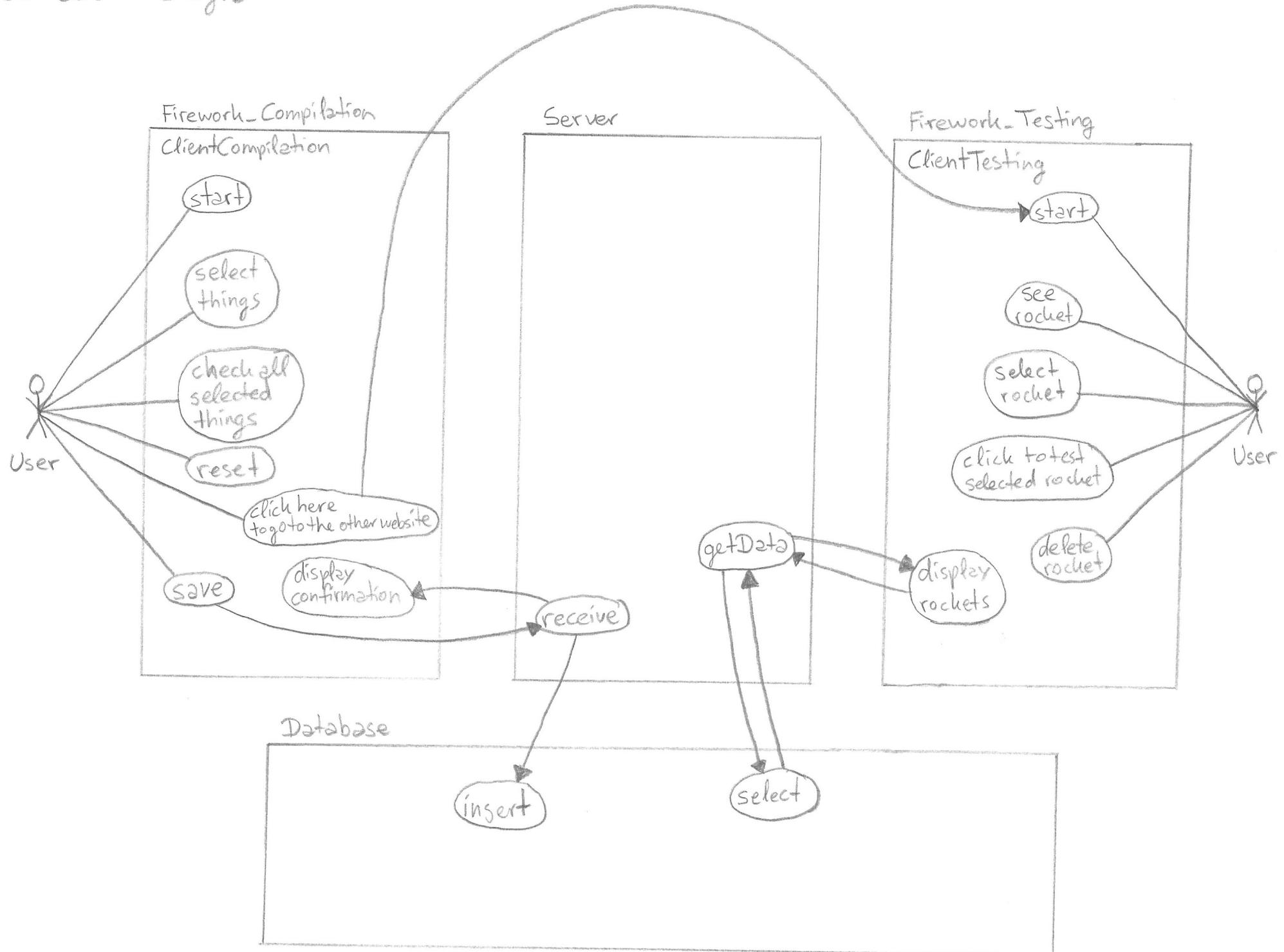
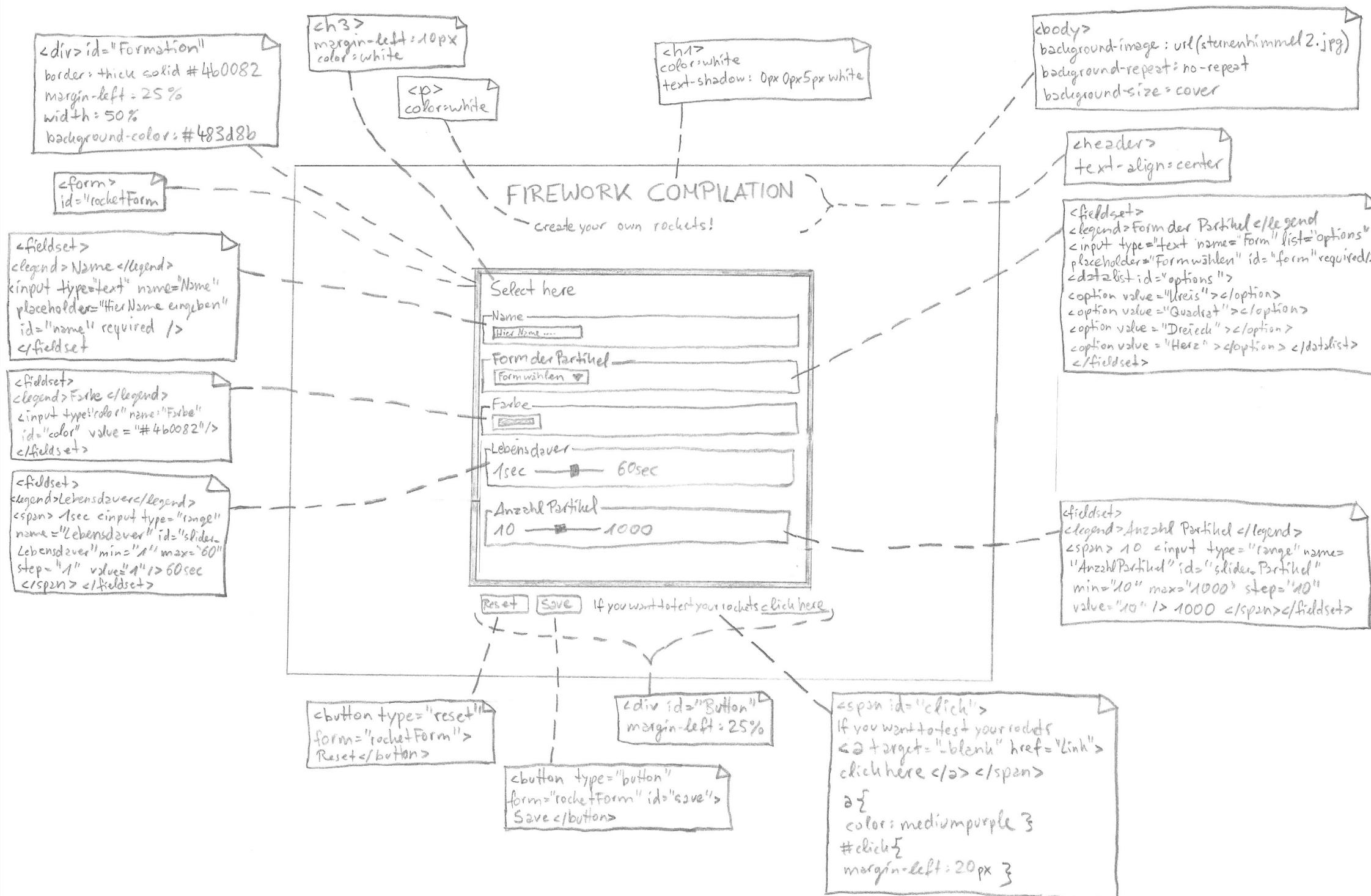


# Use Case Diagram : Firework



# User-Interface Scribble : Firework-Compilation



# User-Interface Scribble : Firework

Wenn keine Rahmen  
in der Datenbank sind  
steht hier <span>No rockets  
created</span>

```
button class="Current"  
background-color:#000033  
width:75%  
color:white  
border-radius:0px  
  
button class="rest"  
background-color:white  
color:#000033  
width:75%  
border-radius:0px
```

```
<span id="click">If you  
want to create more  
rockets <a href="index.html"  
>click here</a></span>
```

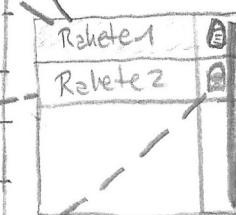
```
button class="rest+Del"  
background-color:white  
color:#000033  
width:25%  
border-radius:0px
```

<h2>  
color:white

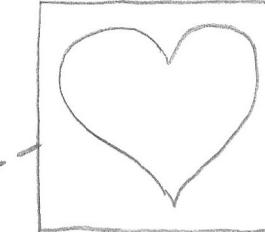
```
#input  
width:100%  
height:100%  
overflow-y:scroll
```

```
CANVAS  
background-color: #000033  
position: absolute  
margin: 0 auto  
top: 0  
bottom: 0  
left: 0  
right: 0
```

0,0  
Choose your  
Rocket

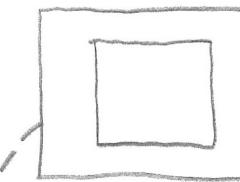


If you want to  
create more  
rockets click here



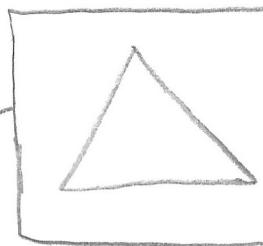
mit  
bezierCurveTo()

fillStyle = this.farbe



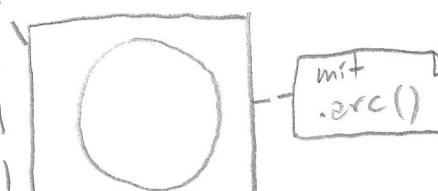
mit  
fillRect (0,0,7,7)

fillStyle = this.farbe



mit  
lineTo()

fillStyle = this.farbe

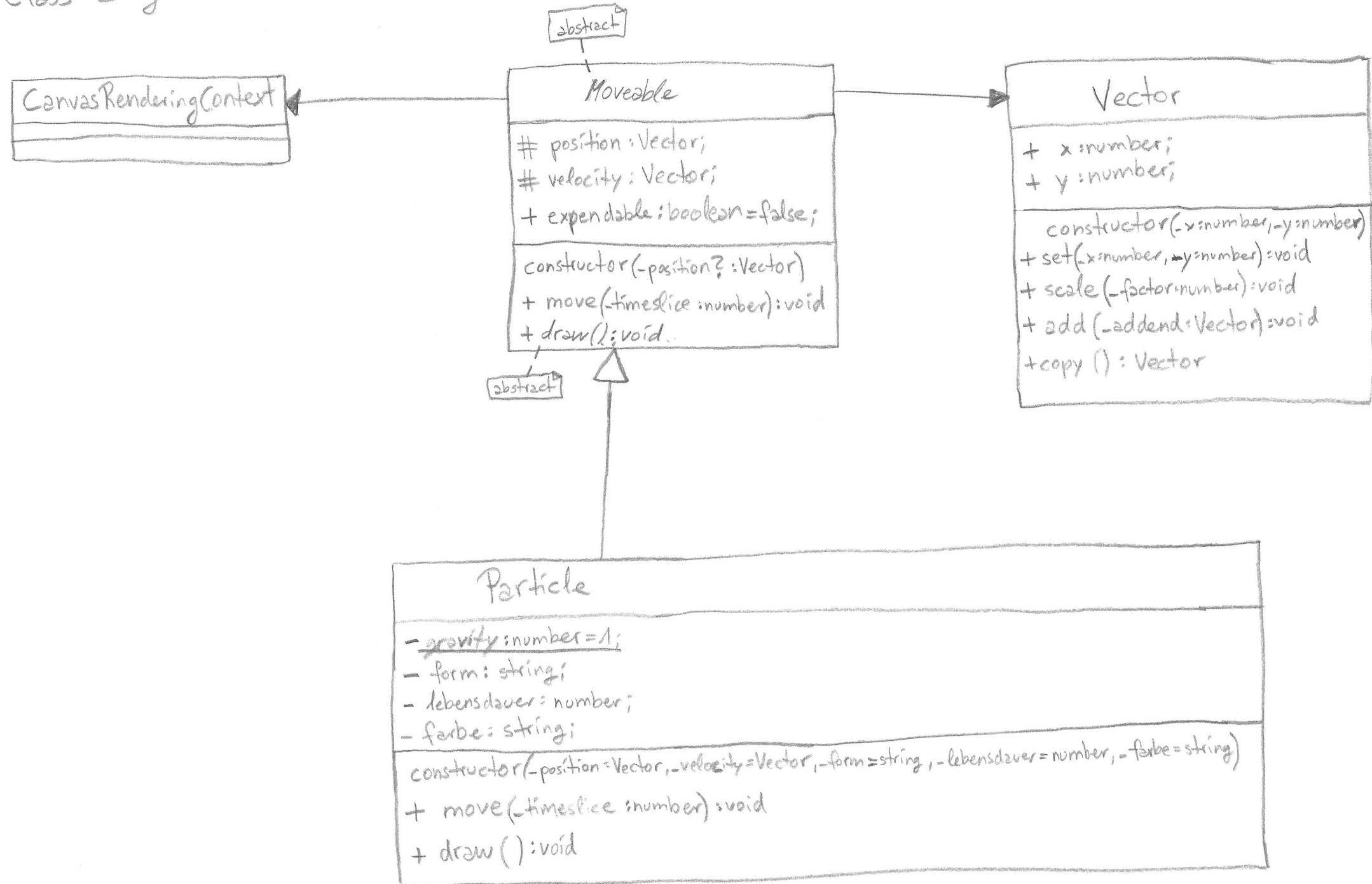


mit  
.arc()

fillStyle = this.farbe



# Class Diagram : Firework



# Activity Diagram : Firework

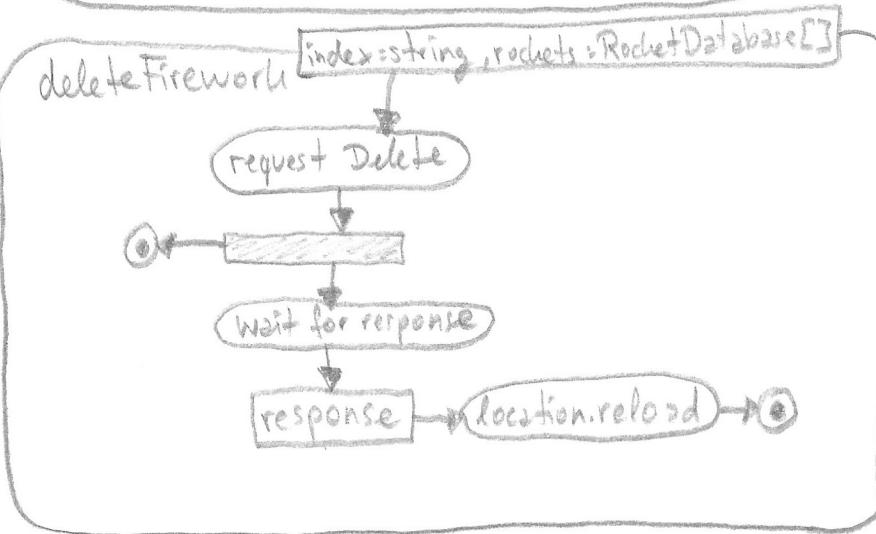
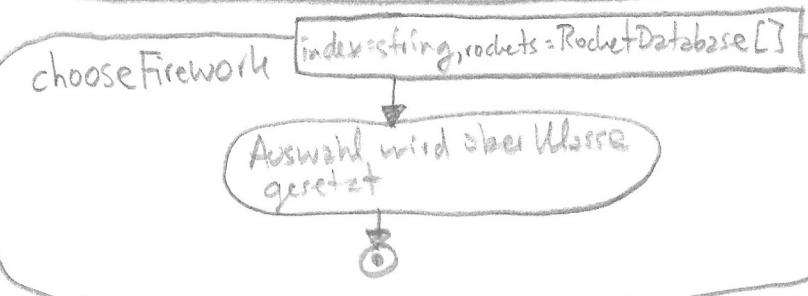
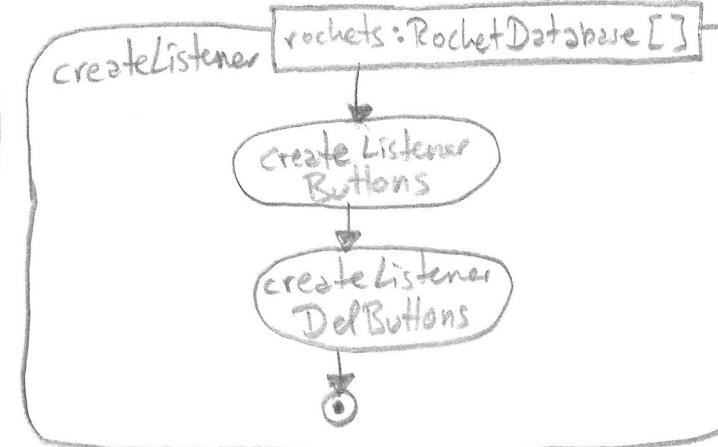
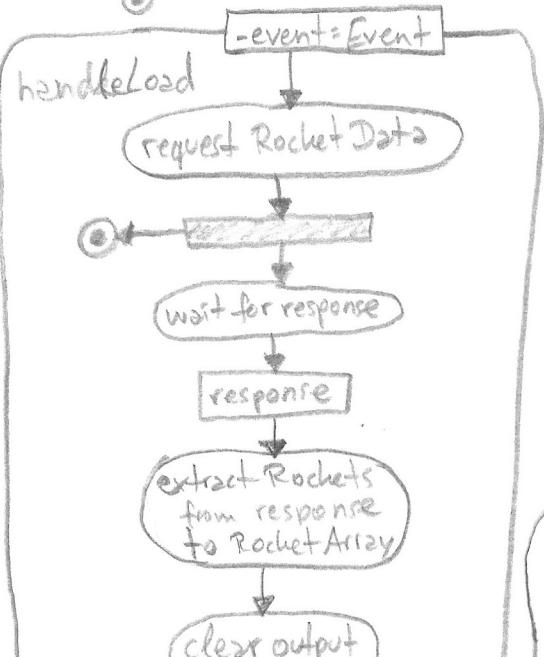
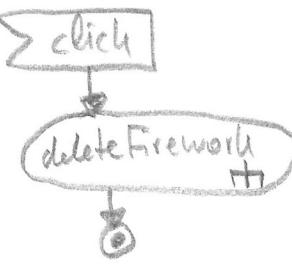
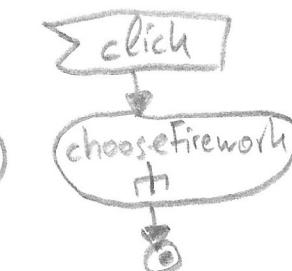
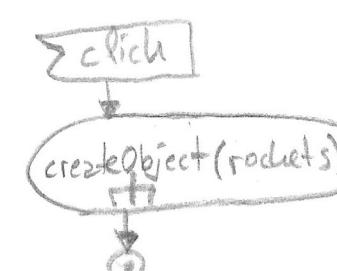
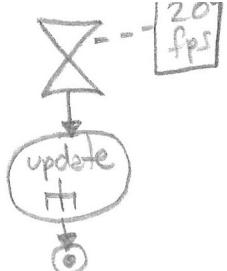
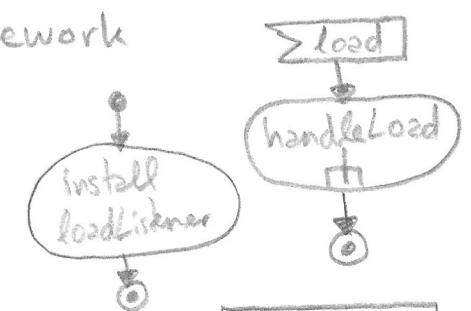
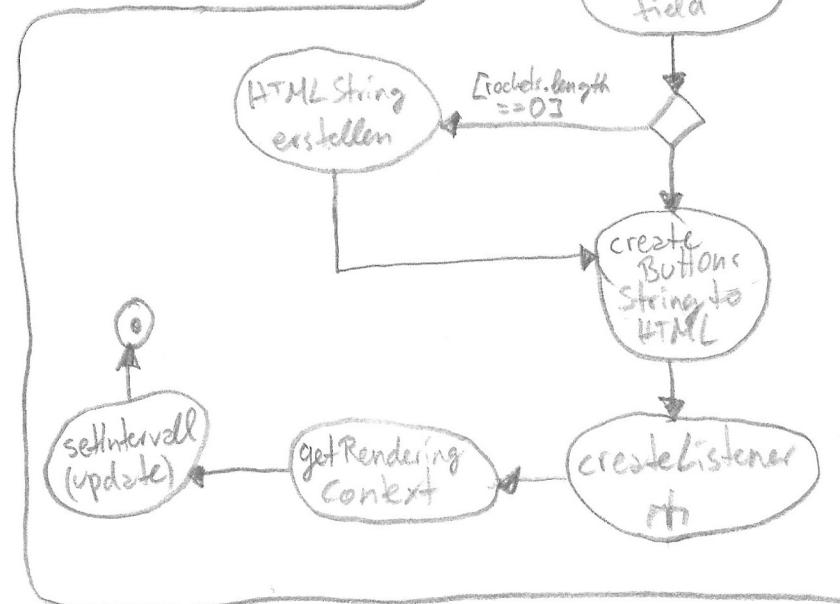
Generate\_Content

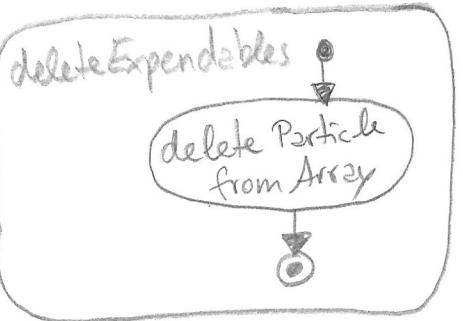
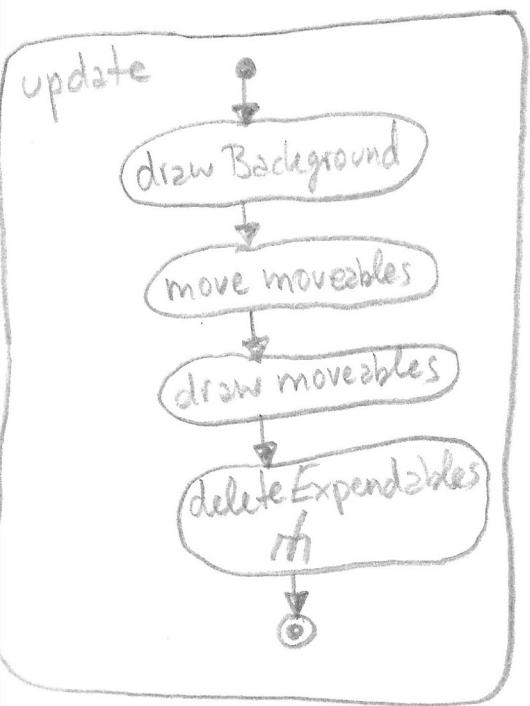
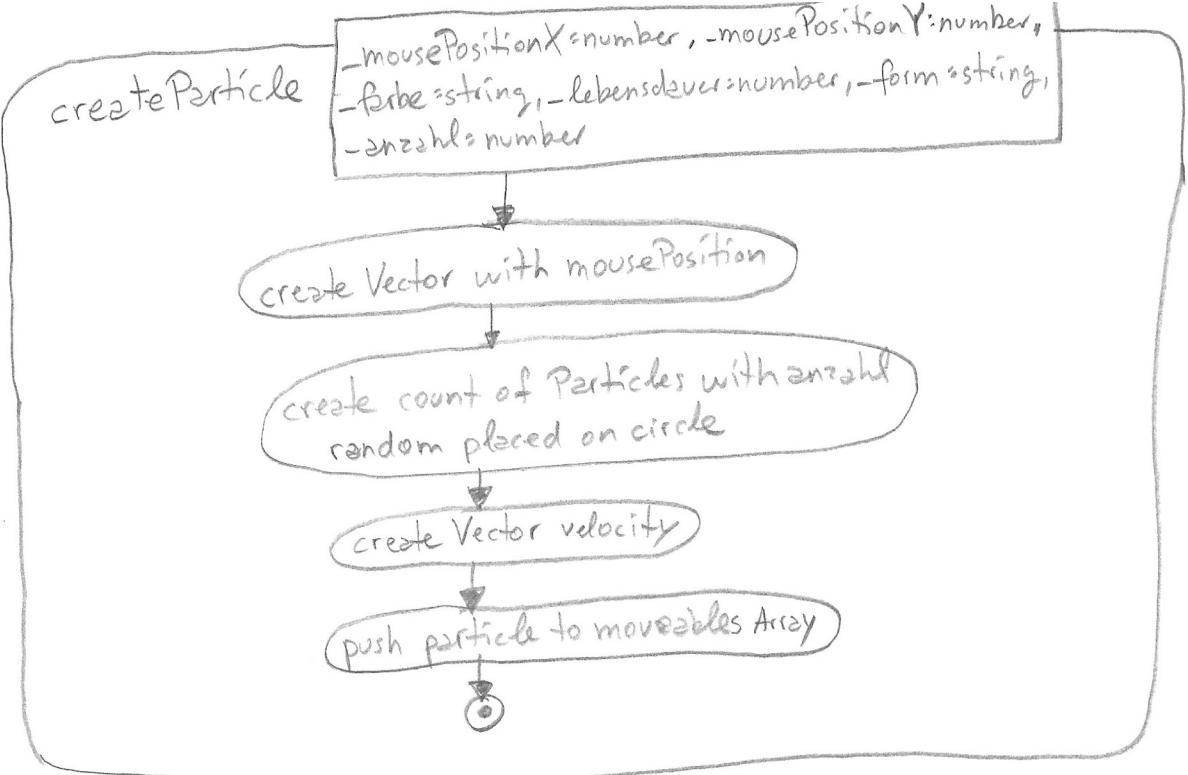
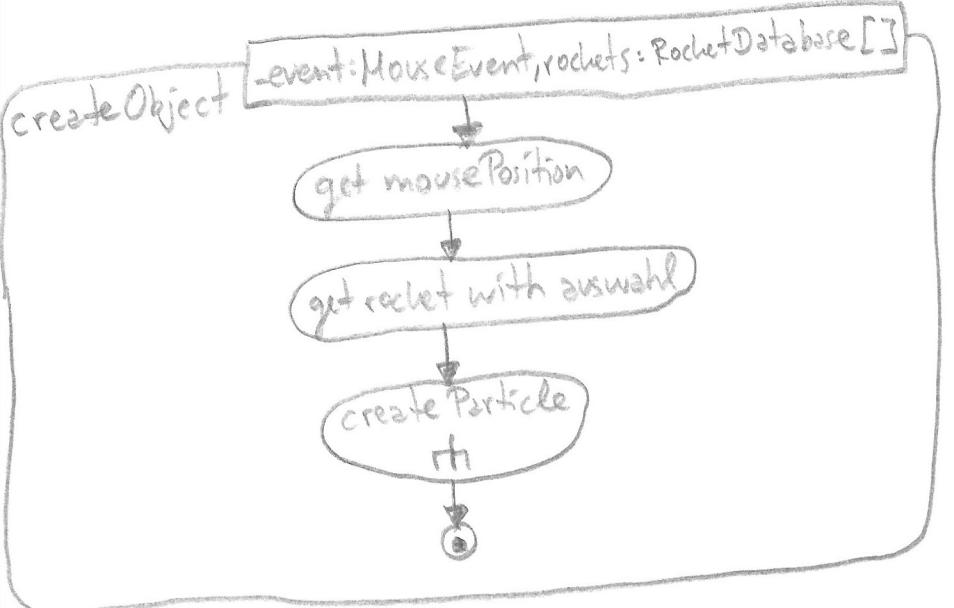
```
interface DeleteItem
    name: string;
```

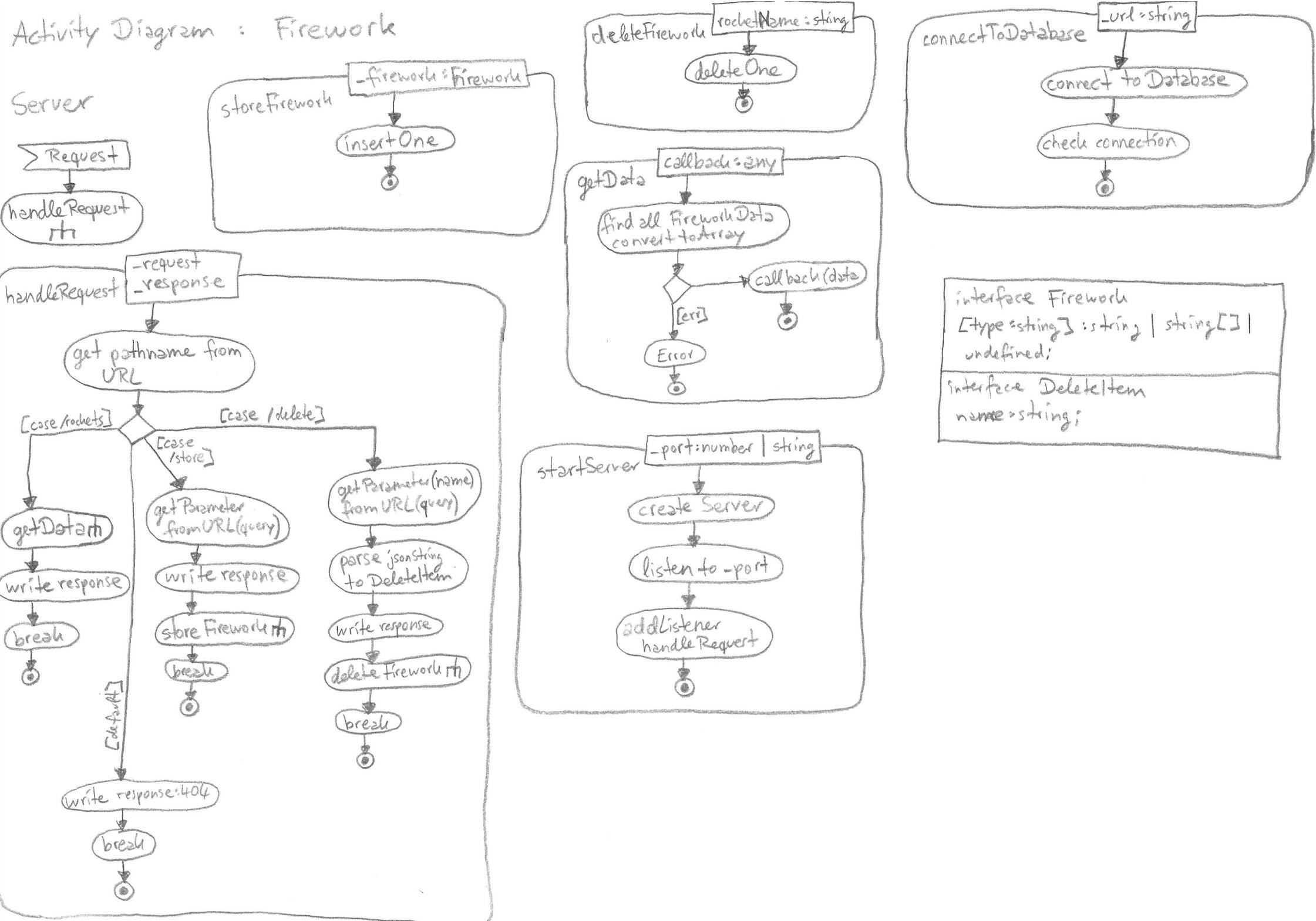
```
interface RocketDatabase
    ID: string;
    Name: string;
    Form: string;
    Farbe: string;
    Lebensdauer: number;
    AnzahlPartikel: number;
```

Globals

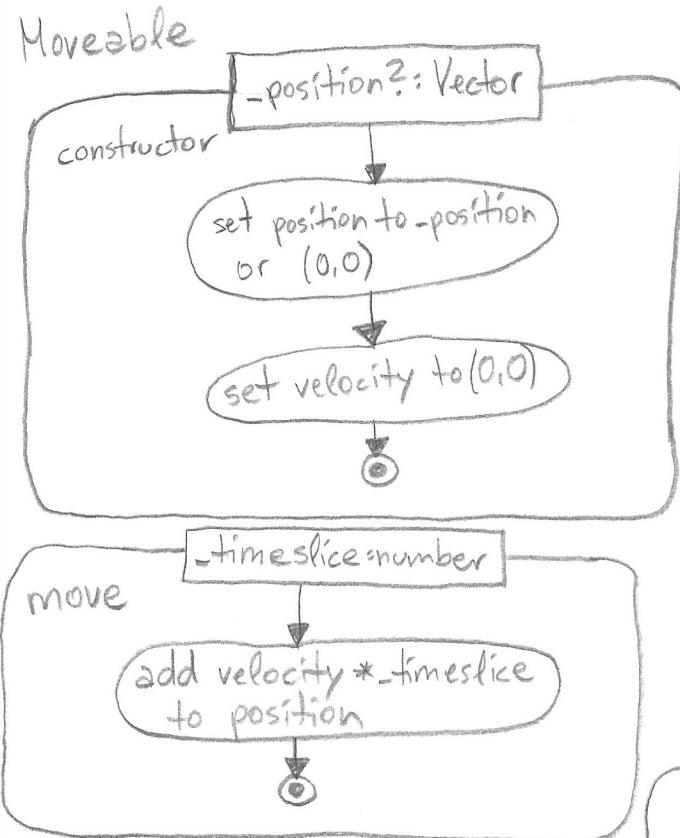
```
let url: string = herokuAppLink
let auswahl: number = 0
cre2
let moveables: Moveable[] = [];
```







# Activity Diagram > Firework



**draw**

```

graph TD
    draw(( ))
    final(( ))
    draw --> final

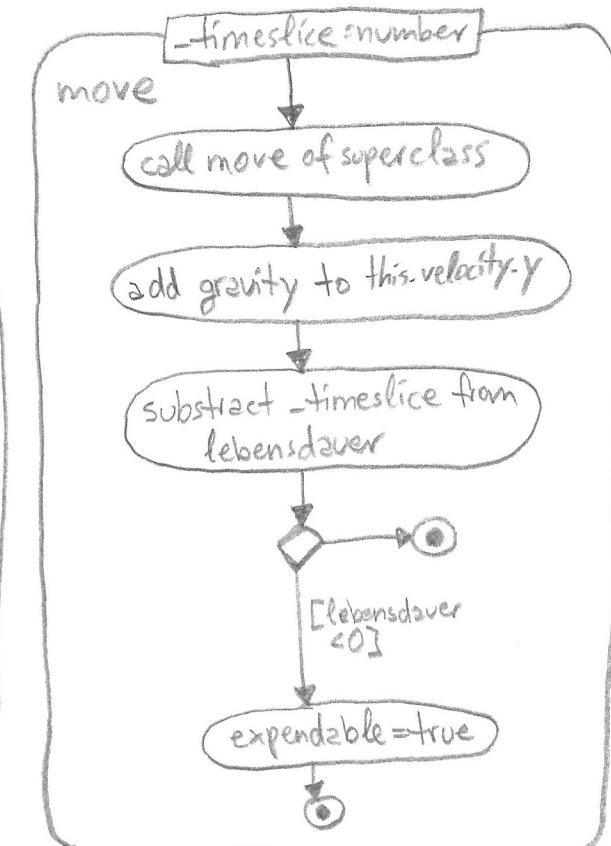
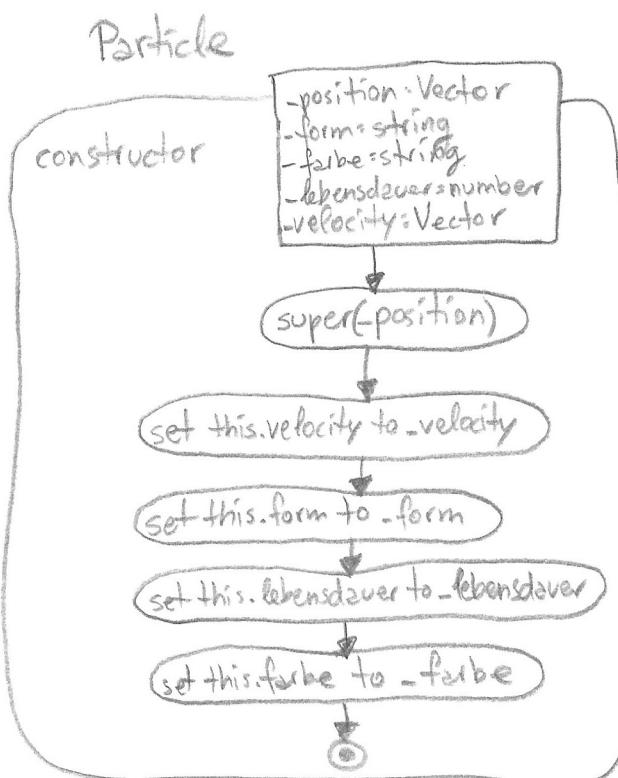
```

**Globals**

```

position:Vector;
velocity:Vector;
expendable:boolean=false;

```



**Globals**

```

gravity: number=1;
form:string;
lebensdauer: number;
farbe:string;

```

