlfeld der Zeileinheit die Option „auswählen“ zu können m...

- für die Förderlinie entweder
    - Agrarwelt und Klima/Maßnahmen: nur Vergrößerungsschutz oder
    - Agrarpolitik und Klima/Maßnahmen, ne. auch mit Tierwohlspekz., aber OHNE Vergrößerungsschutz
  - und für die Faktenlinie entweder
    - Organisationsmodell oder
    - Erweiterung
  - und für Ziellinie entweder
    - keine Angabe, noch
    - bitte um Unterstützung geehrt sein.

Um im Auswahlfeld der *Zieleinheit* die Option *m3* auswählen zu können muss:

- für die *Förderklasse* entweder
  - *Agrarumwelt-(und Klima)Maßnahme: nur Vertragsnaturschutz oder*
  - *Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz*
- **und** für die *Kategorie* entweder
  - *Düngemanagement oder*
  - *Extensivierung*
- **und** für *Zielfläche* darf **weder**
  - *keine Angabe, noch*
  - *bitte um Unterstützung gewählt sein.*

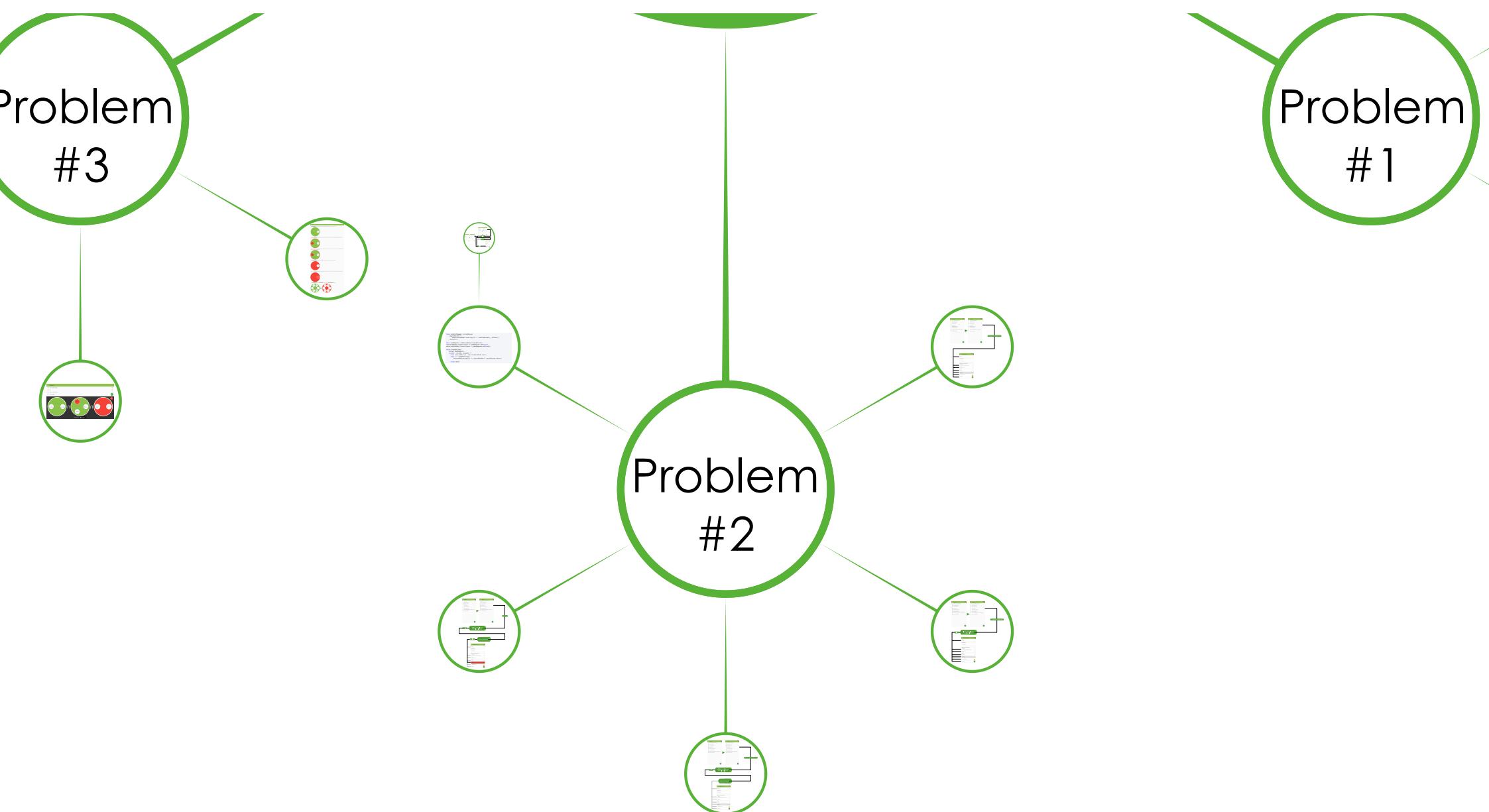
# Problem #1



```
class ZieleinheitChoice extends Choice {  
    static final ka = ZieleinheitChoice("ka", "keine Angabe/Vorgabe");  
    static final m3 = ZieleinheitChoice("m3", "m3 (z.B. Gülle)",  
        condition: (choices) =>  
            (choices.contains(FoerderklasseChoice.aukm_nur_vns) ||  
                choices.contains(FoerderklasseChoice.aukm_ohne_vns)) &&  
            (choices.contains(KategorieChoice.dungmang) ||  
                choices.contains(KategorieChoice.extens)) &&  
            (!choices.contains(ZielflaecheChoice.ka) &&  
                !choices.contains(ZielflaecheChoice.contact)));
```

# Problem #1

- Um die Auswirkungen der Zulassungen die Optionen erweitern zu können reicht:
  - für die Fertigungsteile zu erneutet
    - Anpassungen und Kleinabnahmen, nur Herstellergeschützt oder
    - Änderungen und Kleinabnahmen, die nach Art. Tonvorschriften, OHNE Herstellergeschützt
  - mit der für die Autopartei eindeutig
    - Drogenmanagement oder
    - Alkoholmanagement
  - mit der Zulassung darf wieder
    - keine Angabe, auch
    - Zahl zur Unterscheidung gewählt sein



The image shows a large green circle and a green Y-shaped arrow pointing downwards, enclosing two screenshots of a software interface. The top screenshot displays a list of categories with checkboxes. The bottom screenshot shows a detailed view of a measure with various input fields.

**Screenshot 1: Kategorie**

Kategorie
<input type="checkbox"/> Anbau Zwischenfrucht/Untersaat
<input type="checkbox"/> Anlage/Pflege Struktur
<input type="checkbox"/> DüngeManagement
<input checked="" type="checkbox"/> Extensivierung
<input type="checkbox"/> Flächenentlegung/Brache
<input type="checkbox"/> Nutzungsumwandlung
<input type="checkbox"/> Förderung bestimmter Rassen / Sorten / Kulturen
<input type="checkbox"/> bitte um Unterstützung

**Screenshot 2: Kategorie**

Kategorie
<input type="checkbox"/> Anbau Zwischenfrucht/Untersaat
<input type="checkbox"/> Anlage/Pflege Struktur
<input checked="" type="checkbox"/> DüngeManagement
<input type="checkbox"/> Extensivierung
<input type="checkbox"/> Flächenentlegung/Brache
<input type="checkbox"/> Nutzungsumwandlung
<input type="checkbox"/> Förderung bestimmter Rassen / Sorten / Kulturen
<input type="checkbox"/> bitte um Unterstützung

**Measures Detail**

Maßnahmen Detail
Status → Bearbeitung
Identifikatoren
Maßnahmetitel
Maßnahmencharakteristika
Förderklasse Agrarumwelt- und KlimaMaßnahmen, tv. auch mit finanzieller Beteiligung älterer ORNL Vertragsnaturschutz
Kategorie DüngeManagement
Zielsetzung
Zielfläche
Zielleistung in (ha) (Ges.)
Hauptzielbesetzung Land
Nebenziele

← Kategorie DEBUG

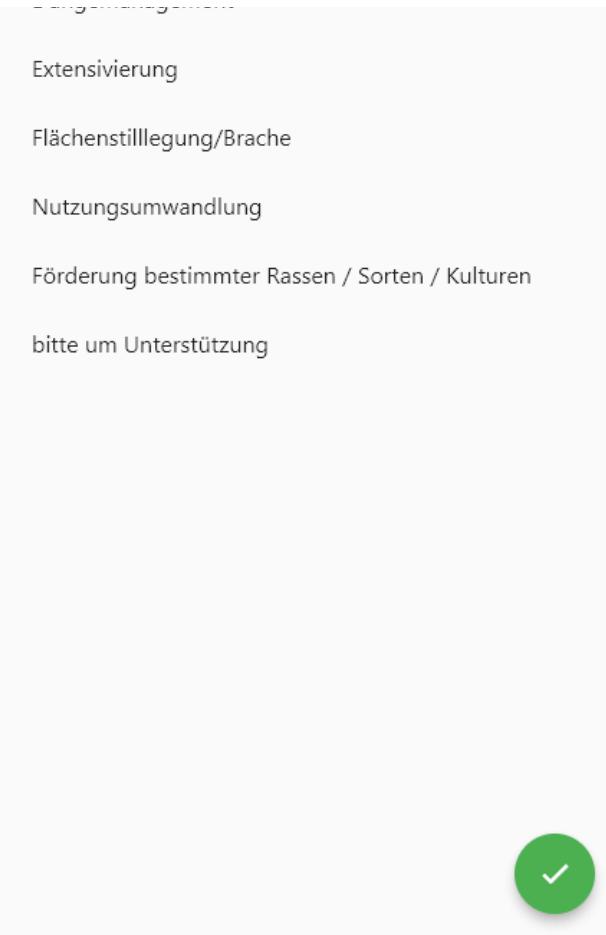
- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



← Kategorie DEBUG

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung





## Düngemanagement

Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Maßnahmencharakteristika

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

Zielsetzung

Zielfläche

Zieleinheit  
 $m^3$  (z.B. Gülle)

Hauptzielsetzung Land

Nebenziele

Kategorie

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement **✓**
- Extensivierung
- Flächenstiftung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

Kategorie

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement **✓**
- Extensivierung
- Flächenstiftung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

Maßnahmen Detail

Status  
in Bearbeitung

Identifikatoren

Maßnahmendst

Maßnahmencharakterika

Förderklasse  
Agrarwende- und Klimmaßnahmen, zw. auch mit Wirtschaftsgesetzen oder OHNE Beitragssatzschutz

Kategorie  
Düngemanagement

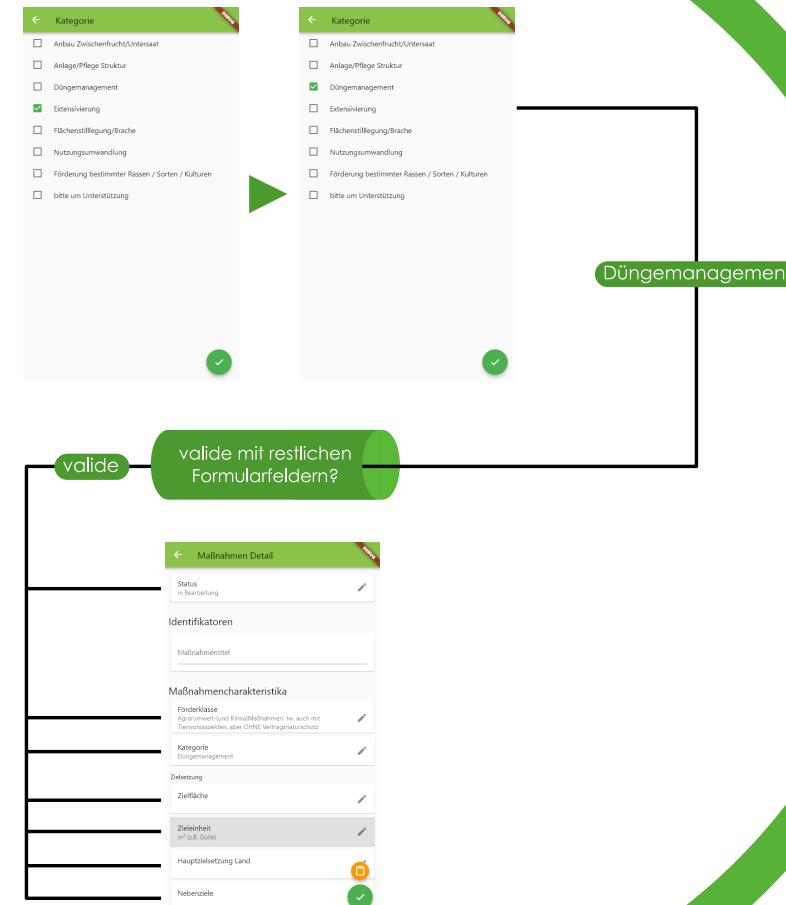
Zielsetzung

Zielfläche

Zielheit  
soz (z.B. Güte)

Hauptzielsetzung Land

Nebenziele



← Kategorie DEBUG

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



← Kategorie DEBUG

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

Dü

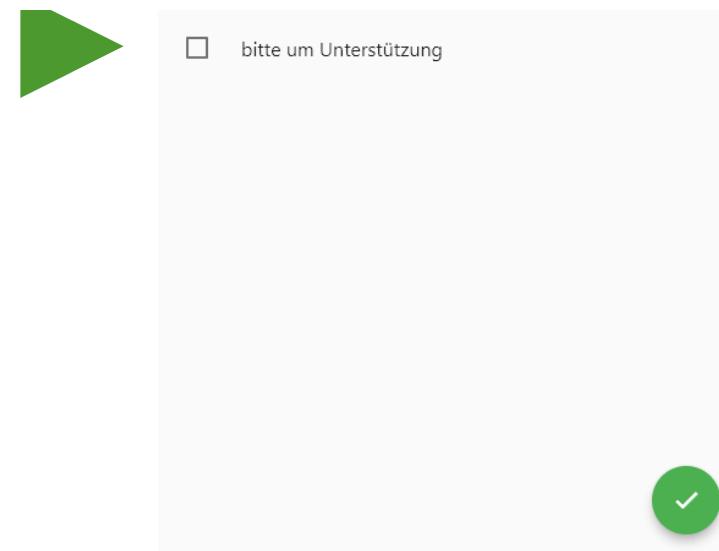
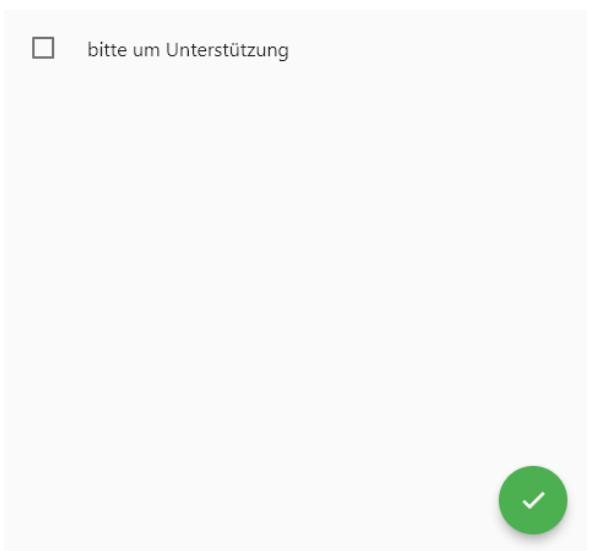


- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



Düngemanagement





Düngemanagement



valide

valide mit restlichen  
Formularfeldern?

← Maßnahmen Detail DEBUG

Status  
in Bearbeitung

Identifikatoren

Maßnahmentitel

Maßnahmencharakteristika

Förderklasse  
Agrarumwelt-(und Klima)Maßnahmen, tw. auch mit Tierwohlaspekten, aber OHNE Vertragsnaturschutz

Kategorie  
Düngemanagement

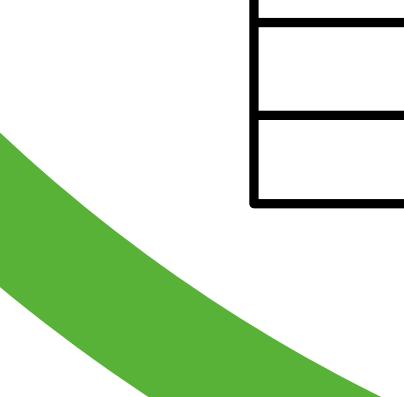
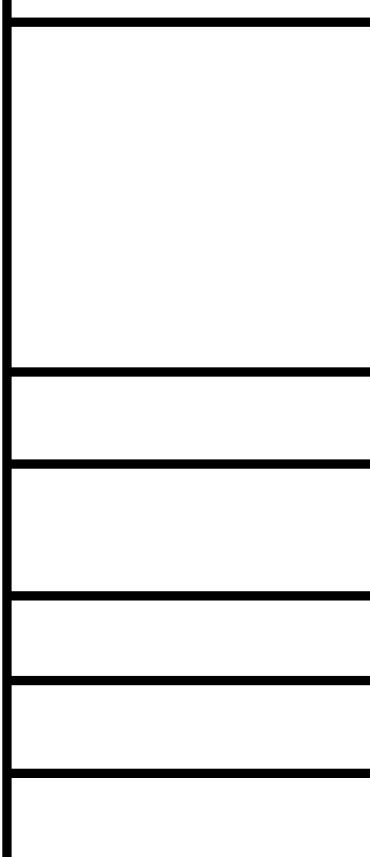
Zielsetzung

Zielfläche

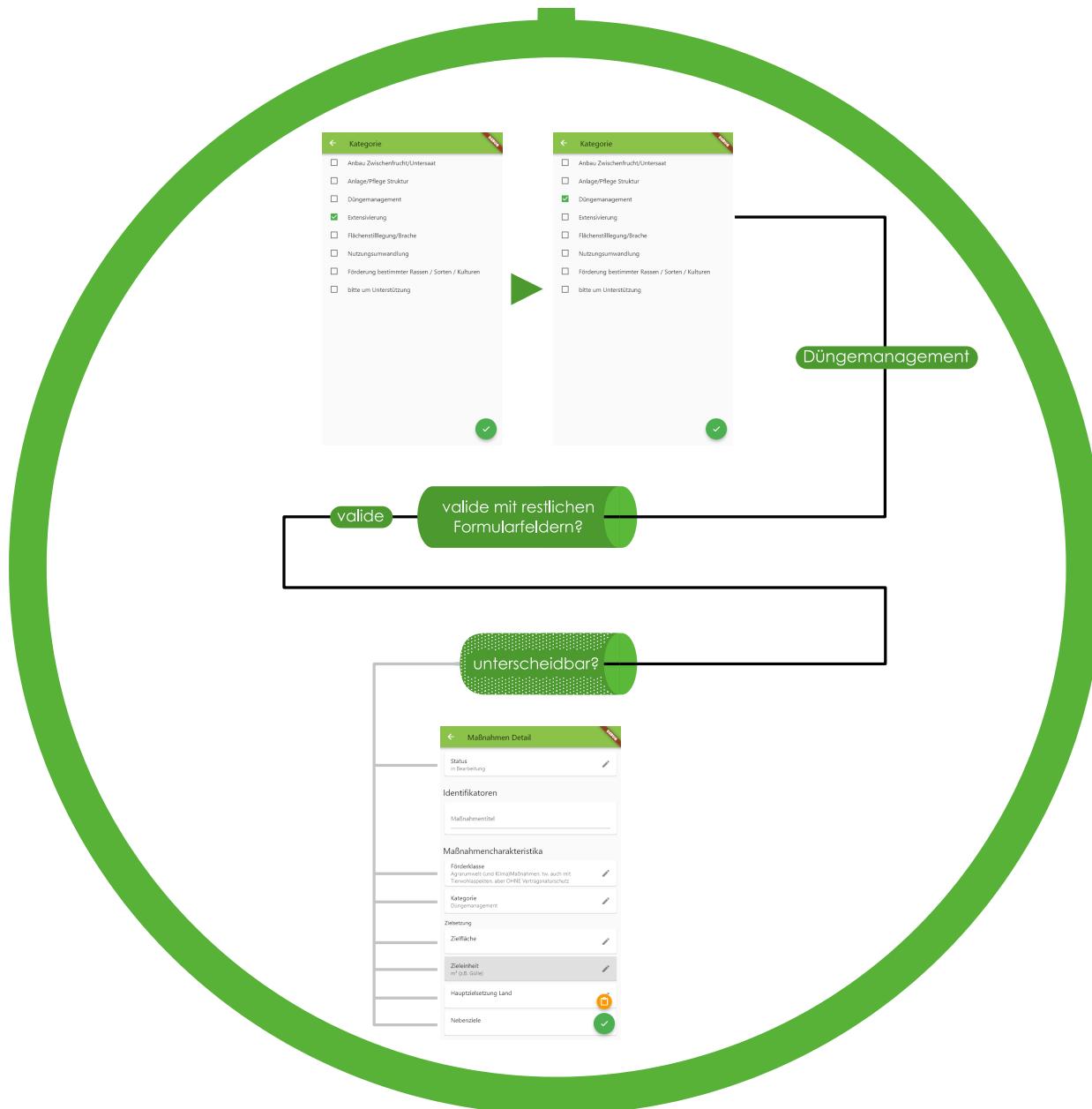
Zieleinheit  
 $m^3$  (z.B. Gülle)

Hauptzielsetzung Land 

Nebenziele 







Kategorie

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

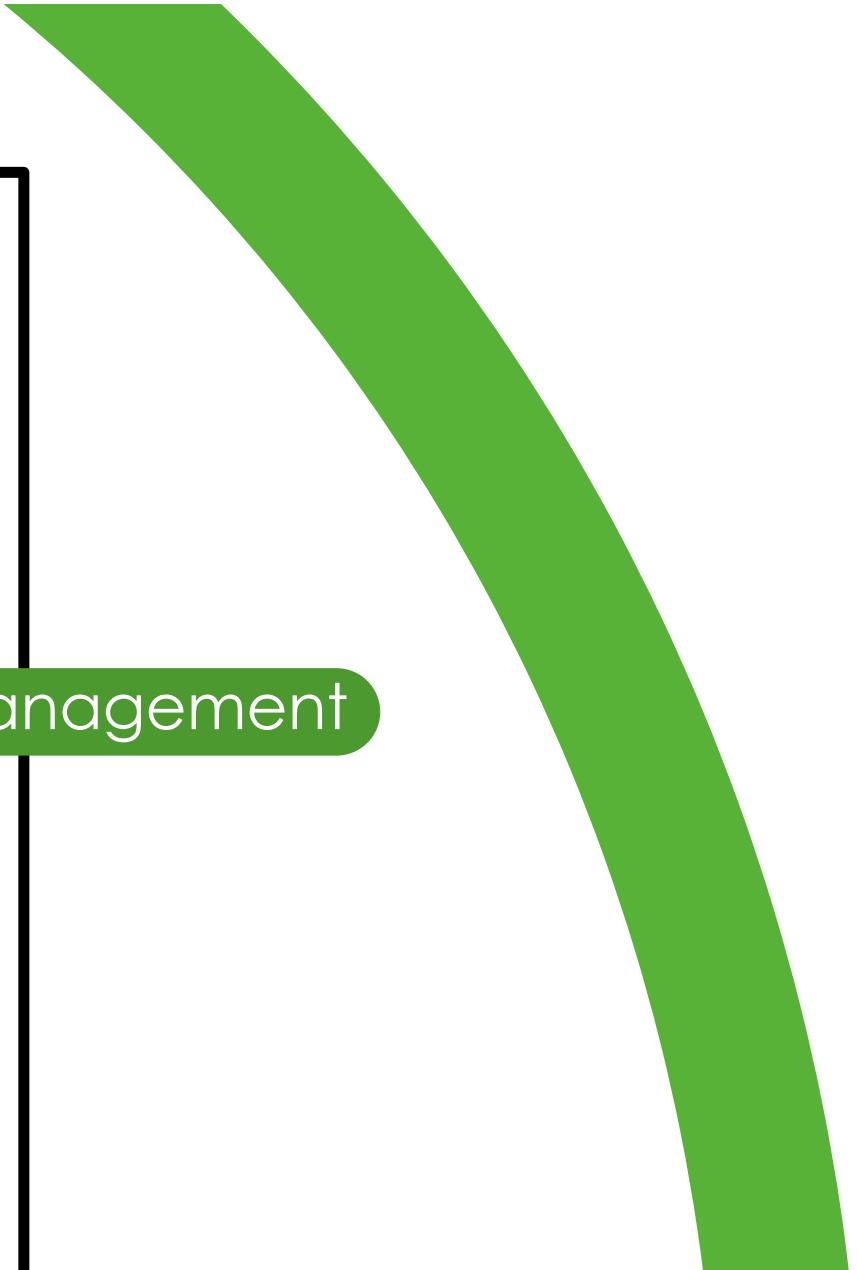
Kategorie

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



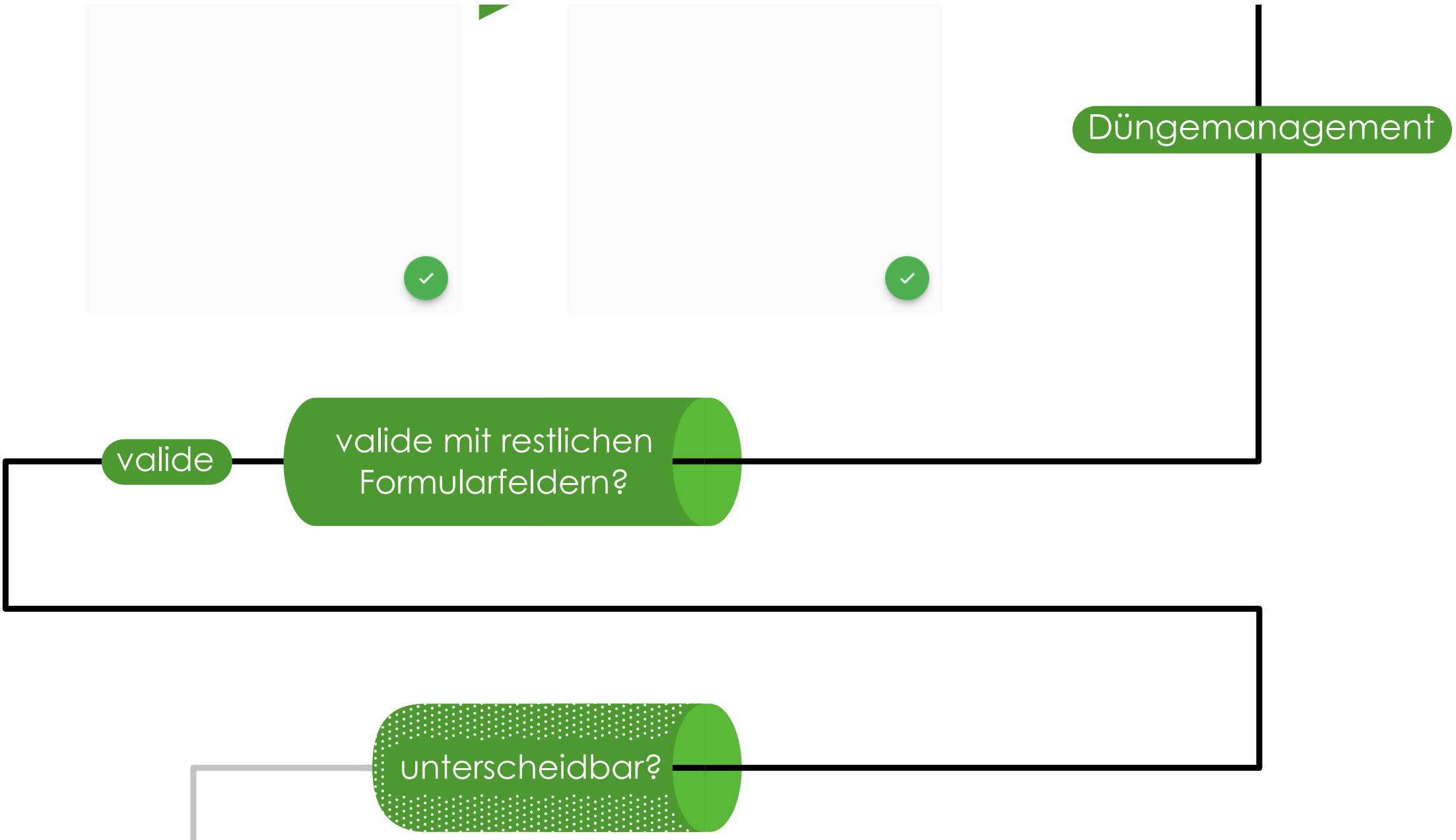
DÜ

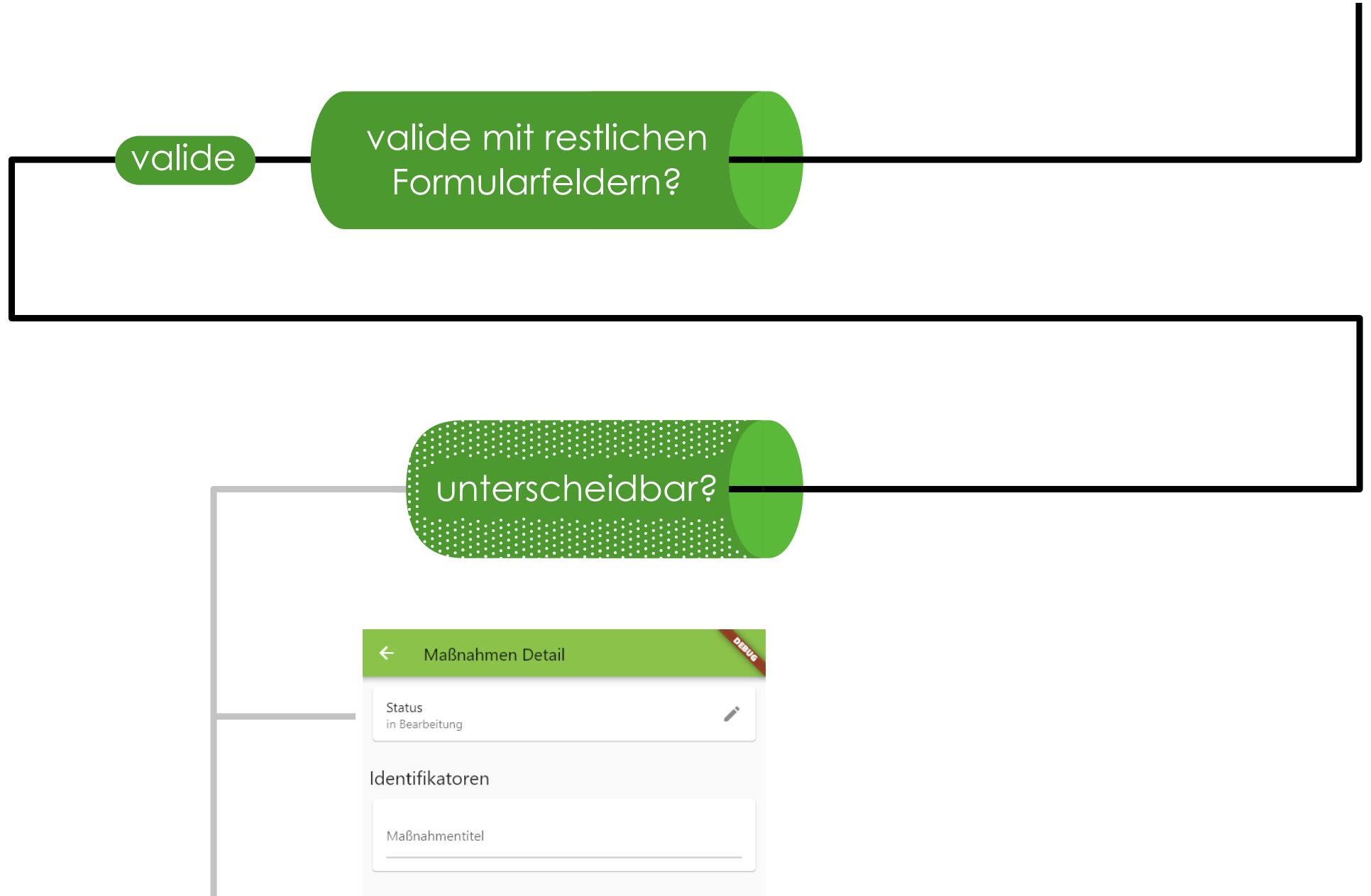
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

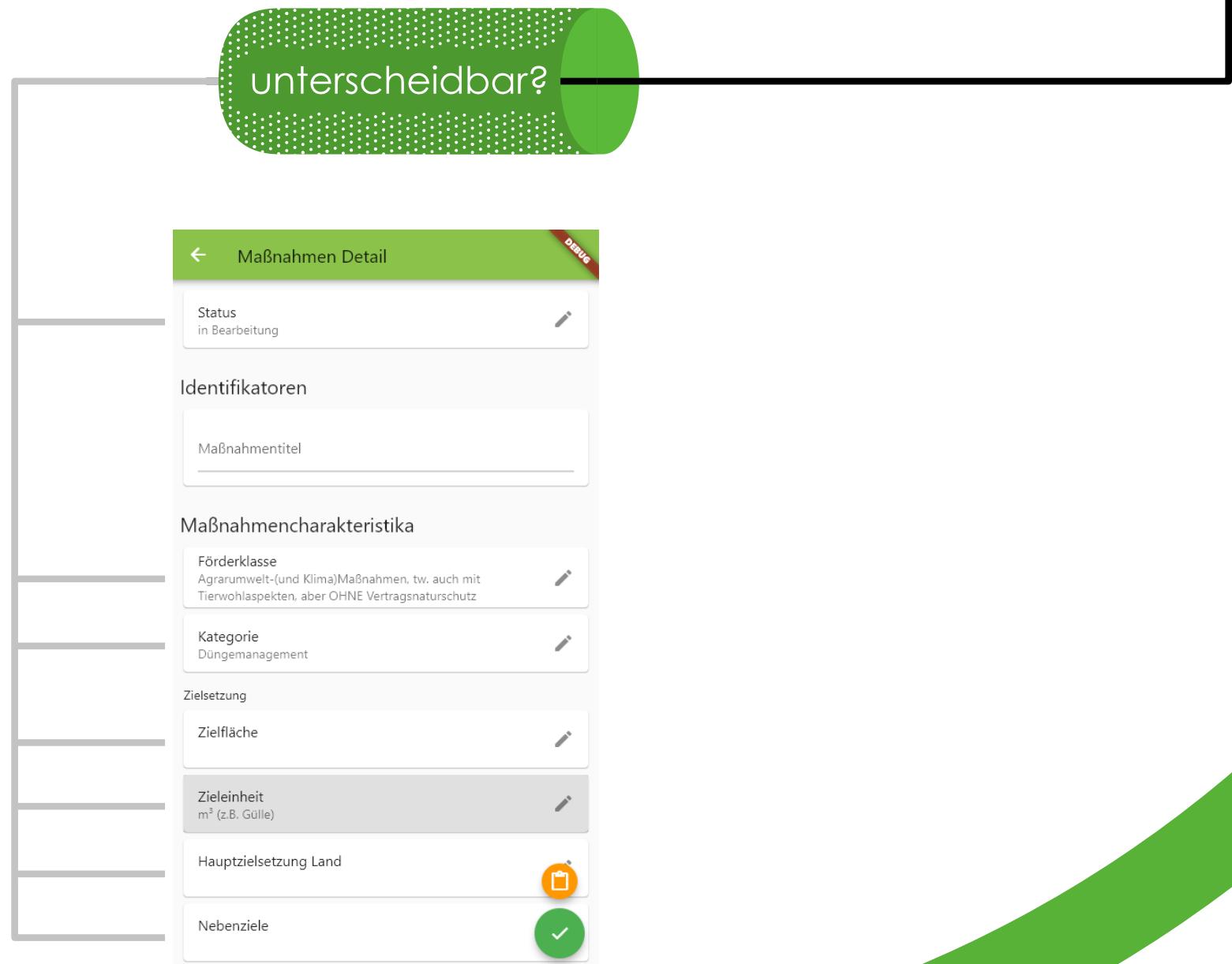


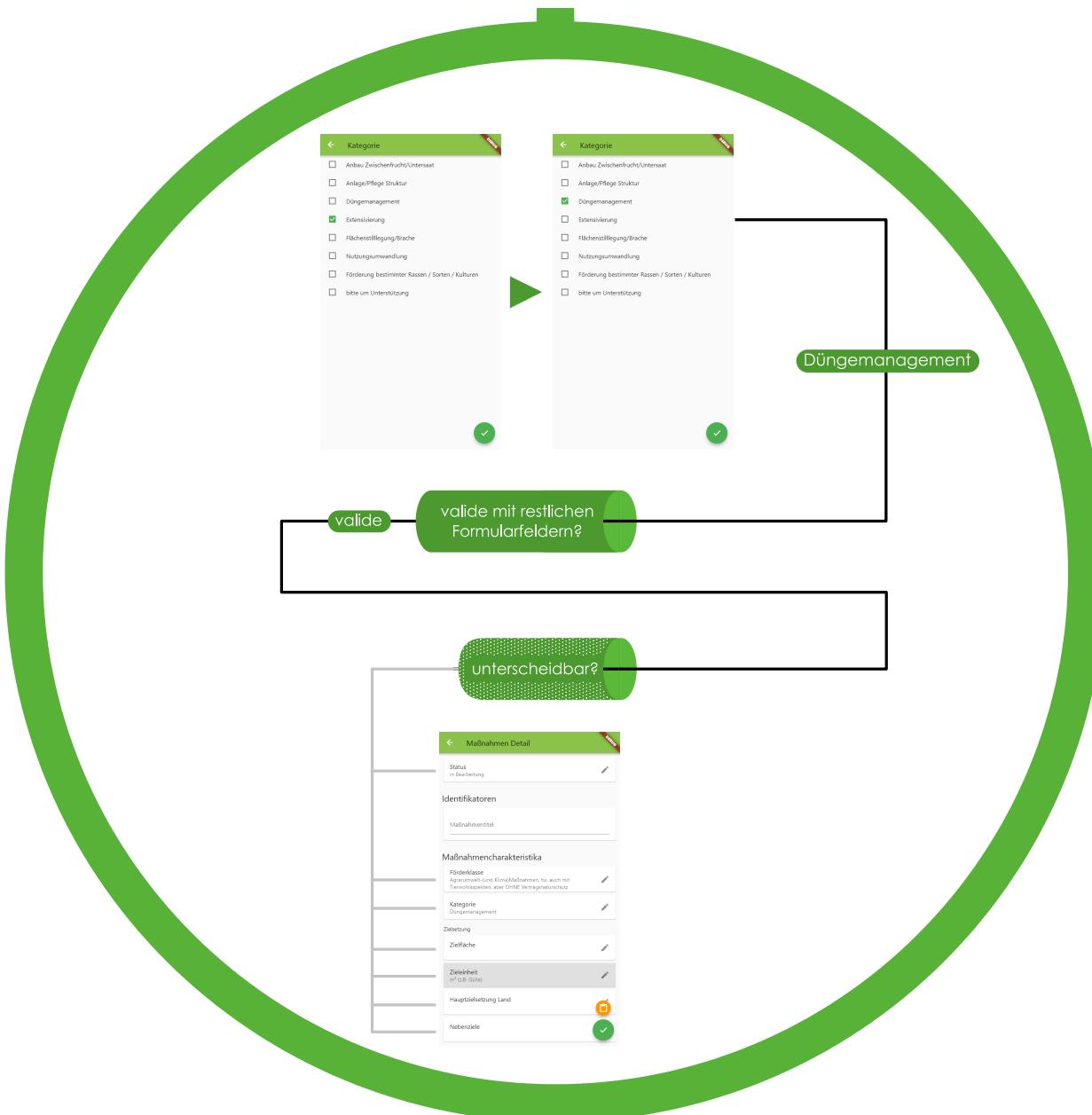
Düngemanagement













← Kategorie DEBUG

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



← Kategorie DEBUG

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



Brache



tlichen

- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung



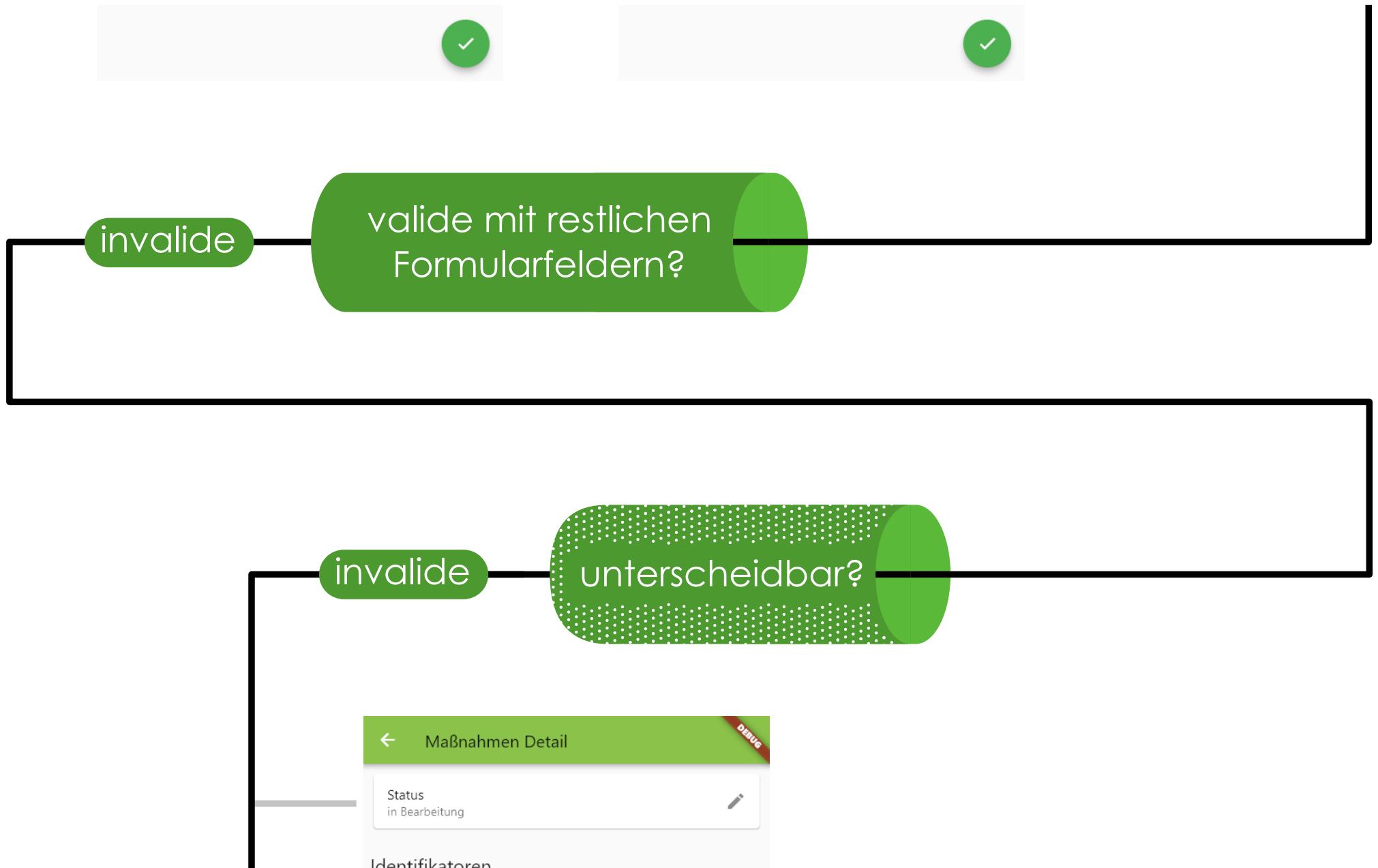
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

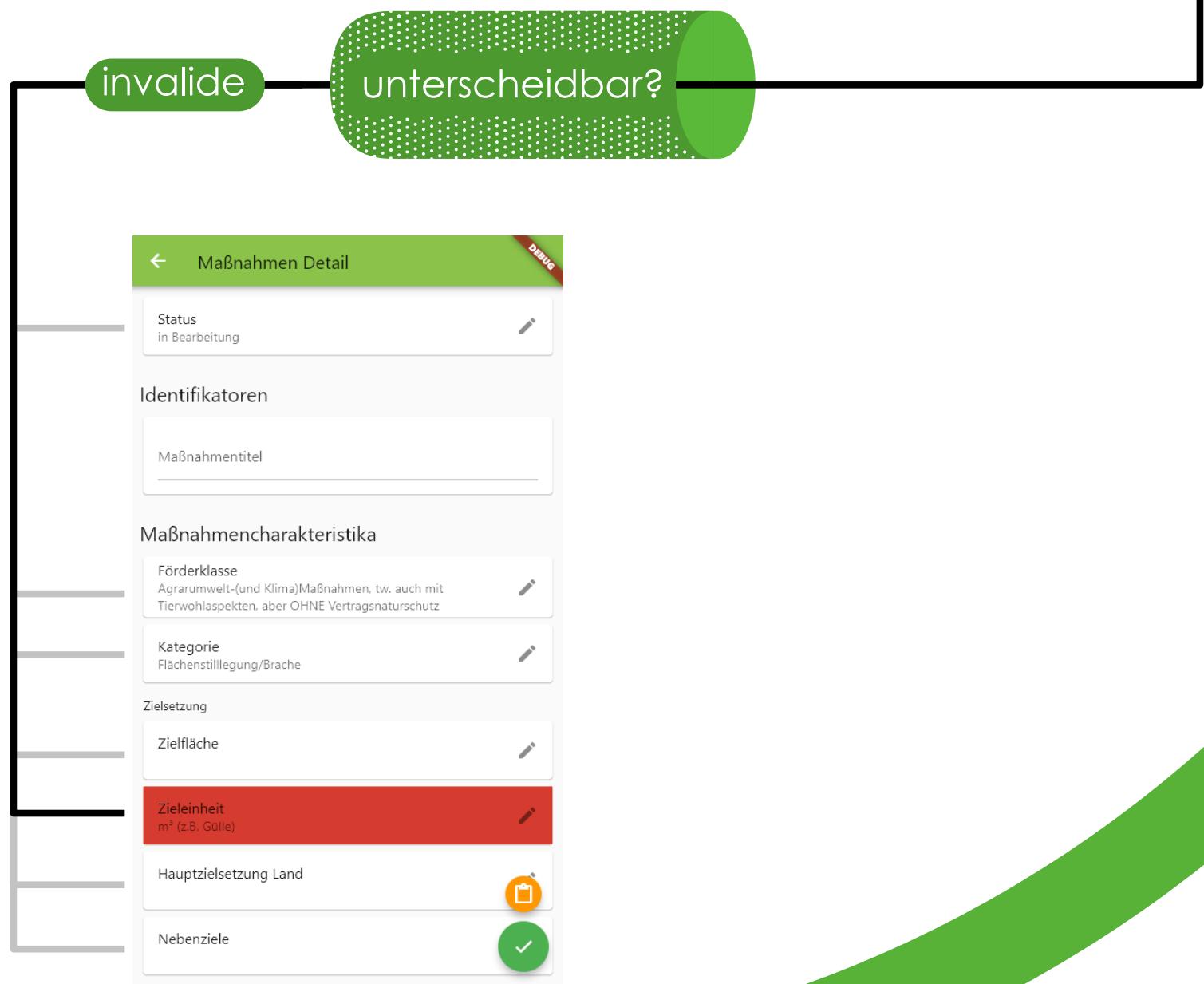
Brache



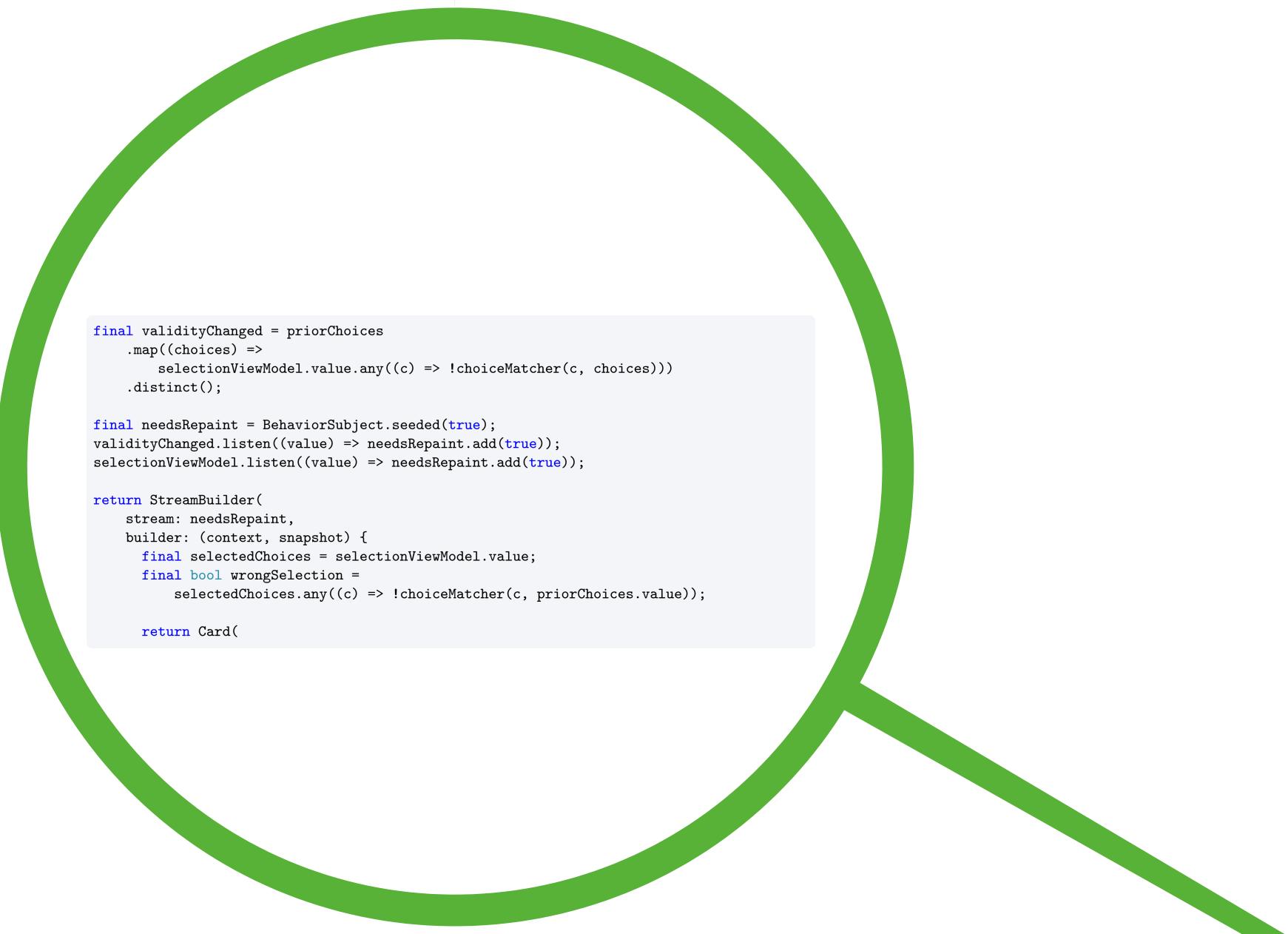
invalid

valide mit restlichen  
Formularfeldern?









```
final validityChanged = priorChoices
    .map((choices) =>
        selectionViewModel.value.any((c) => !choiceMatcher(c, choices)))
    .distinct();

final needsRepaint = BehaviorSubject.seeded(true);
validityChanged.listen((value) => needsRepaint.add(true));
selectionViewModel.listen((value) => needsRepaint.add(true));

return StreamBuilder(
    stream: needsRepaint,
    builder: (context, snapshot) {
        final selectedChoices = selectionViewModel.value;
        final bool wrongSelection =
            selectedChoices.any((c) => !choiceMatcher(c, priorChoices.value));

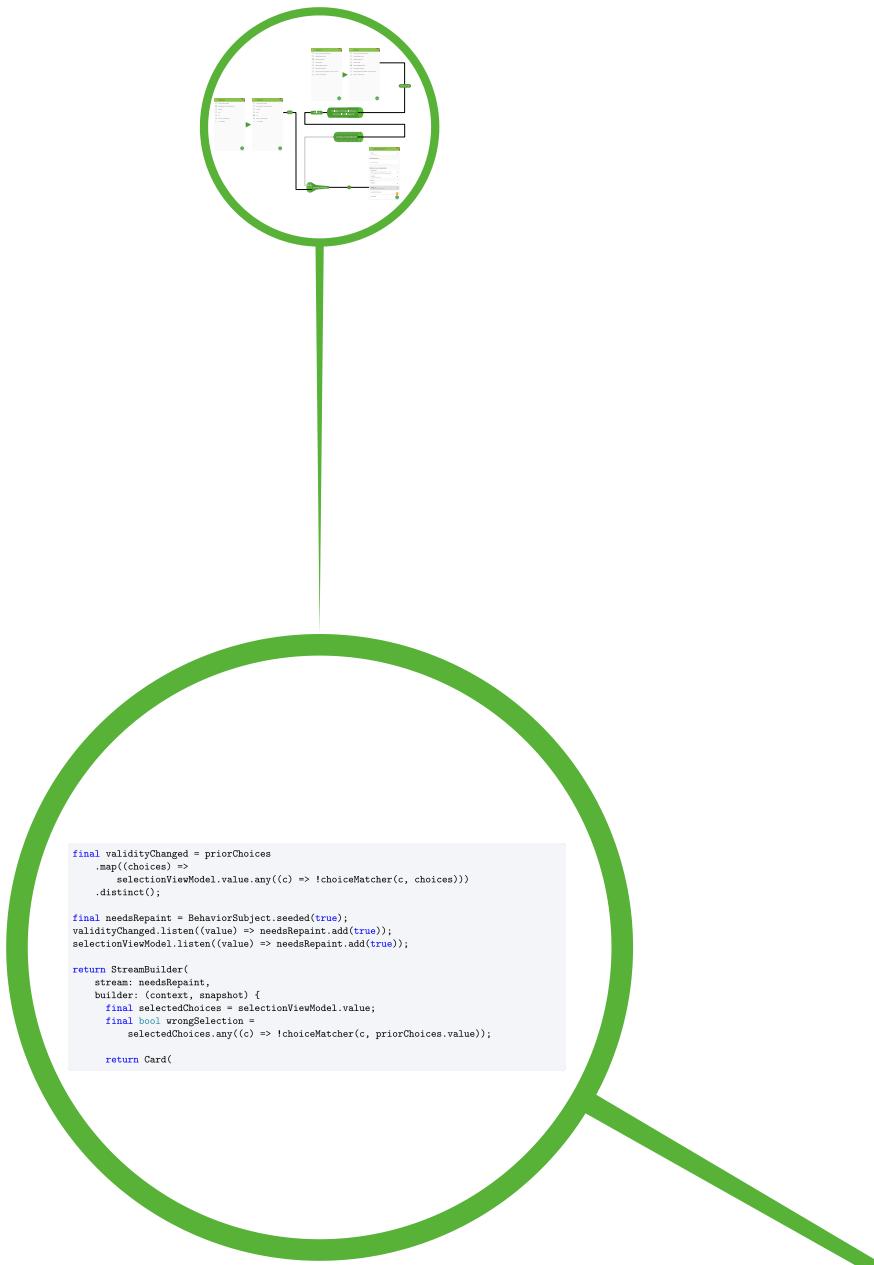
        return Card(
```

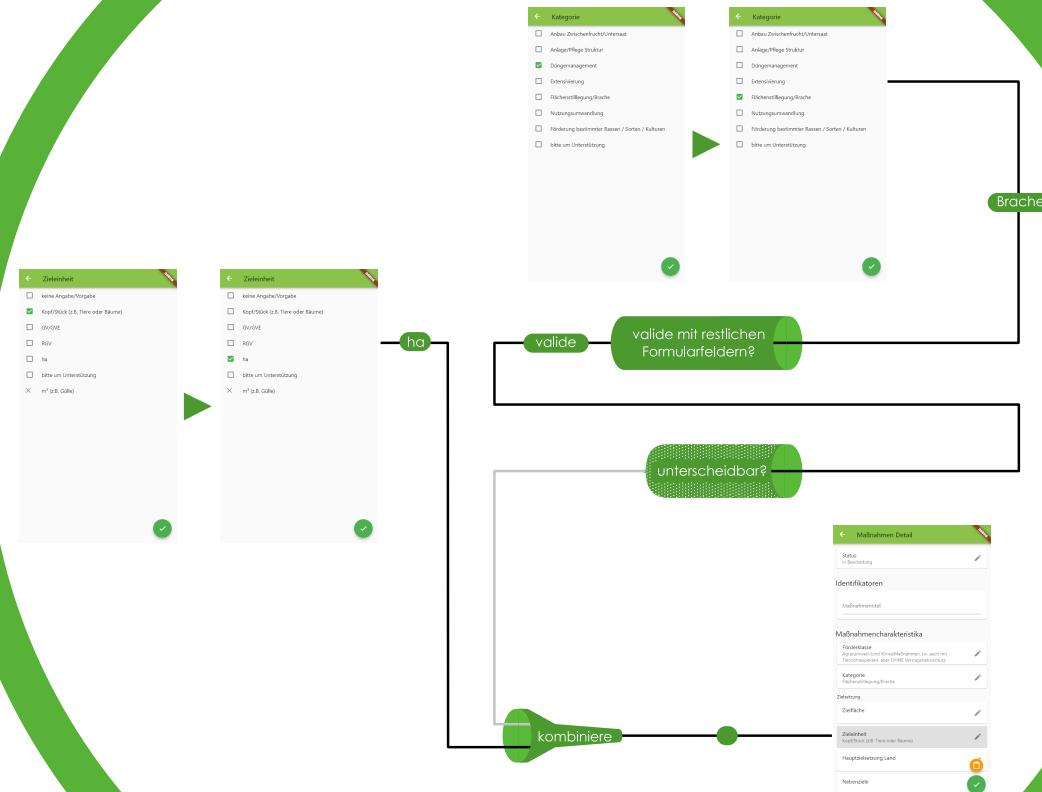
```
final validityChanged = priorChoices
    .map((choices) =>
        selectionViewModel.value.any((c) => !choiceMatcher(c, choices)))
    .distinct();

final needsRepaint = BehaviorSubject.seeded(true);
validityChanged.listen((value) => needsRepaint.add(true));
selectionViewModel.listen((value) => needsRepaint.add(true));

return StreamBuilder(
    stream: needsRepaint,
    builder: (context, snapshot) {
        final selectedChoices = selectionViewModel.value;
        final bool wrongSelection =
            selectedChoices.any((c) => !choiceMatcher(c, priorChoices.value));

        return Card(
```





← Zieleinheit

DEBÜS

- keine Angabe/Vorgabe
- Kopf/Stück (z.B. Tiere oder Bäume)
- GV/GVE
- RGV
- ha
- bitte um Unterstützung
- m<sup>3</sup> (z.B. Gülle)

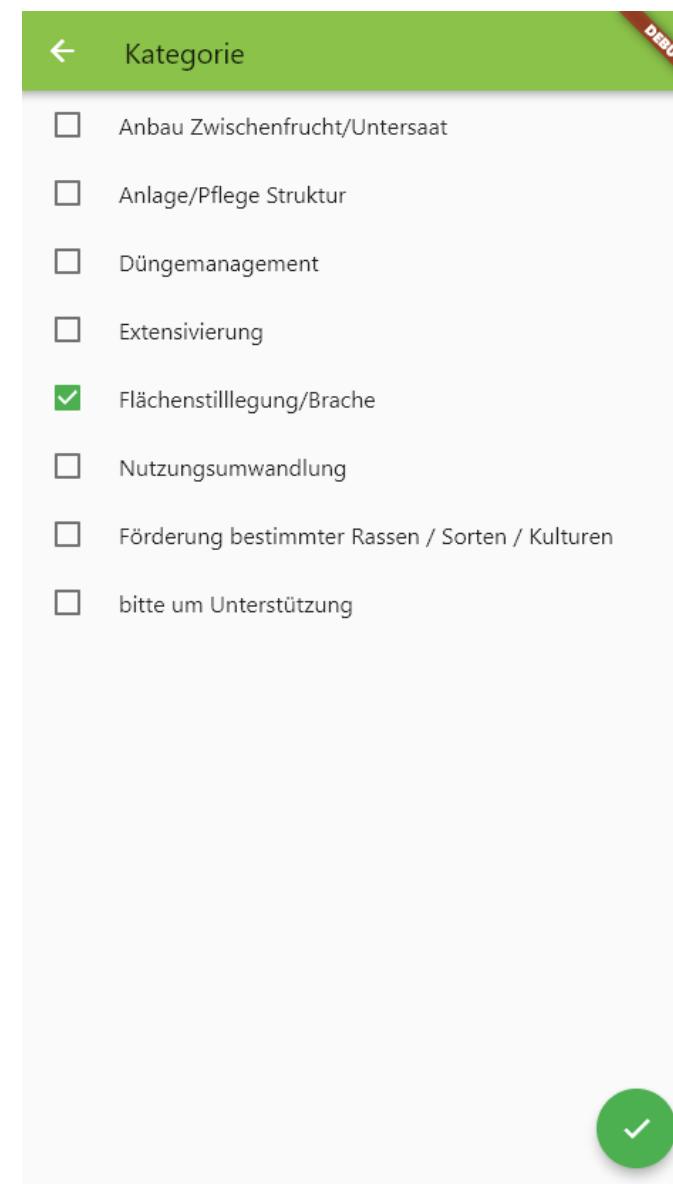
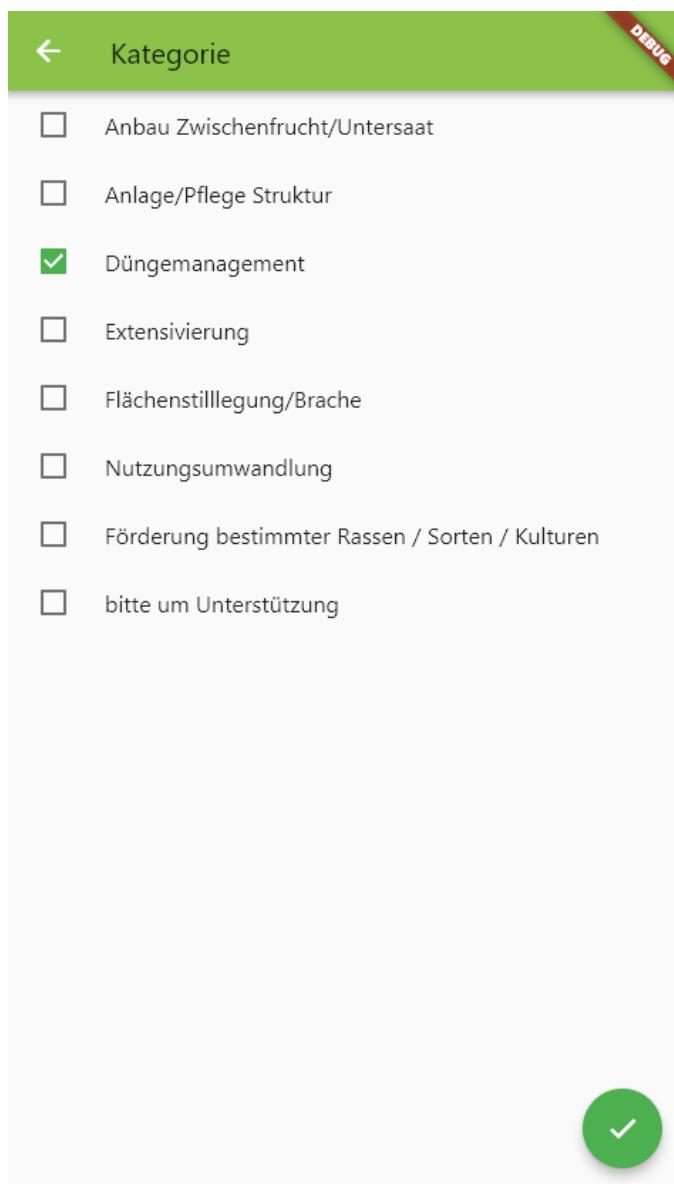


## Kategorie

DEBUE

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung





Kategorie

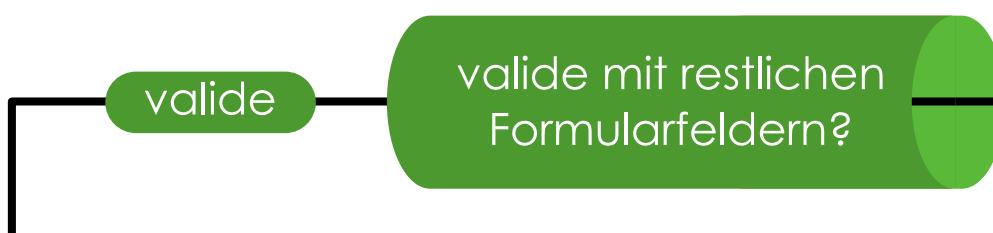
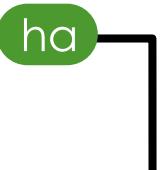
- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

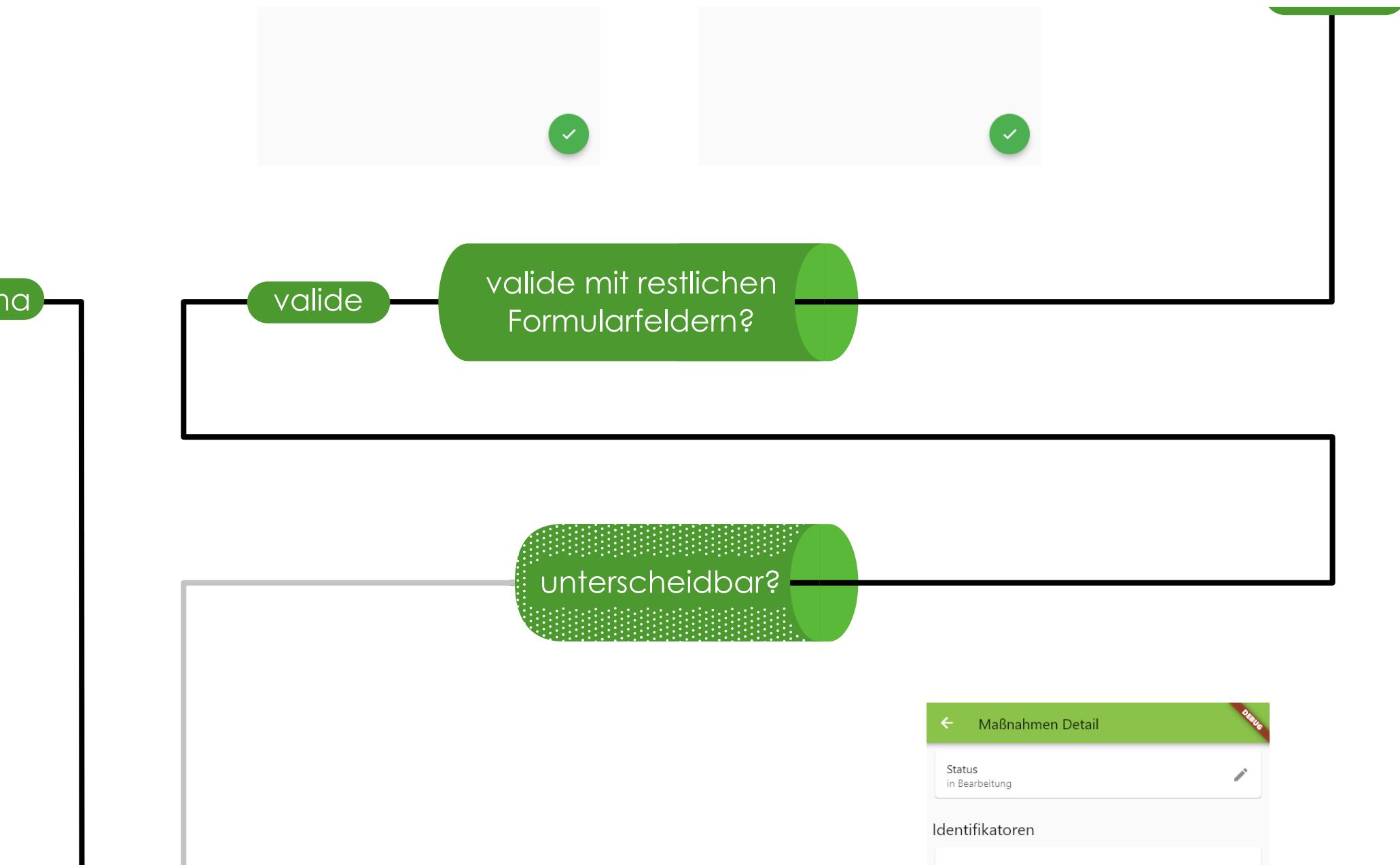


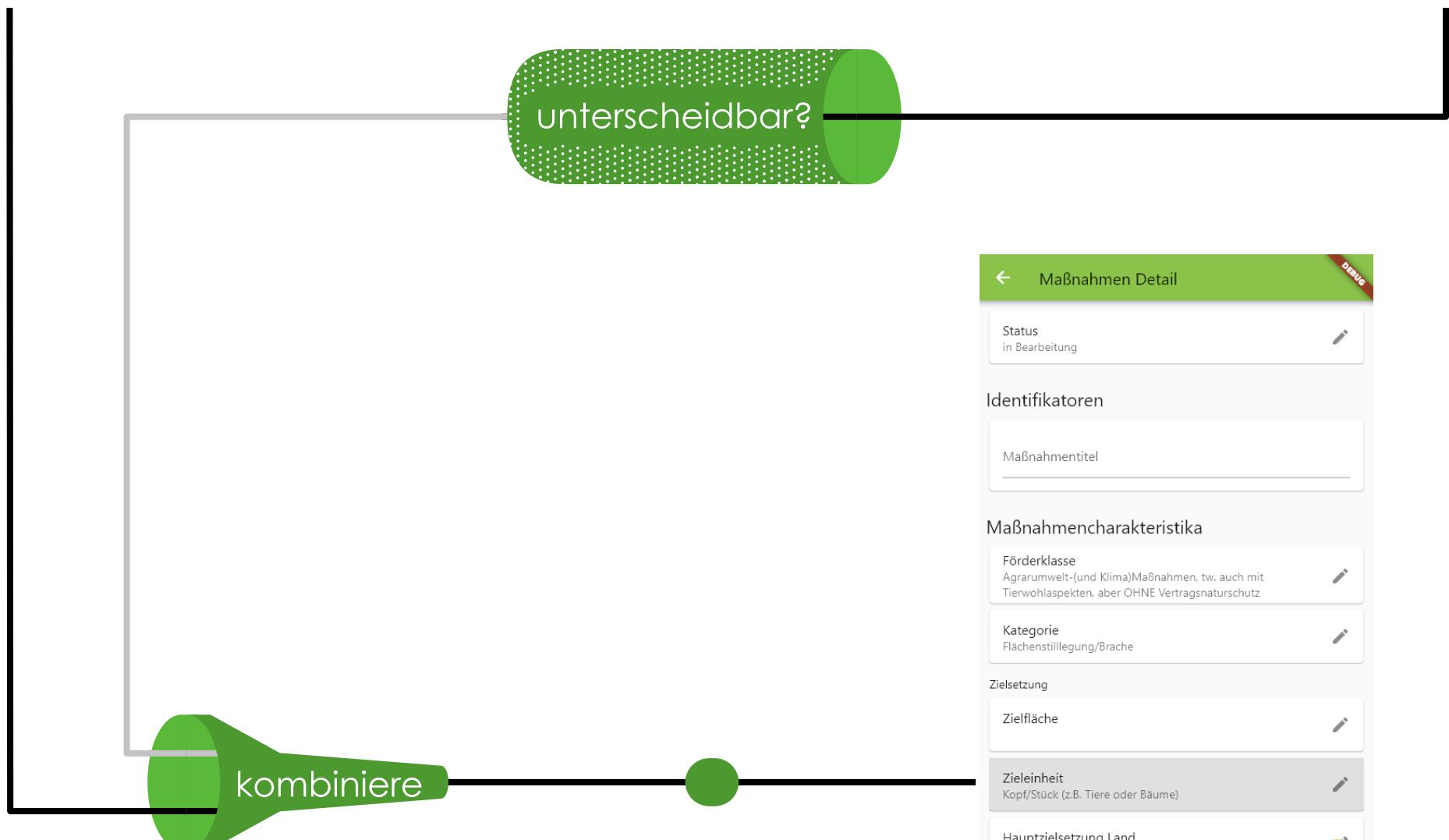
Kategorie

- Anbau Zwischenfrucht/Untersaat
- Anlage/Pflege Struktur
- Düngemanagement
- Extensivierung
- Flächenstilllegung/Brache
- Nutzungsumwandlung
- Förderung bestimmter Rassen / Sorten / Kulturen
- bitte um Unterstützung

Brache







← Zieleinheit DEBUG

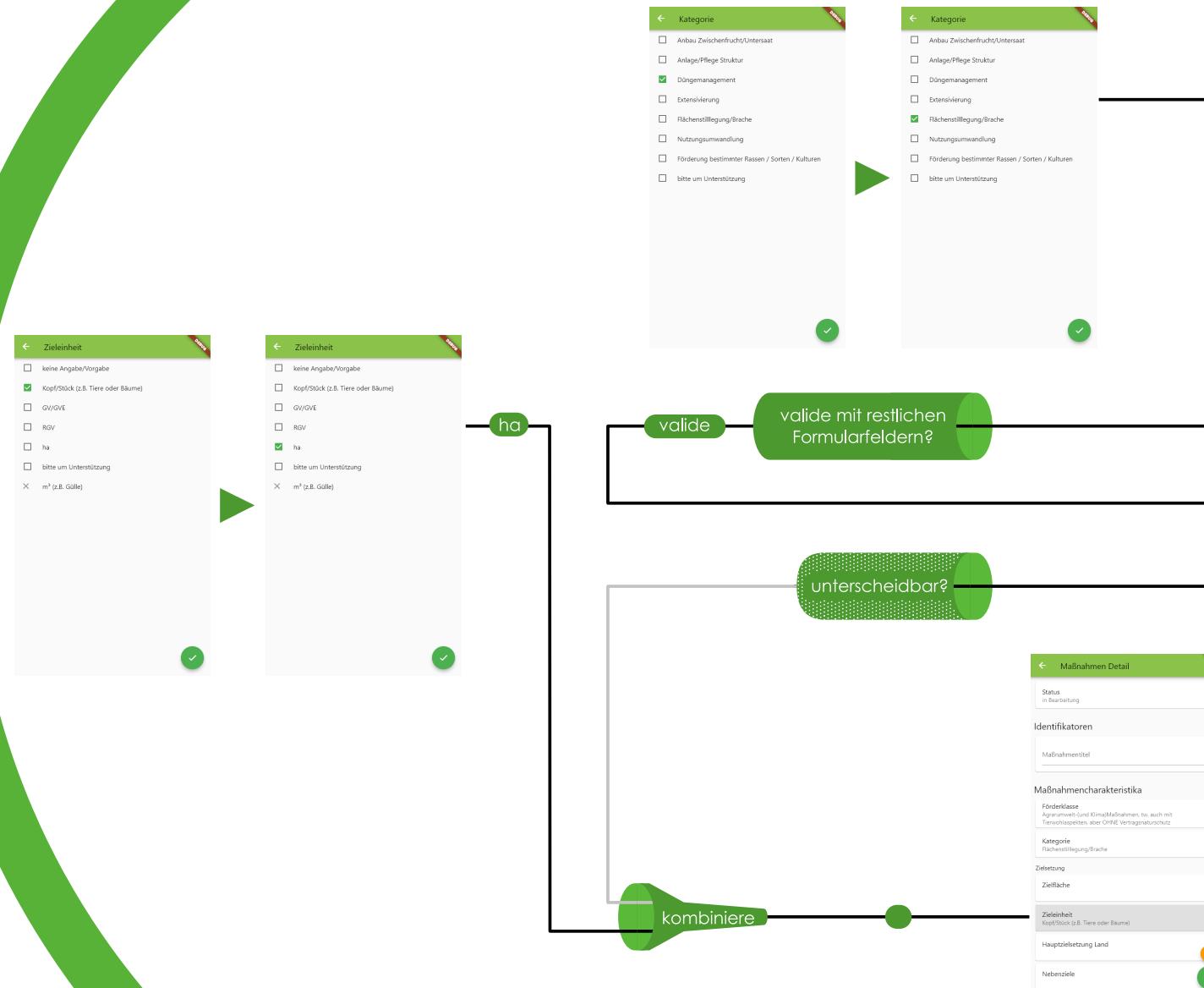
- keine Angabe/Vorgabe
- Kopf/Stück (z.B. Tiere oder Bäume)
- GV/GVE
- RGV
- ha
- bitte um Unterstützung
- m<sup>3</sup> (z.B. Gülle)



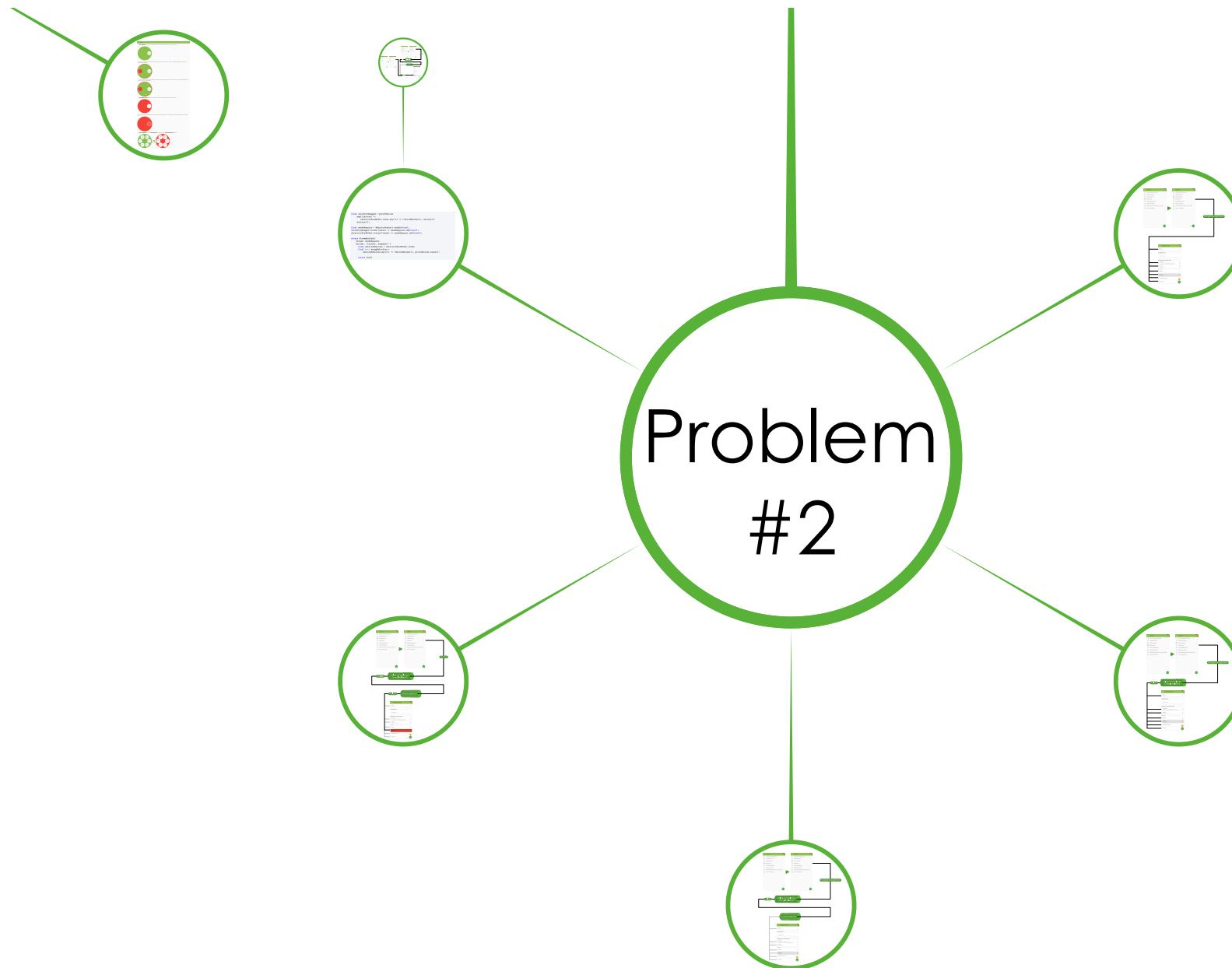
← Zieleinheit DEBUG

- keine Angabe/Vorgabe
- Kopf/Stück (z.B. Tiere oder Bäume)
- GV/GVE
- RGV
- ha
- bitte um Unterstützung
- m<sup>3</sup> (z.B. Gülle)

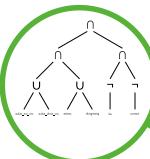




# Problem #2



# Problem #3



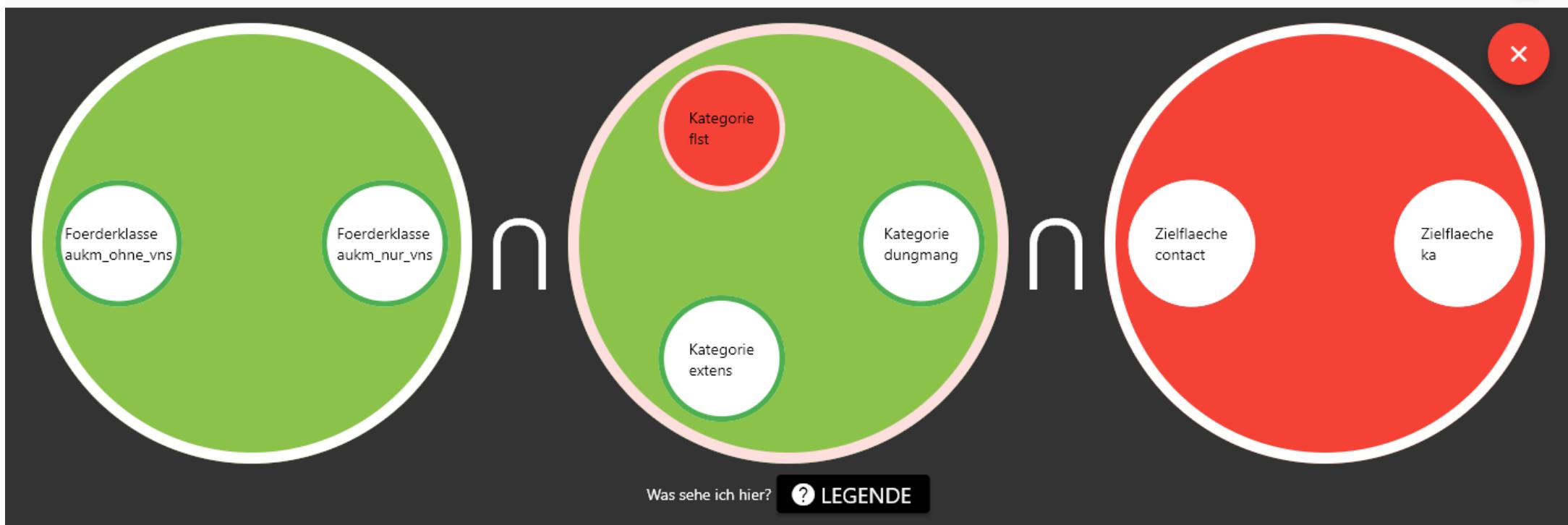
← Zieleinheit

keine Angabe/Vorgabe (i)

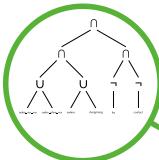
ha (i)

bitte um Unterstützung (i)

m<sup>3</sup> (z.B. Gülle) (✓)

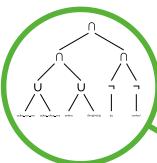


# Problem #3



```
class ZieleinheitChoice extends Choice {  
    static final ka = ZieleinheitChoice("ka", "keine Angabe/Vorgabe");  
    static final m3 = ZieleinheitChoice("m3", "m3 (z.B. Gölle)",  
        condition: (choices) =>  
            (choices.contains(FoerderklasseChoice.aukm_nur_vns) ||  
             choices.contains(FoerderklasseChoice.aukm_ohne_vns)) &&  
            (choices.contains(KategorieChoice.dungmang) ||  
             choices.contains(KategorieChoice.extens)) &&  
            (!choices.contains(ZielflaecheChoice.ka) &&  
             !choices.contains(ZielflaecheChoice.contact)));
```

# Problem #3



```
git clone https://github.com/.../BinarySearchTree.git
cd BinarySearchTree
javac Main.java
java Main
```

A screenshot of a terminal window showing the execution of Java code related to a binary search tree. The code includes imports for java.util.\* and java.io.\*.

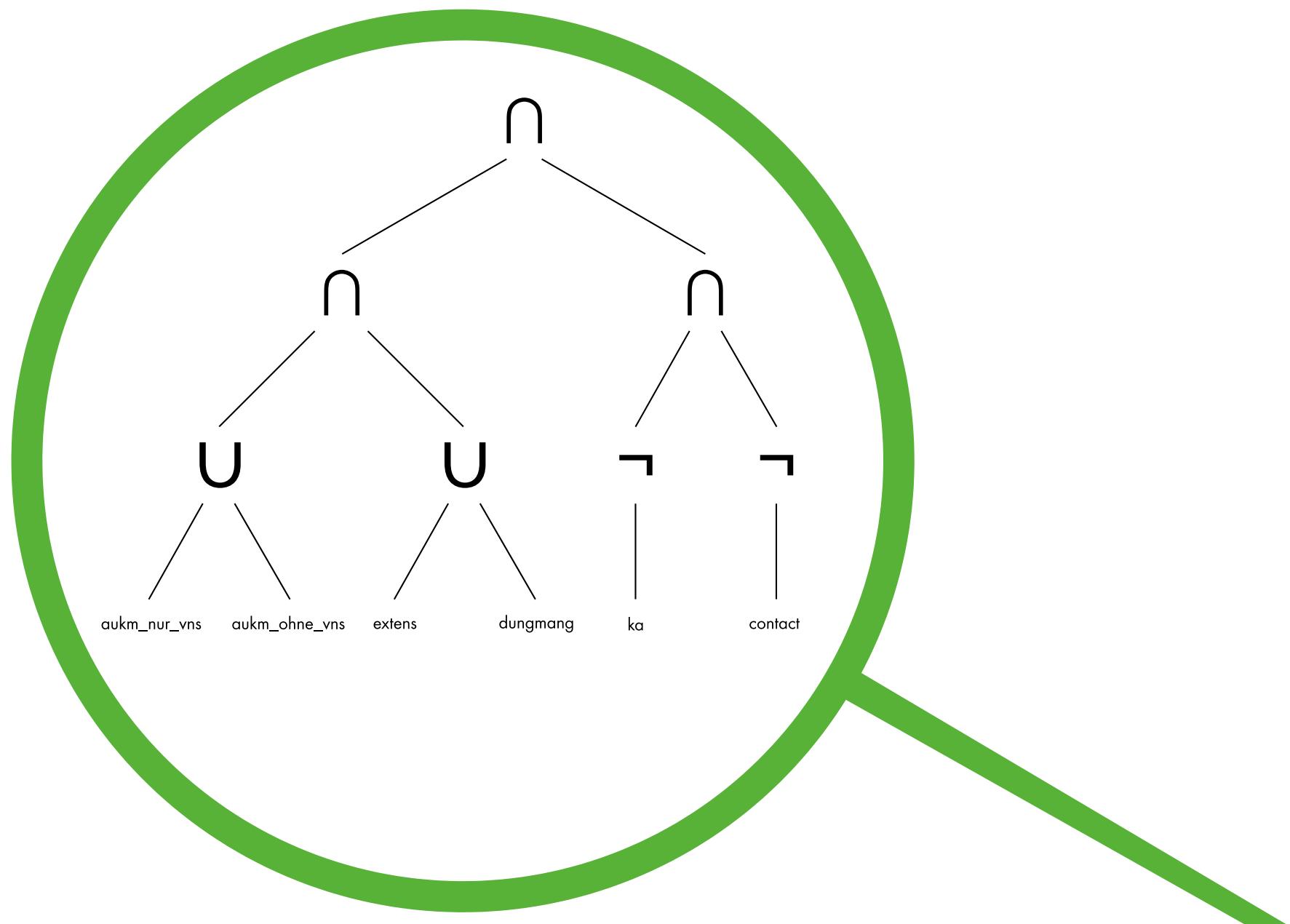


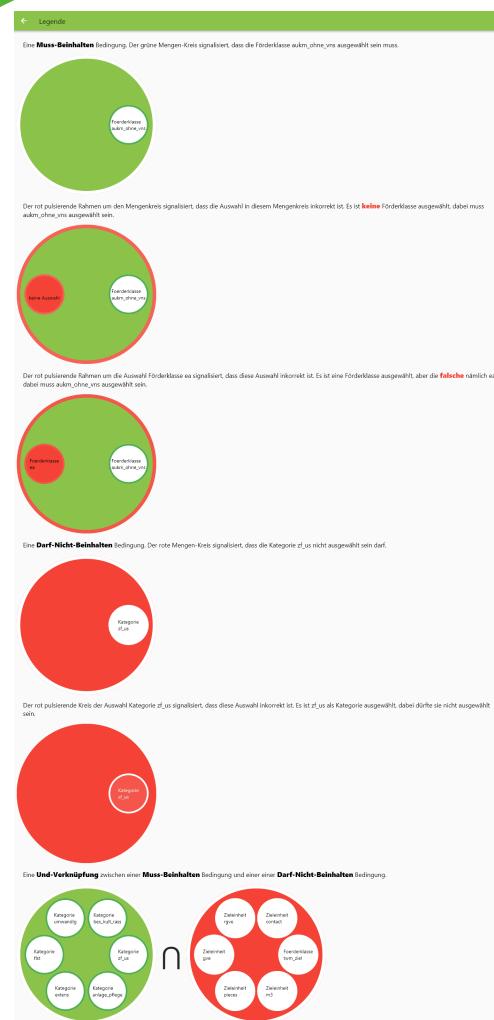
Quelle: <http://www.artswfl.com/wp-content/uploads/2012/10/006-43.jpg>





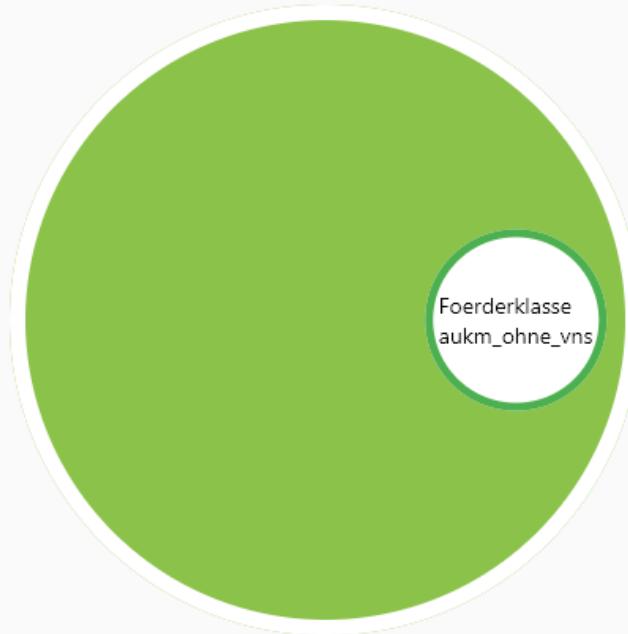
Quelle: <http://www.artswfl.com/wp-content/uploads/2012/10/006-43.jpg>





← Legende

Eine **Muss-Beinhalten** Bedingung. Der grüne Mengen-Kreis signalisiert, dass die Förderklasse aukm\_ohne\_vns ausgewählt sein muss.

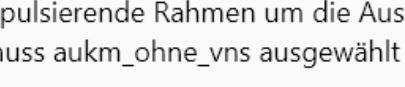
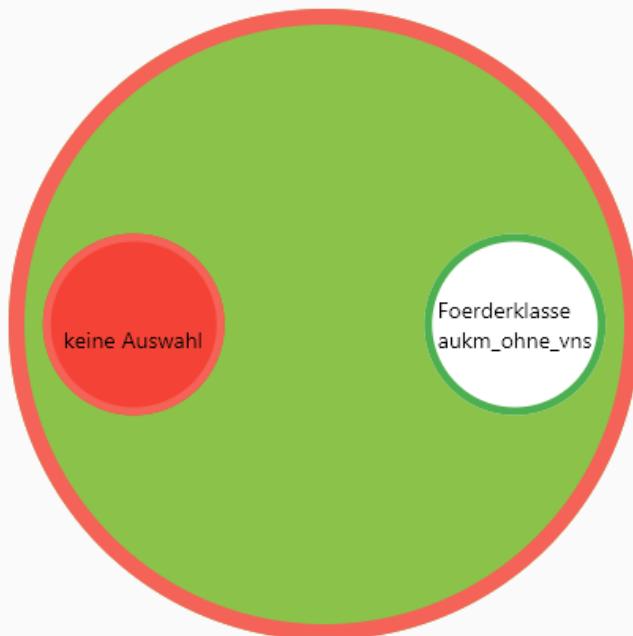


Der rot pulsierende Rahmen um den Mengenkreis signalisiert, dass die Auswahl in diesem Mengenkreis inkorrekt ist. Es ist **keine** Förderklasse ausgewählt, dabei muss aukm\_ohne\_vns ausgewählt sein.



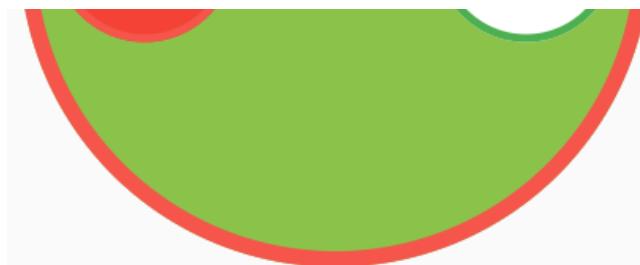


Der rot pulsierende Rahmen um den Mengenkreis signalisiert, dass die Auswahl in diesem Mengenkreis inkorrekt ist. Es ist **keine** Förderklasse ausgewählt, dabei muss aukm\_ohne\_vns ausgewählt sein.

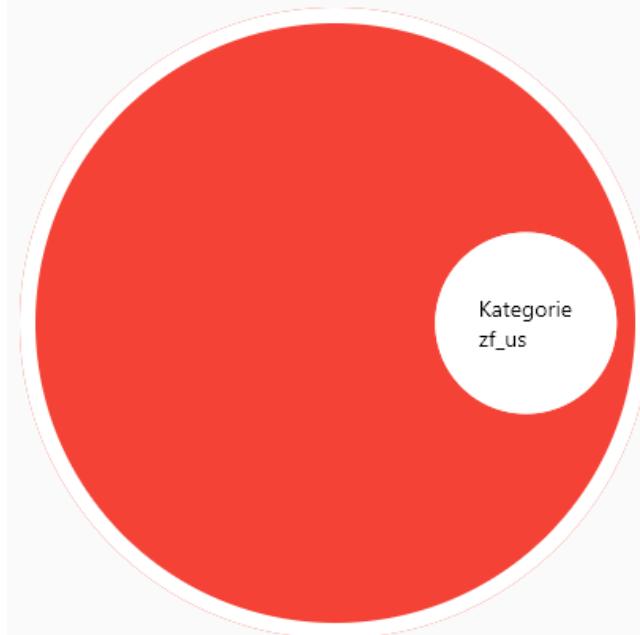


Der rot pulsierende Rahmen um die Auswahl Förderklasse ea signalisiert, dass diese Auswahl inkorrekt ist. Es ist eine Förderklasse ausgewählt, aber die **falsche** nämlich ea, dabei muss aukm\_ohne\_vns ausgewählt sein.





Eine **Darf-Nicht-Beinhalten** Bedingung. Der rote Mengen-Kreis signalisiert, dass die Kategorie zf\_us nicht ausgewählt sein darf.

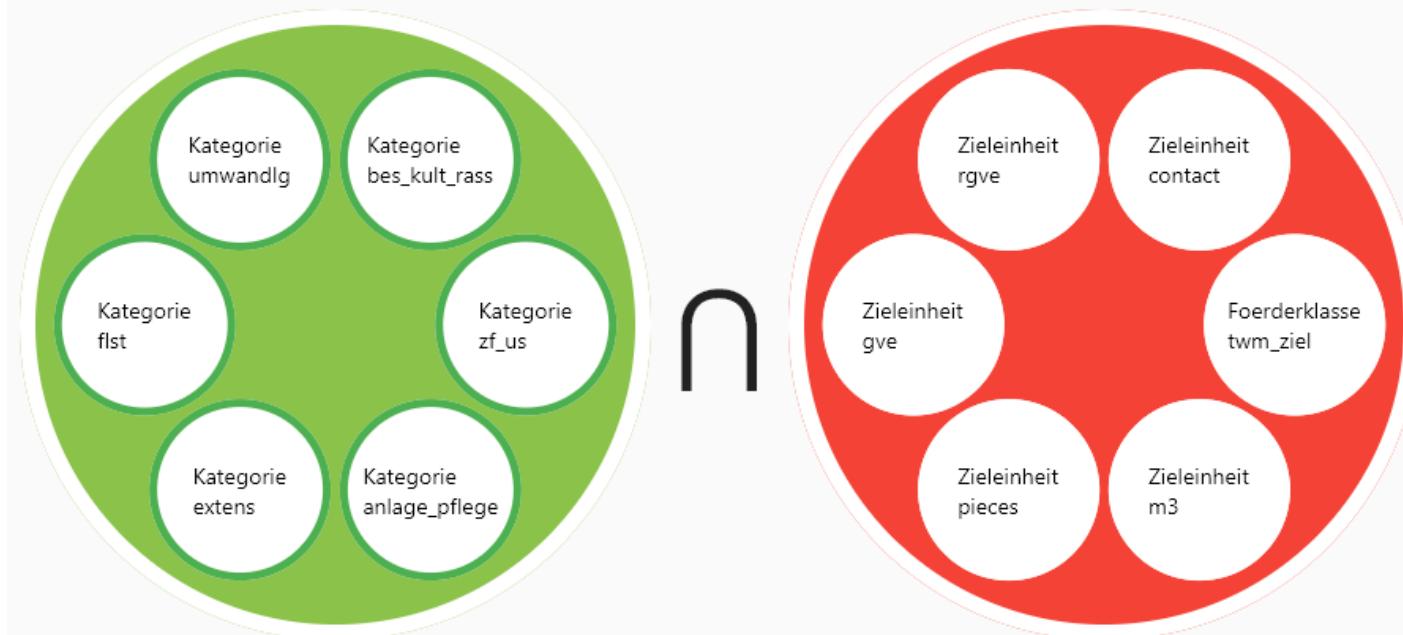


Der rot pulsierende Kreis der Auswahl Kategorie zf\_us signalisiert, dass diese Auswahl inkorrekt ist. Es ist zf\_us als Kategorie ausgewählt, dabei dürfte sie nicht ausgewählt sein.

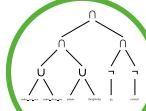


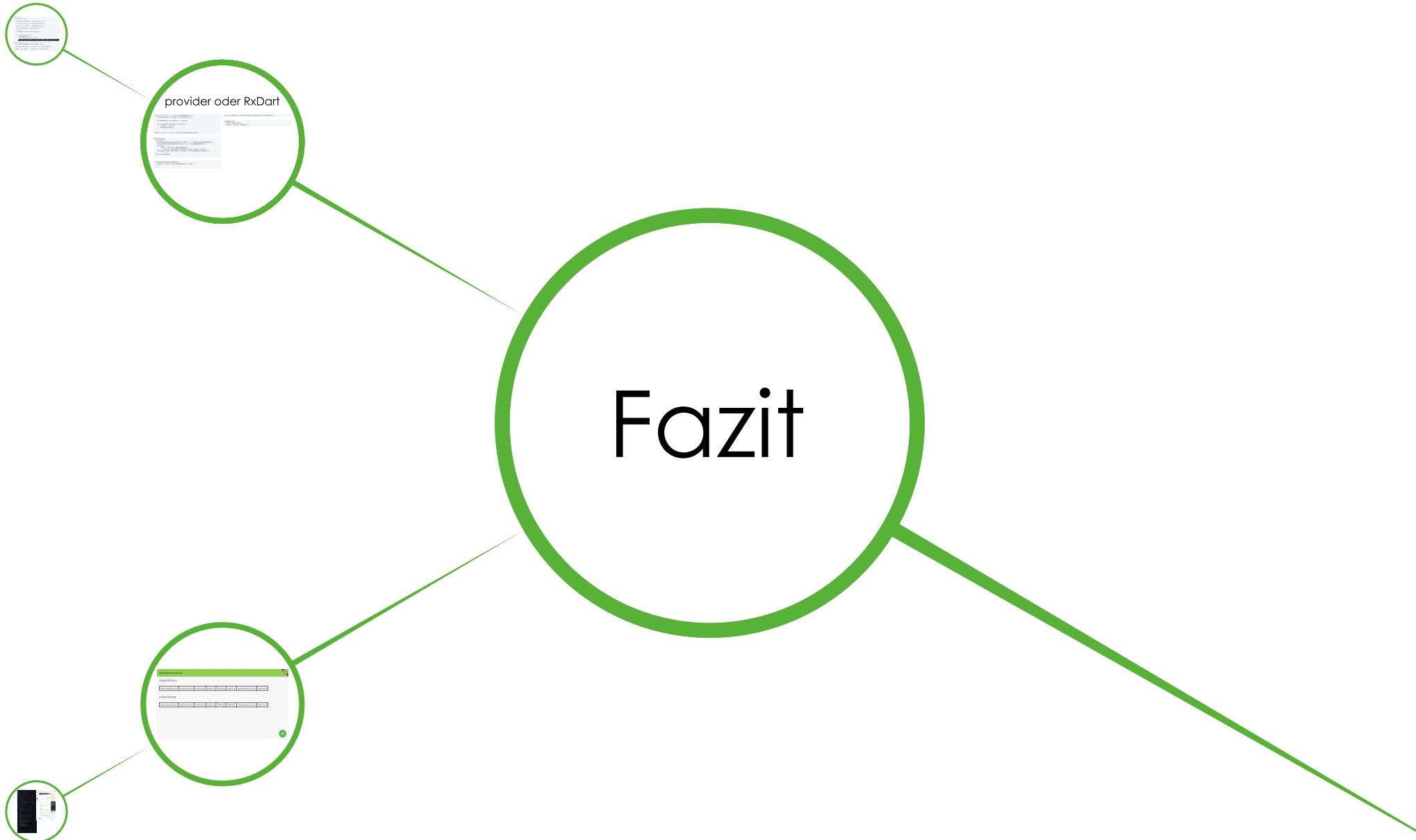
zf\_us

Eine **Und-Verknüpfung** zwischen einer **Muss-Beinhalten** Bedingung und einer einer **Darf-Nicht-Beinhalten** Bedingung.



# Problem #3





# Welche Technologie ?

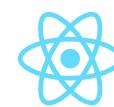


wenduna

## Test-Bibliotheken



Flutter



React Native



Flutter



Jest



React Native Testing Library



appium



Detox

# Fazit



## Maßnahmen Master

DEBUG

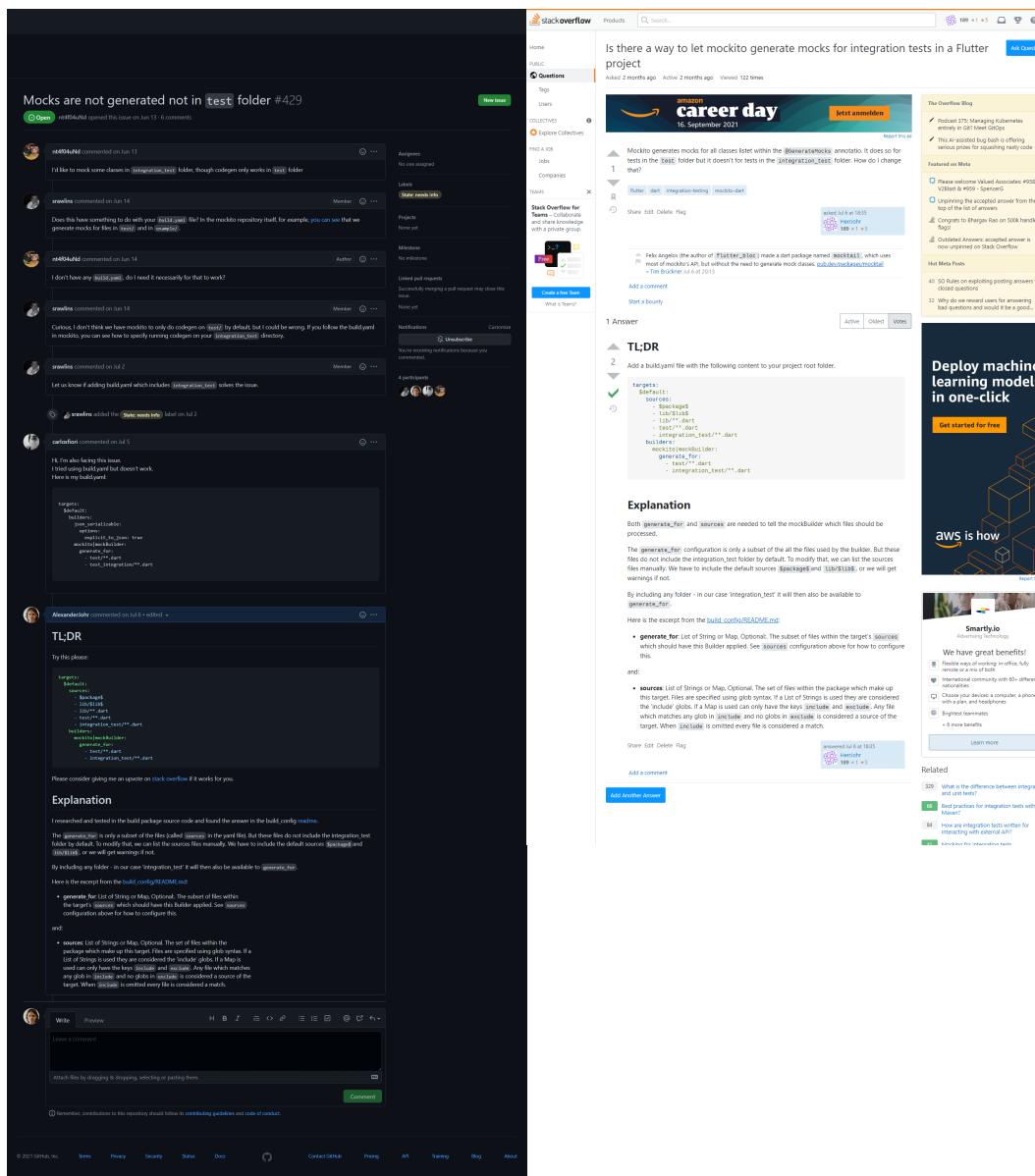
## Abgeschlossen

Zuletzt bearbeitet am	Maßnahmentitel	Förderklasse	Kategorie	Zielfläche	Zieleinheit	Hauptzielsetzung Land	Nebenziele
-----------------------	----------------	--------------	-----------	------------	-------------	-----------------------	------------

## In Bearbeitung

Zuletzt bearbeitet am	Maßnahmentitel	Förderklasse	Kategorie	Zielfläche	Zieleinheit	Hauptzielsetzung Land	Nebenziele
-----------------------	----------------	--------------	-----------	------------	-------------	-----------------------	------------





# Mocks are not generated not in test folder #429

Open

nt4f04uNd opened this issue on Jun 13 · 6 comments



nt4f04uNd commented on Jun 13

...

I'd like to mock some classes in `integration_test` folder, though codegen only works in `test` folder



srawlins commented on Jun 14

Member

...

Does this have something to do with your `build.yaml` file? In the Mockito repository itself, for example, you can see that we generate mocks for files in `test/` and in `example/`.



nt4f04uNd commented on Jun 14

Author

...

I don't have any `build.yaml`, do I need it necessarily for that to work?



AlexanderJohr commented on Jul 6 • edited

...

## TL;DR

Try this please:

```
targets:  
$default:  
sources:  
- $package$  
- lib/$lib$  
- lib/**.dart  
- test/**.dart  
- integration_test/**.dart  
builders:  
mockito|mockBuilder:  
generate_for:  
- test/**.dart  
- integration_test/**.dart
```

Please consider giving me an upvote on [stack overflow](#) if it works for you.

## Explanation

# Explanation

I researched and tested in the build package source code and found the answer in the `build_config` `readme`.

The `generate_for` is only a subset of the files (called `sources` in the yaml file). But these files do not include the `integration_test` folder by default. To modify that, we can list the sources files manually. We have to include the default sources `$package$` and `lib/$lib$`, or we will get warnings if not.

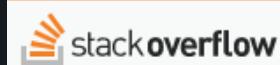
By including any folder - in our case '`integration_test`' it will then also be available to `generate_for`.

Here is the excerpt from the `build_config/README.md`:

- **generate\_for:** List of String or Map, Optional:. The subset of files within the target's `sources` which should have this Builder applied. See `sources` configuration above for how to configure this.

and:

- **sources:** List of Strings or Map, Optional. The set of files within the package which make up this target. Files are specified using glob syntax. If a List of Strings is used they are considered the 'include' globs. If a Map is used can only have the keys `include` and `exclude`. Any file which matches any glob in `include` and no globs in `exclude` is considered a source of the target. When `include` is omitted every file is considered a match.



Products



Search...

189 • 1 • 5



Home

PUBLIC

Questions

Tags

Users

COLLECTIVES

Explore Collectives

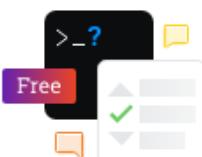
FIND A JOB

Jobs

Companies

TEAMS

**Stack Overflow for Teams** – Collaborate and share knowledge with a private group.

[Create a free Team](#)[What is Teams?](#)

# Is there a way to let Mockito generate mocks for integration tests in a Flutter project

[Ask Question](#)[Report this ad](#)

Mockito generates mocks for all classes listed within the `@GenerateMocks` annotation. It does so for tests in the `test` folder but it doesn't for tests in the `integration_test` folder. How do I change that?

[flutter](#) [dart](#) [integration-testing](#) [mockito-dart](#)
[Share](#) [Edit](#) [Delete](#) [Flag](#)

asked Jul 6 at 18:35

HerrJohr

189 • 1 • 5

Felix Angelov (the author of `flutter_bloc`) made a dart package named `mocktail`, which uses most of Mockito's API, but without the need to generate mock classes. [pub.dev/packages/mocktail](#)  
– Tim Brückner Jul 6 at 20:13

[Add a comment](#)[Start a bounty](#)

## The Overflow Blog

Podcast 375: Managing Kubernetes entirely in Git? Meet GitOps

This AI-assisted bug bash is offering serious prizes for squashing nasty code

## Featured on Meta

Please welcome Valued Associates: #958 - V2Blast & #959 - SpencerG

Unpinning the accepted answer from the top of the list of answers

Congrats to Bhargav Rao on 500k handled flags!

Outdated Answers: accepted answer is now unpinned on Stack Overflow

## Hot Meta Posts

40 SO Rules on exploiting posting answers to closed questions

32 Why do we reward users for answering

## TL;DR

2

Add a build.yaml file with the following content to your project root folder.



```
targets:  
  $default:  
    sources:  
      - $package$  
      - lib/$lib$  
      - lib/**.dart  
      - test/**.dart  
      - integration_test/**.dart  
    builders:  
      mockito|mockBuilder:  
        generate_for:  
          - test/**.dart  
          - integration_test/**.dart
```

## Explanation

Both `generate_for` and `sources` are needed to tell the `mockBuilder` which files should be processed.

The `generate_for` configuration is only a subset of the all the files used by the builder. But these files do not include the `integration_test` folder by default. To modify that, we can list the `sources` files manually. We have to include the default sources `$package$` and `lib/$lib$`, or we will get warnings if not.

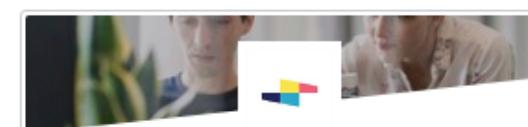
By including any folder - in our case '`integration_test`' it will then also be available to `generate_for`.

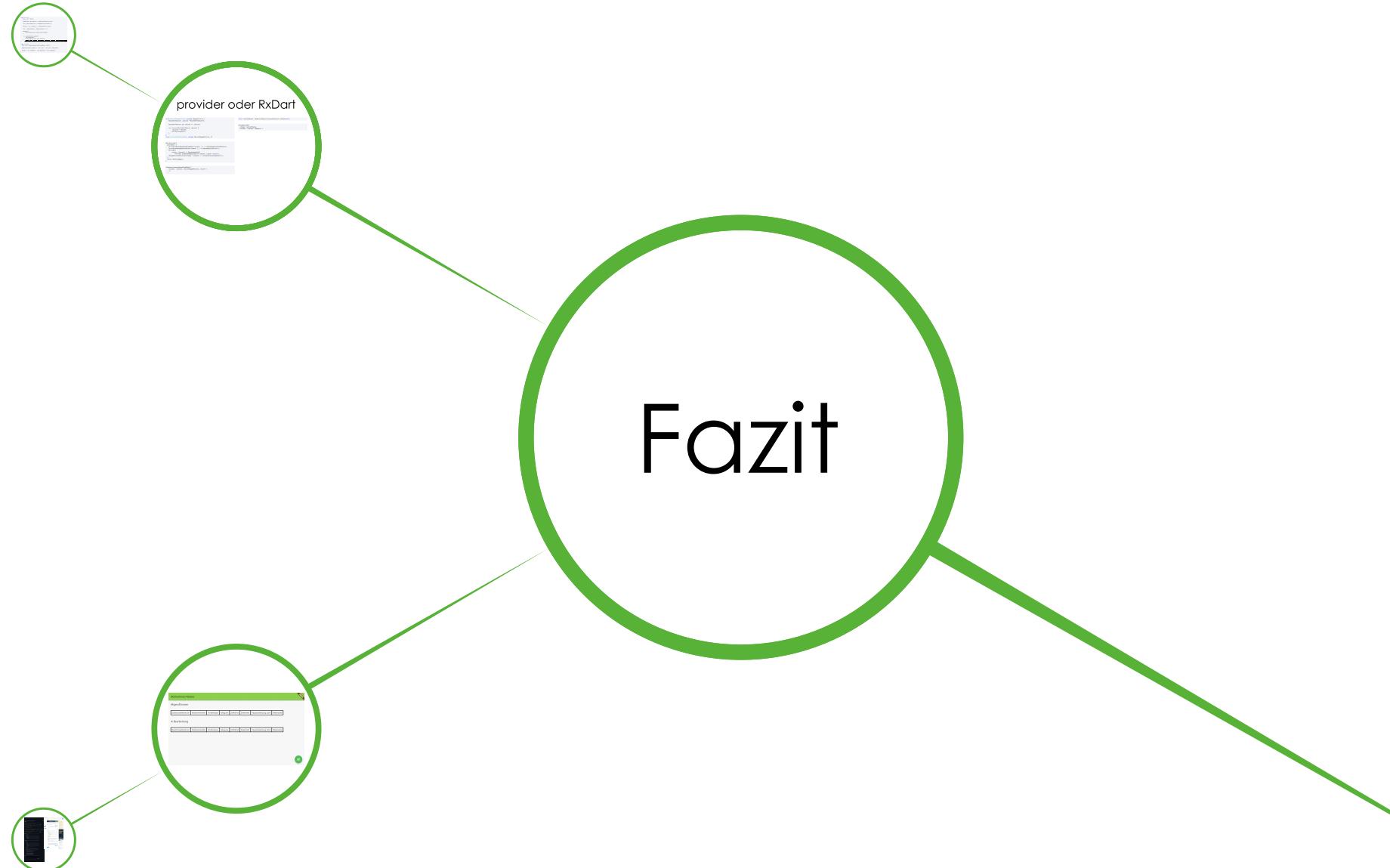
[Here is the excerpt from the `build\_config/README.md`.](#)

# Deploy machine learning models in one-click

[Get started for free](#)

aws is how

[Report this slide](#)

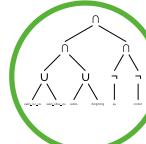


# Problem # 1

- Um im Ausland der Zielseite die Option **nr. auszuüben** zu können muss:
  - für die Produktionsentstehung
    - Agrarwelt und Klima/Möglichkeiten, vor Verbrauchsmärkten oder
    - Agrarwelt und Klima/Möglichkeiten, zw. auch mit Tiefwegen  
GÄNE Betriebsweisen
  - und für die Logistik entweder
    - Distributionsmanagement oder
    - Extrabeförderung
  - und für Zielseite darf **weder**
    - keine Angabe noch
    - bitte um Unterstützung gewählt sein.

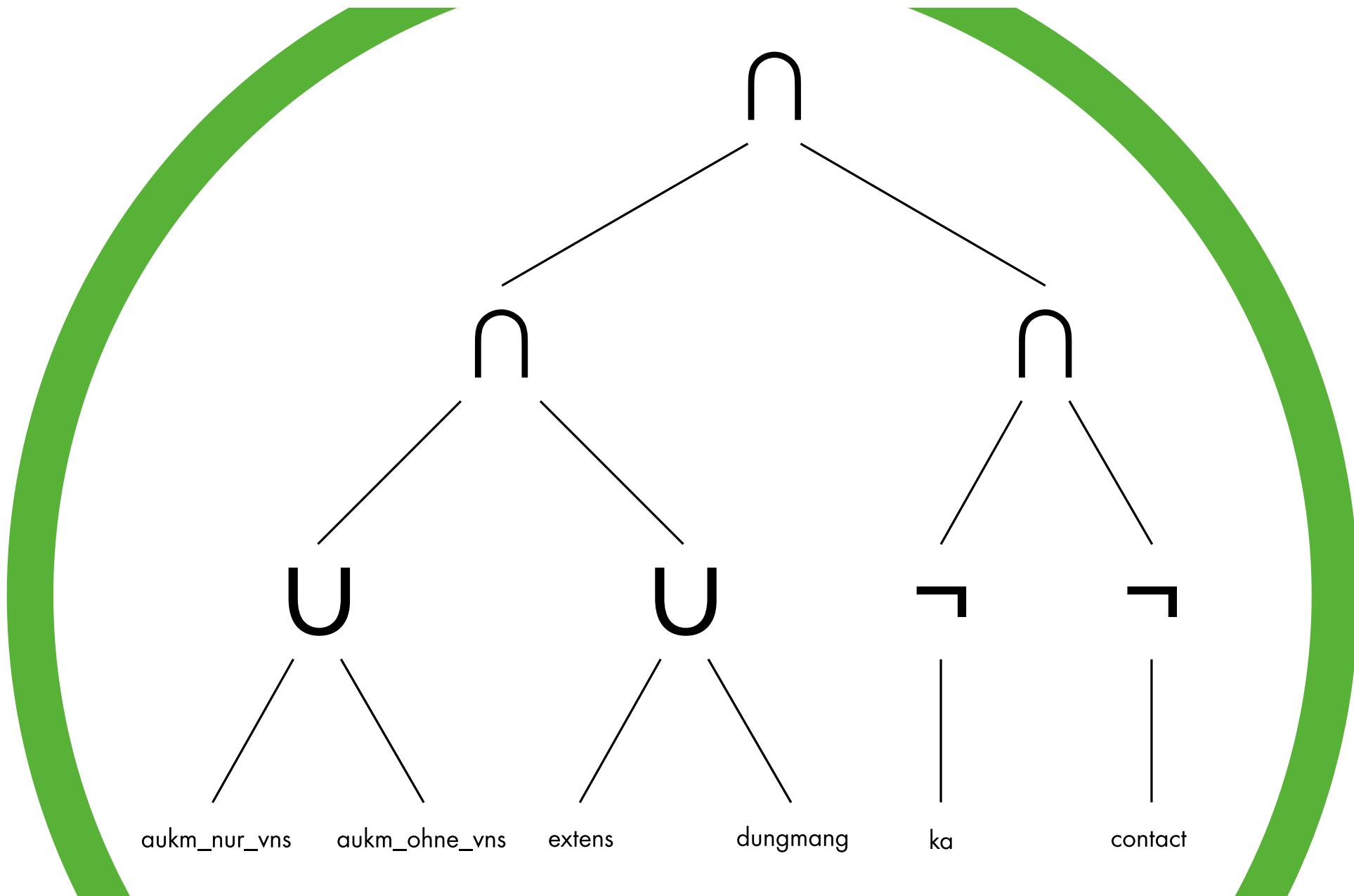
```
class ZieleinheitChoice extends Choice {  
    static final ka = ZieleinheitChoice("ka", "keine Angabe/Vorgabe");  
    static final m3 = ZieleinheitChoice("m3", "m3 (z.B. Gölle)",  
        condition: (choices) =>  
            (choices.contains(FoerderklasseChoice.aukm_nur_vns) ||  
             choices.contains(FoerderklasseChoice.aukm_ohne_vns)) &&  
            (choices.contains(KategorieChoice.dungmang) ||  
             choices.contains(KategorieChoice.extens)) &&  
            (!choices.contains(ZielflaecheChoice.ka) &&  
             !choices.contains(ZielflaecheChoice.contact)));
```

# Problem #3



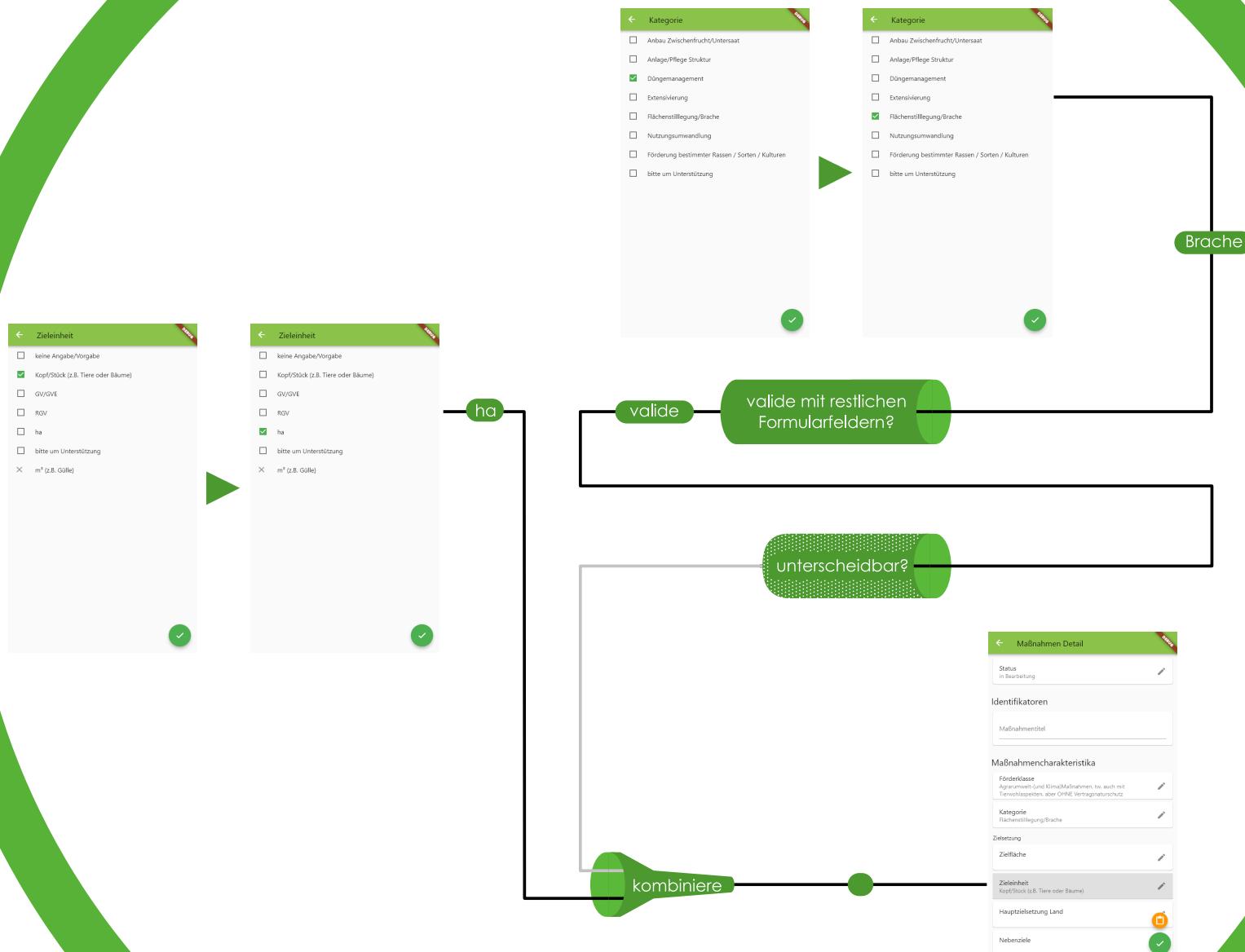
DATA STRUCTURE

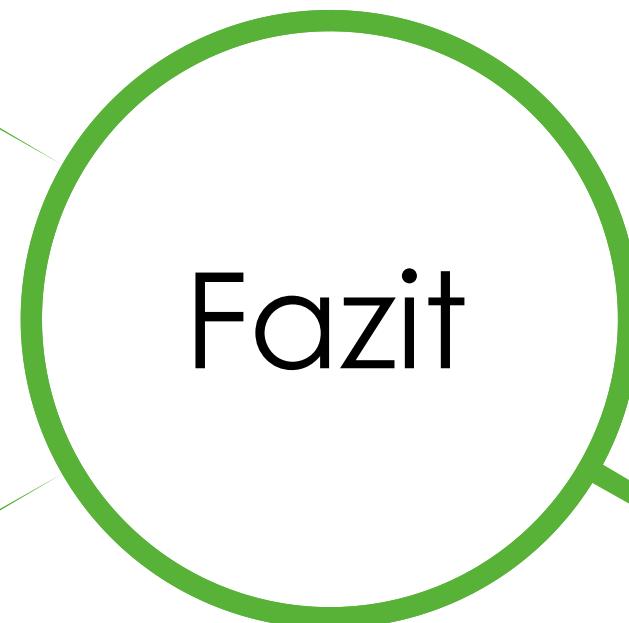




# Problem #2







# provider oder RxDart

```
class ChoiceChangeNotifier extends ChangeNotifier {
  BuiltSet<Choice> _choices = BuiltSet<Choice>();

  BuiltSet<Choice> get choices => _choices;

  set choices(BuiltSet<Choice> choices) {
    _choices = choices;
    notifyListeners();
  }
}

class LetzterStatusViewModel extends ChoiceChangeNotifier {}
```

```
final letzterStatus = BehaviorSubject<LetzterStatus?>.seeded(null);
```

```
StreamBuilder(
  stream: letzterStatus,
  builder: (context, snapshot) {
```

```
MultiProvider(
  providers: [
    Provider<MassnahmenFormViewModel>(create: (_) => MassnahmenFormViewModel()),
    Provider<MassnahmenJsonFile>(create: (_) => MassnahmenJsonFile()),
    Provider(
      create: (context) => MassnahmenModel(
        Provider.of<MassnahmenJsonFile>(context, listen: false))),
    ChangeNotifierProvider(create: (context) => LetzterStatusViewModel()),
  ],
  child: MaterialApp(),
)
```

```
Consumer<LetzterStatusViewModel>(
  builder: (context, choiceChangeNotifier, child) {
  },
)
```

```
class ChoiceChangeNotifier extends ChangeNotifier {  
    BuiltSet<Choice> _choices = BuiltSet<Choice>();  
  
    BuiltSet<Choice> get choices => _choices;  
  
    set choices(BuiltSet<Choice> choices) {  
        _choices = choices;  
        notifyListeners();  
    }  
}  
class LetzterStatusViewModel extends ChoiceChangeNotifier {}
```

```
        }
    }

class LetzterStatusViewModel extends ChoiceChangeNotifier {}
```

```
MultiProvider(
    providers: [
        Provider<MassnahmenFormViewModel>(create: (_) => MassnahmenFormViewModel()),
        Provider<MassnahmenJsonFile>(create: (_) => MassnahmenJsonFile()),
        Provider(
            create: (context) => MassnahmenModel(
                Provider.of<MassnahmenJsonFile>(context, listen: false)),
            ChangeNotifierProvider(create: (context) => LetzterStatusViewModel())),
    ],
    child: MaterialApp(),
)
```

```
Consumer<LetzterStatusViewModel>(
    builder: (context, choiceChangeNotifier, child) {
```

```
」,  
child: MaterialApp(),  
)
```

```
Consumer<LetzterStatusViewModel>(  
    builder: (context, choiceChangeNotifier, child) {  
    },  
)
```

```
final letzterStatus = BehaviorSubject<LetzterStatus?>.seeded(null);
```

```
StreamBuilder(  
    stream: letzterStatus,  
    builder: (context, snapshot) {
```

# provider oder RxDart

```
class CartController {
  final Cart _cart = Cart();
  final StreamController<Cart> _additionalController = StreamController<Cart>();
  final StreamController<Product> _productController = StreamController<Product>();
  final StreamController<ItemCount> _itemCountController = StreamController<ItemCount>();
  final BehaviorSubject<List<Product>> _productsSubject = BehaviorSubject<List<Product>>();
  final BehaviorSubject<String> _searchSubject = BehaviorSubject<String>();
  final BehaviorSubject<String> _errorSubject = BehaviorSubject<String>();
  final BehaviorSubject<Cart> _cartSubject = BehaviorSubject<Cart>();
  final BehaviorSubject<Cart> _cartWithItemsSubject = BehaviorSubject<Cart>();

  void addProduct(Product product) {
    _productsSubject.add(product);
    _cartWithItemsSubject.add(_cart.cartWithItems);
  }

  void addAdditionalProduct(Cart additionalCart) {
    _additionalController.stream.listen((Cart cart) {
      _cartWithItemsSubject.add(cart);
    });
  }

  void search(String query) {
    _searchSubject.add(query);
  }

  void removeProductFromCart(Product product) {
    _cartWithItemsSubject.add(_cart.removeProductFromCart(product));
  }

  void updateCart(Cart cart) {
    _cartSubject.add(cart);
  }
}
```

```
class ChoiceChangeNotifier extends ChangeNotifier {
  BuiltSet<Choice> _choices = BuiltSet<Choice>();
  BuiltSet<Choice> get choices => _choices;
  set choices(BuiltSet<Choice> choices) {
    _choices = choices;
    notifyListeners();
  }
}

class LetzterStatusViewModel extends ChoiceChangeNotifier {}

final letzterStatus = BehaviorSubject<LetzterStatus?>.seeded(null);

StreamBuilder(
  stream: letzterStatus,
  builder: (context, snapshot) {
```

```
MultiplexProvider(
  providers: [
    Provider<MassnahmenFormViewModel>(create: (_) => MassnahmenFormViewModel()),
    Provider<MassnahmenJsonFile>(create: (_) => MassnahmenJsonFile()),
    Provider<MassnahmenModel>(create: (context) => MassnahmenModel(),
      Provider.of<MassnahmenJsonFile>(context, listen: false)),
    Provider.of<MassnahmenModel>(context, listen: false),
    ChangeNotifierProvider(create: (context) => LetzterStatusViewModel()),
  ],
  child: MaterialApp(),
)

Consumer<LetzterStatusViewModel>(
  builder: (context, choiceChangeNotifier, child) {
    return child;
  }
)
```

```
class CartBloc{  
    final _cart = Cart();  
  
    Sink<Product> get addition => _additionController.sink;  
  
    final _additionController = StreamController<Product>();  
  
    Stream<int> get itemCount => _itemCountSubject.stream;  
  
    final _itemCountSubject = BehaviorSubject<int>();  
  
    CartBloc(){  
        _additionController.stream.listen(_handle);  
    }  
  
    void _handle(Product product){  
        _cart.add(product);  
        _itemCountSubject.add(_cart.itemCount);  
    }  
}
```

Quelle: Google LLC, Build reactive mobile apps with Flutter (Google I/O '18) TC: 27:37

```
class CartBloc{  
    final _cart = BehaviorSubject<Cart>(seedValue: Cart());  
  
    addProduct(Product product) => _cart.value = _cart.value..add(product);  
  
    Stream<int> get itemCount => _cart.map((cart) => cart.itemCount);  
}
```

```
Sink<Product> get addition => _additionalController.sink;  
  
final _additionalController = StreamController<Product>();  
  
Stream<int> get itemCount => _itemCountSubject.stream;  
  
final _itemCountSubject = BehaviorSubject<int>();  
  
CartBloc(){  
    _additionalController.stream.listen(_handle);  
}  
  
void _handle(Product product){  
    _cart.add(product);  
    _itemCountSubject.add(_cart.itemCount);  
}
```

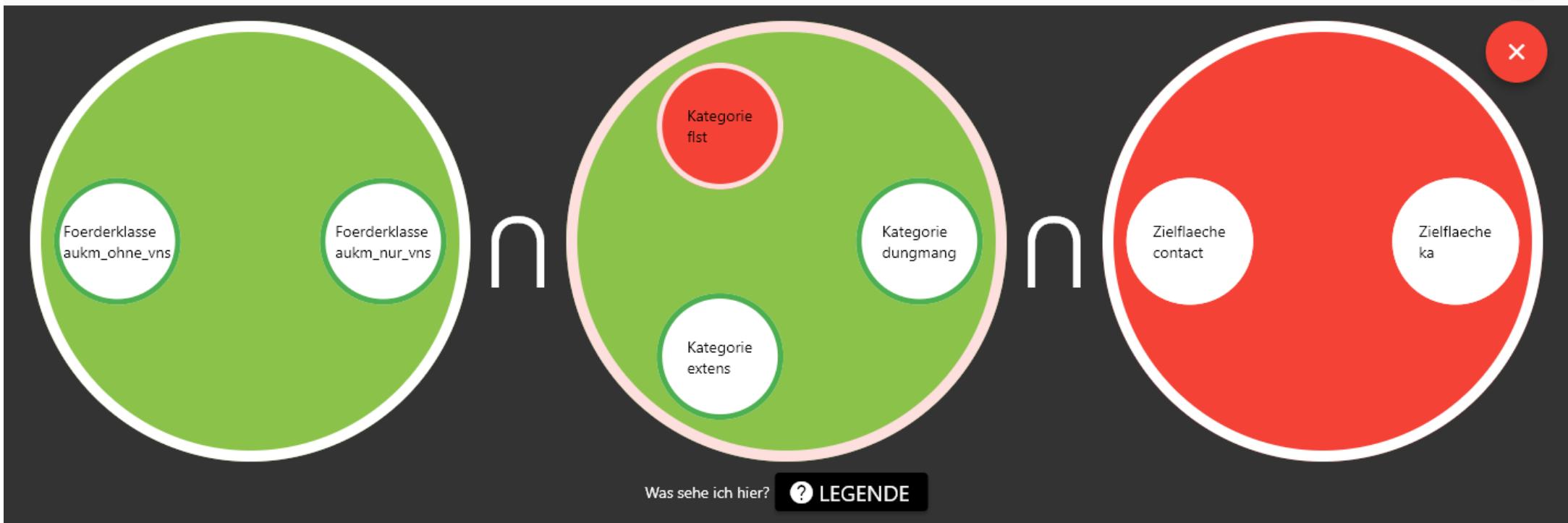
---

```
class CartBloc{  
    final _cart = Cart();  
  
    Sink<Product> get addition => _additionalController.sink;  
  
    final _additionalController = StreamController<Product>();  
  
    Stream<int> get itemCount => _itemCountSubject.stream;  
  
    final _itemCountSubject = BehaviorSubject<int>();  
  
    CartBloc(){  
        _additionaController.stream.listen(_handle);  
    }  
  
    void _handle(Product product){  
        _cart.add(product);  
        _itemCountSubject.add(_cart.itemCount);  
    }  
}
```

Quelle: Google LLC, Build reactive mobile apps with Flutter (Google I/O '18) TC: 27:37

```
class CartBloc{  
    final _cart = BehaviorSubject<Cart>(seedValue: Cart());  
  
    addProduct(Product product) => _cart.value = _cart.value..add(product);  
  
    Stream<int> get itemCount => _cart.map((cart) => cart.itemCount);  
}
```



[← Zieleinheit](#) keine Angabe/Vorgabe (i) ha (i) bitte um Unterstützung (i) m<sup>3</sup> (z.B. Gülle) 



Quelle: <http://www.artswfl.com/wp-content/uploads/2012/10/006-43.jpg>

