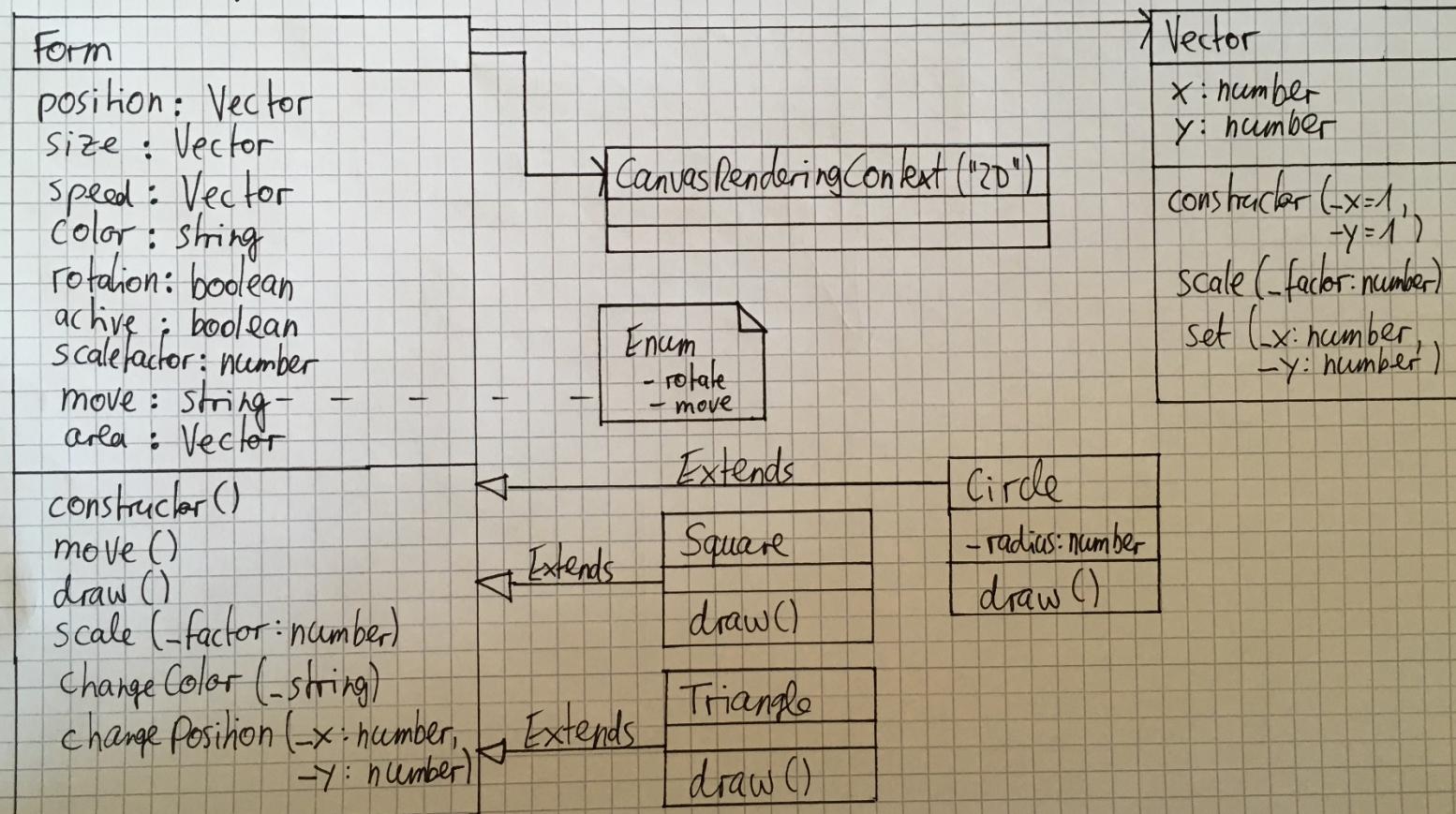
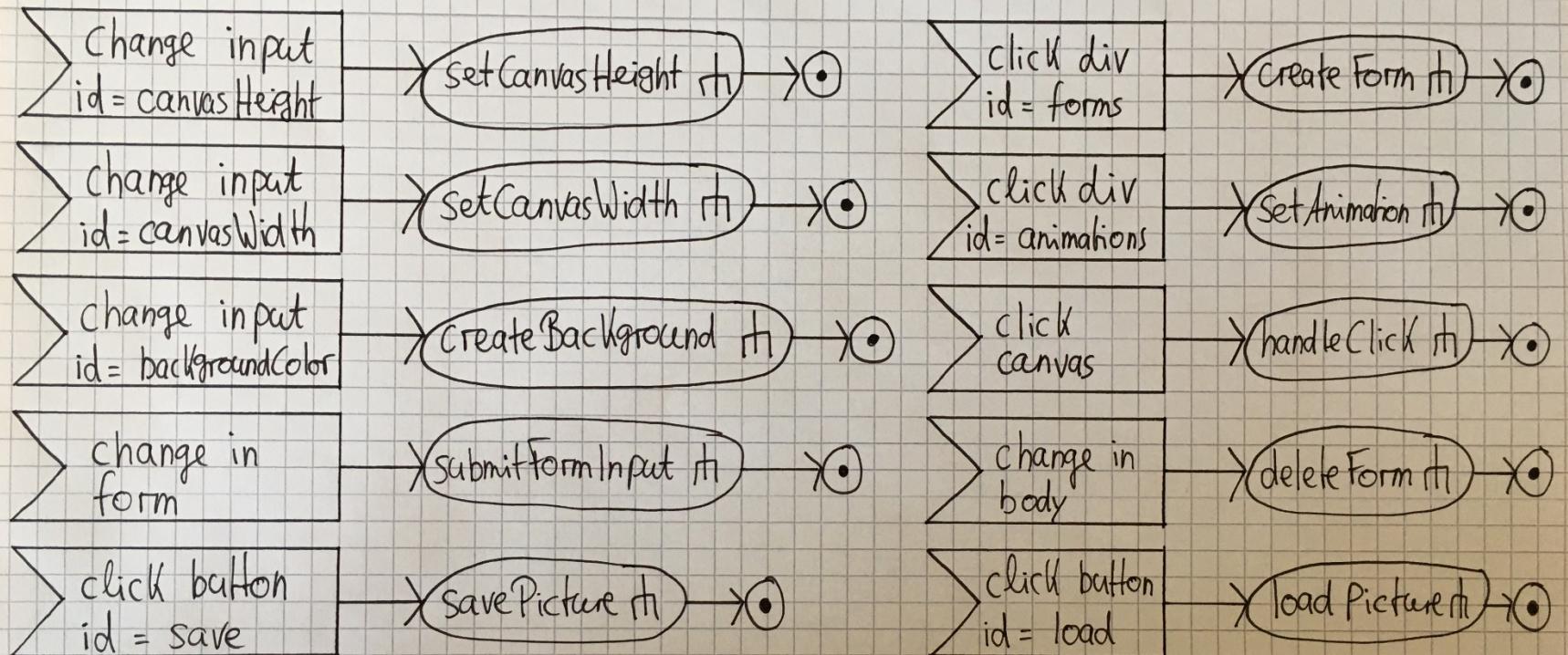


Klassendiagramm





```
canvas : HTMLCanvasElement  
Crc2 : CanvasRenderingContext2D("2D")  
height = canvas.height  
width = canvas.width  
figures : Form[]  
backgroundImageData : ImageData
```

load listener installieren
mit handleLoad fn

changePosition

```
-x : number  
-y : number
```

```
this.PositionX = -x  
this.PositionY = -y
```

scale

```
-factor : number
```

```
this.size = (-factor)
```

changeColor

```
-newColor : string
```

```
this.color = -newColor
```

handle Load

```
Canvas = get canvas  
CFC2 = canvas.getContext ("2D")
```

click- und change listener
installieren und Input-
felder erstellen

500x500 Rechteck
erschaffen

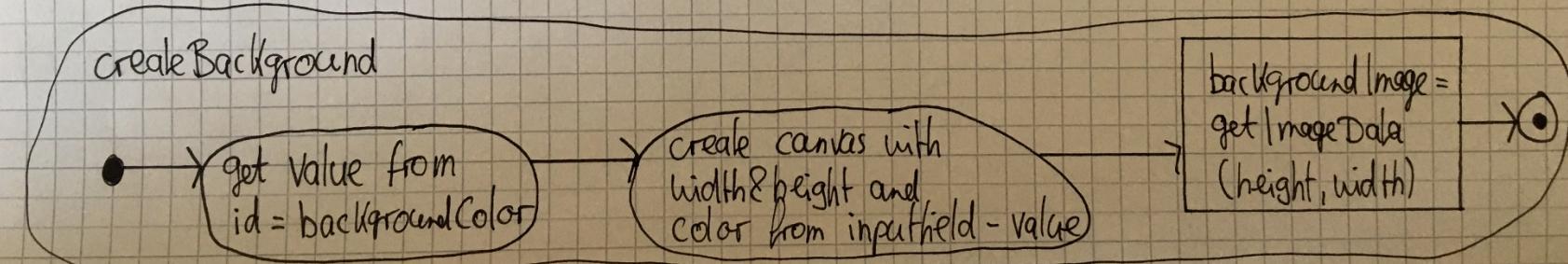
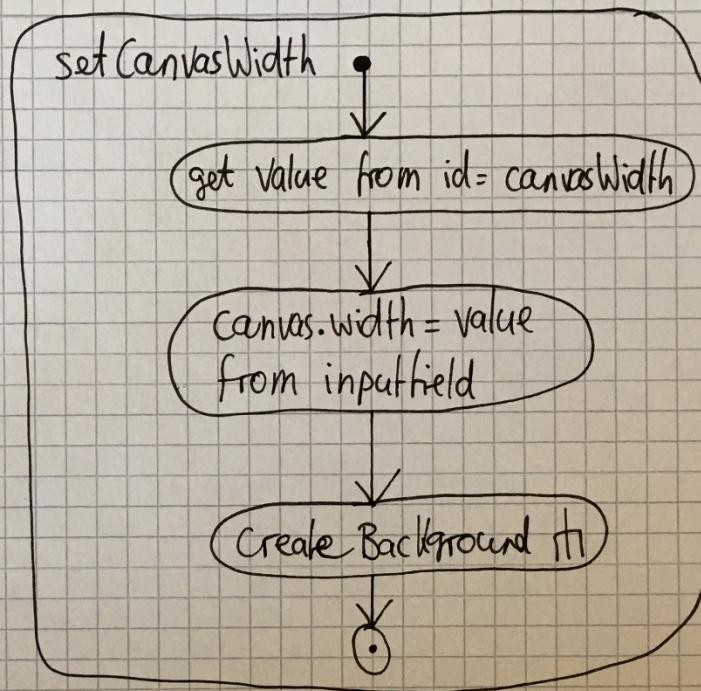
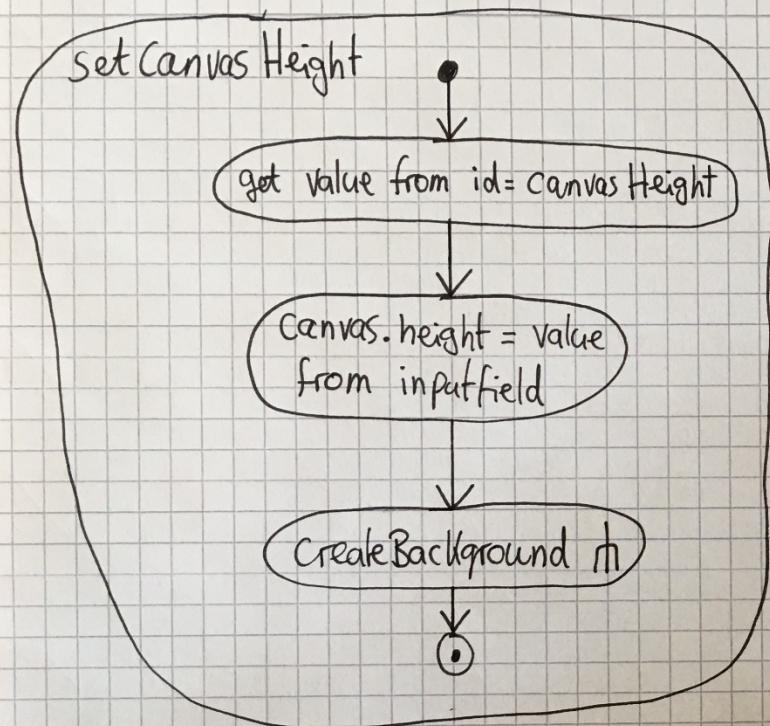
animate it

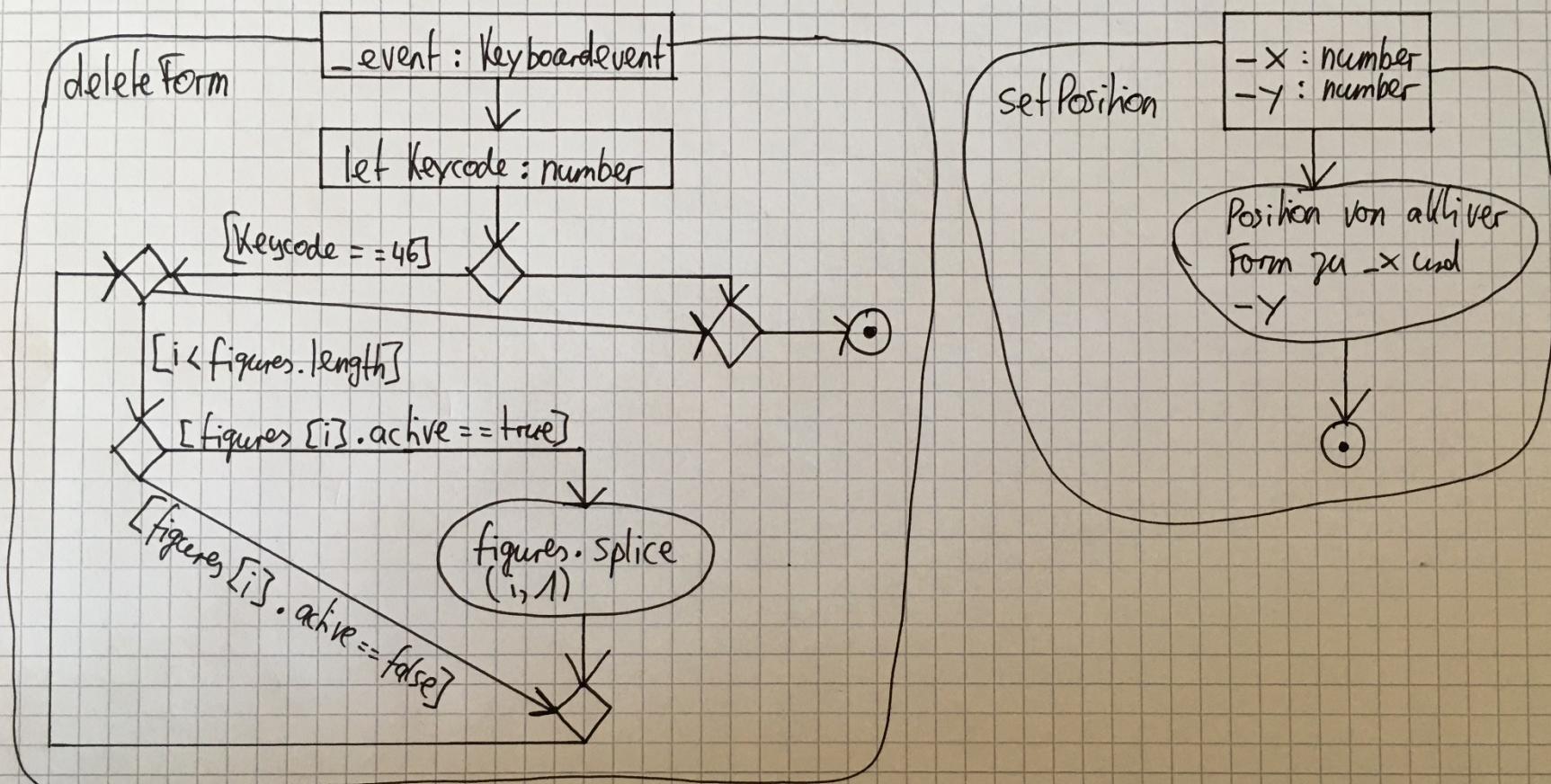
load Picture

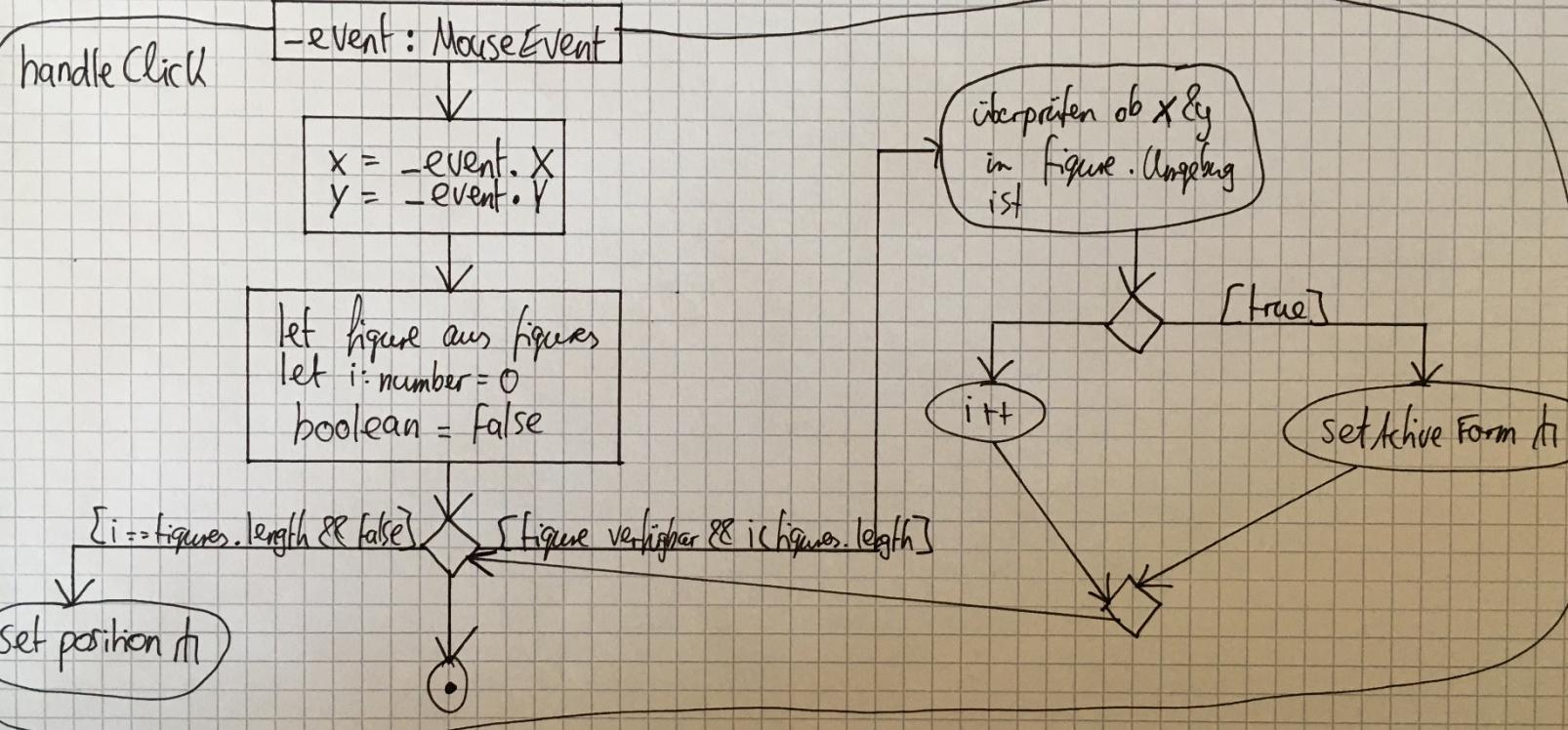
-event: Event

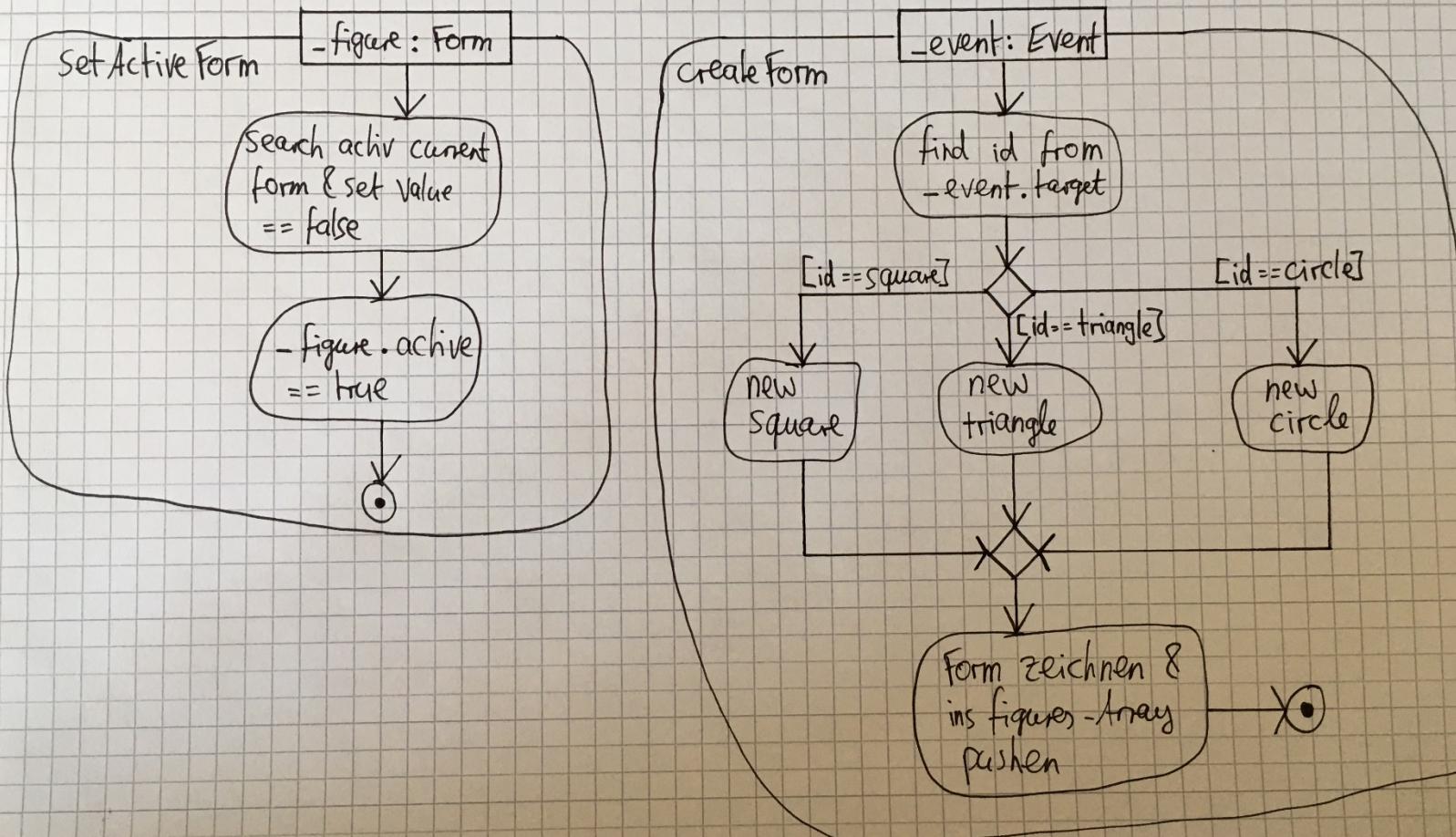
Anhage am Server
verschicken, um
gewünschtes Bild
zu bekommen

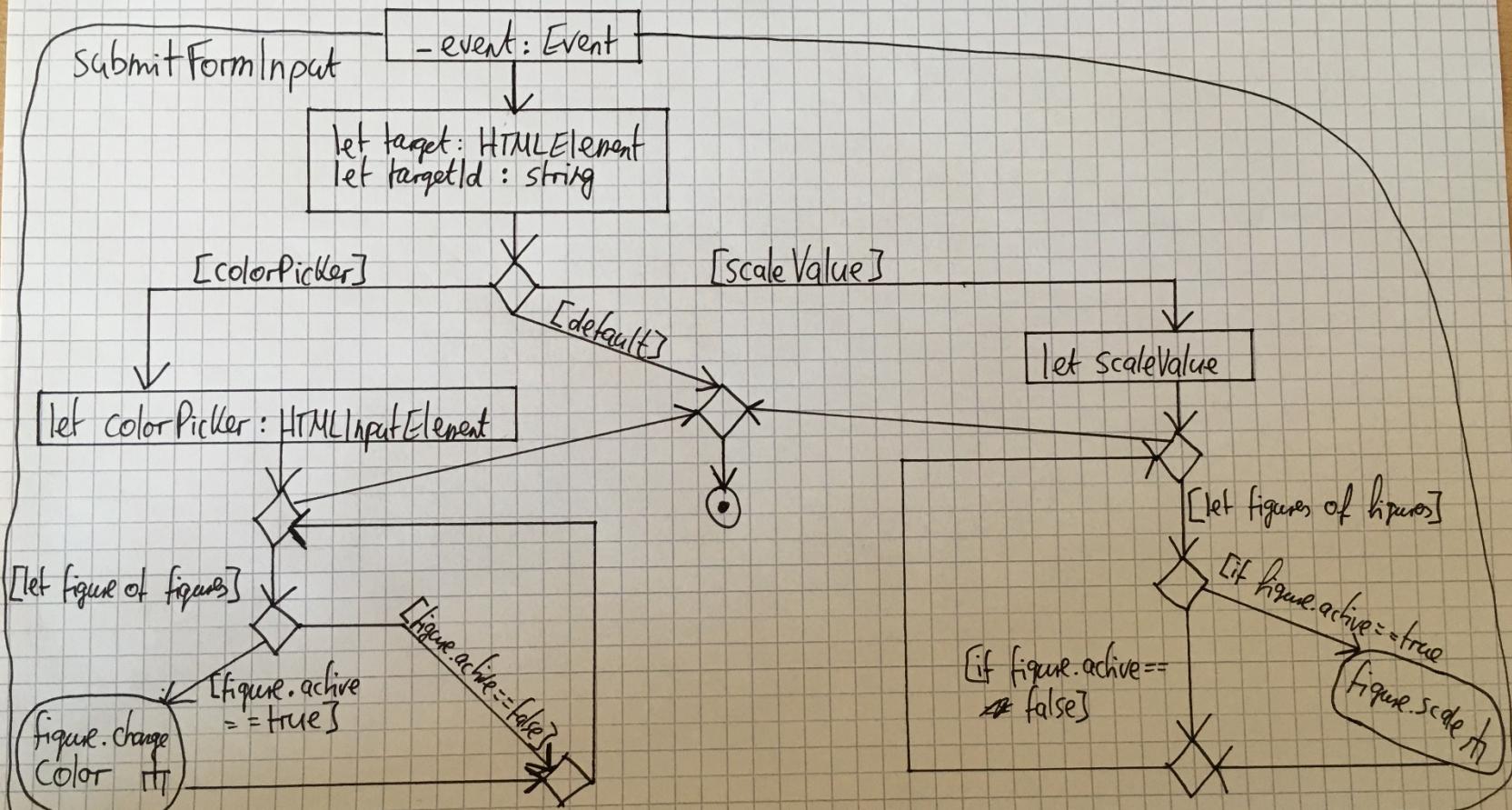
Bilder in der
datalist mit
id = library
anzeigen











constructor

```
this.active = true  
this.rotation = 0  
this.position = new Vector  
width/2 & height/2  
this.size = new Vector (50,50)  
this.moveType = "move"  
this.color = #fff
```

animate

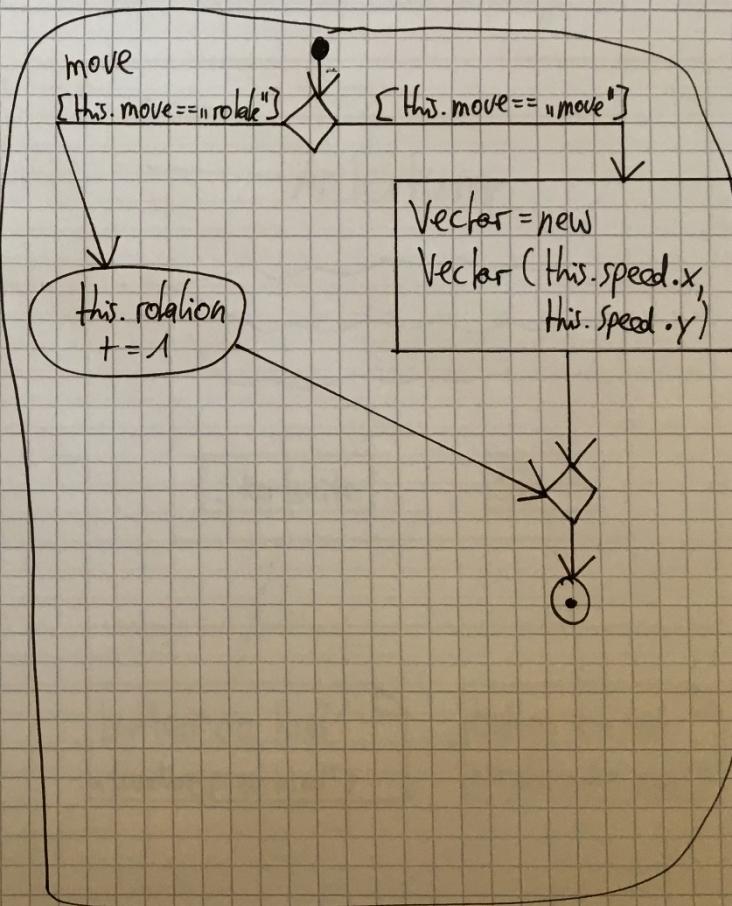
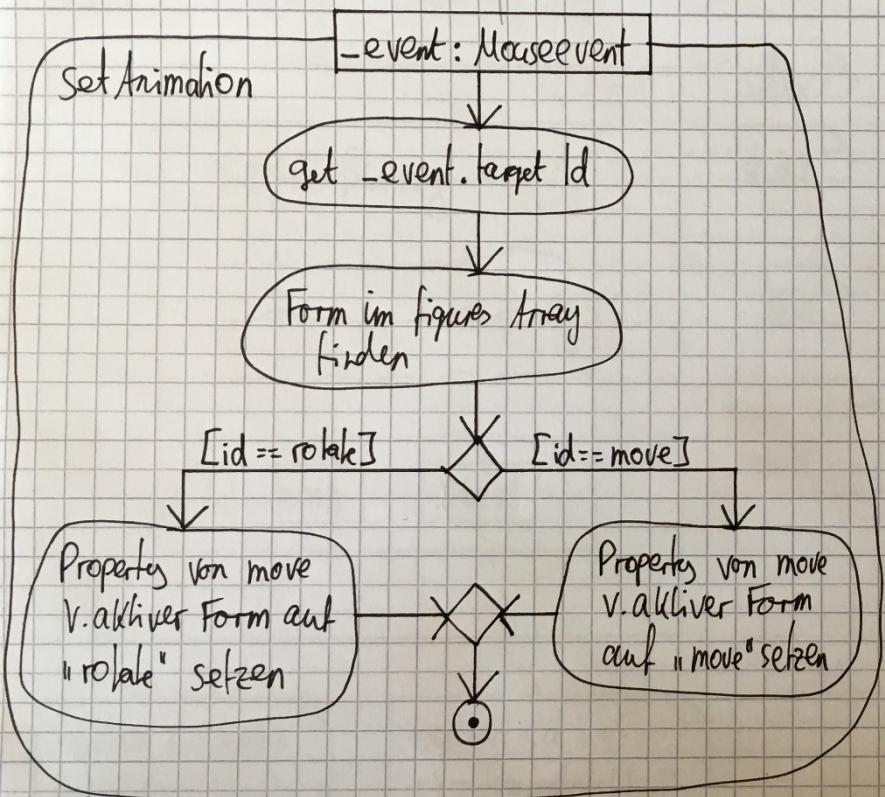
```
crc2.putImageData  
(backgroundImage,0,0)
```

let figure at figures

[figure verfügbar]

figure.move(1/2)

figure.draw()



update FormList

[if child available]

first child aus
AllForms löschen

let title: HTML
SpanElement

neuen Span
erstellen

neuen Span in
Übersicht-div
einfügen

let list: HTMLSpanElement

Typ anzeigen mit Farbvalue
aus Farbfeld

.addEventlistener auf span
mit setvalue funktion

an Übersicht-div anfügen