

# UI - Scribble : Firework

<input>  
type = "hidden"  
id = "check"

<input>  
type = "text"  
id = "name"  
required

<select>  
name = "color"  
id = "color"

<input>  
type = "radio"  
name = "pattern"  
id = "circle"/"star"/"cross"

<input>  
type = "range"  
id = "size"

<input>  
type = "range"  
id = "lifespan"

<button>  
type = "button"  
id = "createButton"  
>click

<button>  
type = "button"  
id = "saveButton"  
>click

Create your Firework

Name

Color

Pattern

Size

Lifespan

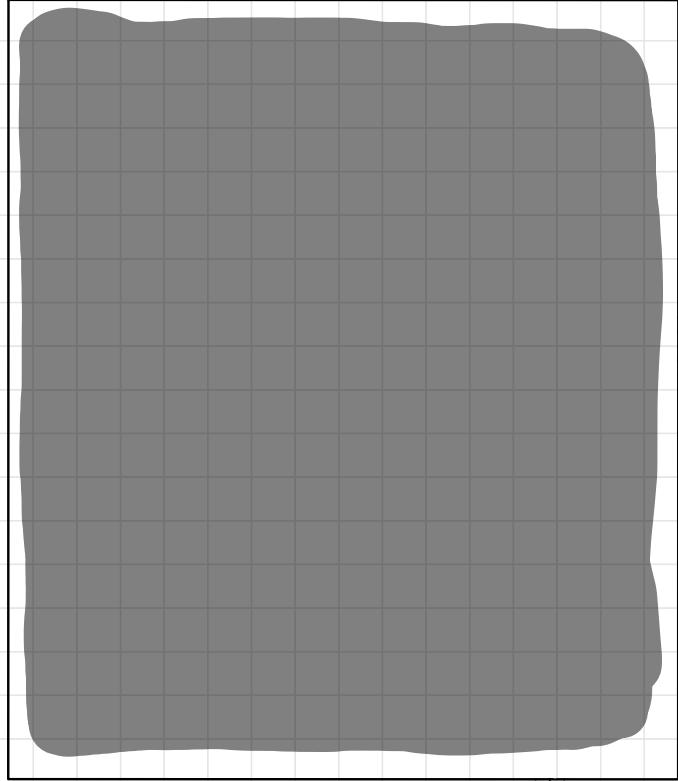
Create

Save current Firework

<div>  
id = "create"  
>change

<img>  
with id

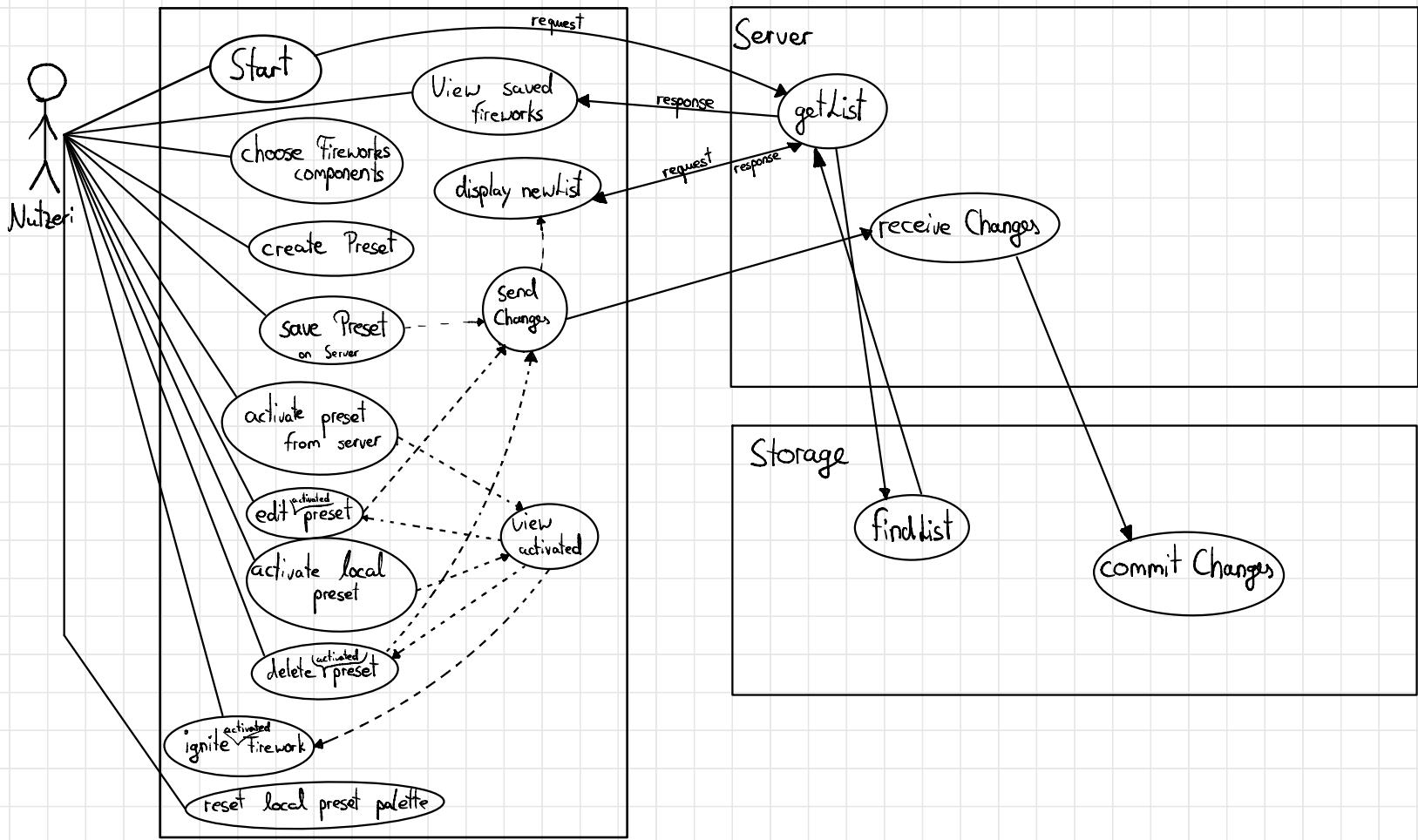
<canvas>  
width = "800"  
height = "55"  
>click



<button>  
type = "reset"  
id = "reset"  
>click

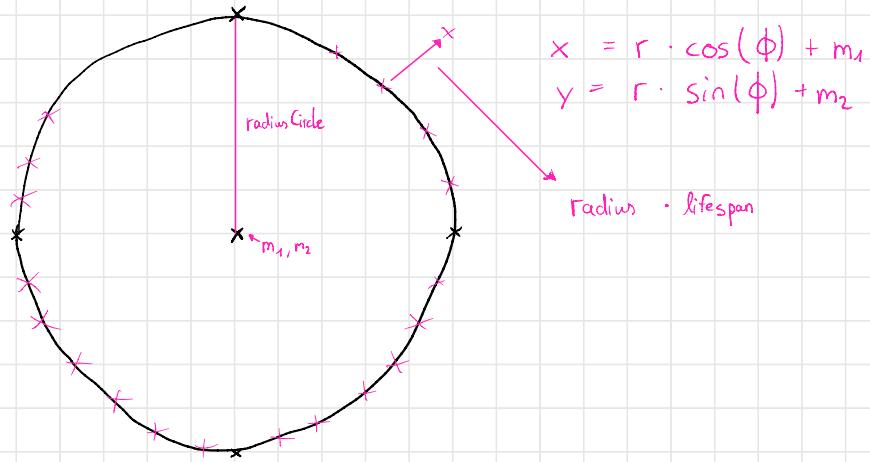
<li>  
id = "listElement" + index  
>click

# Use-Case - Diagram



# Firework: Canvas

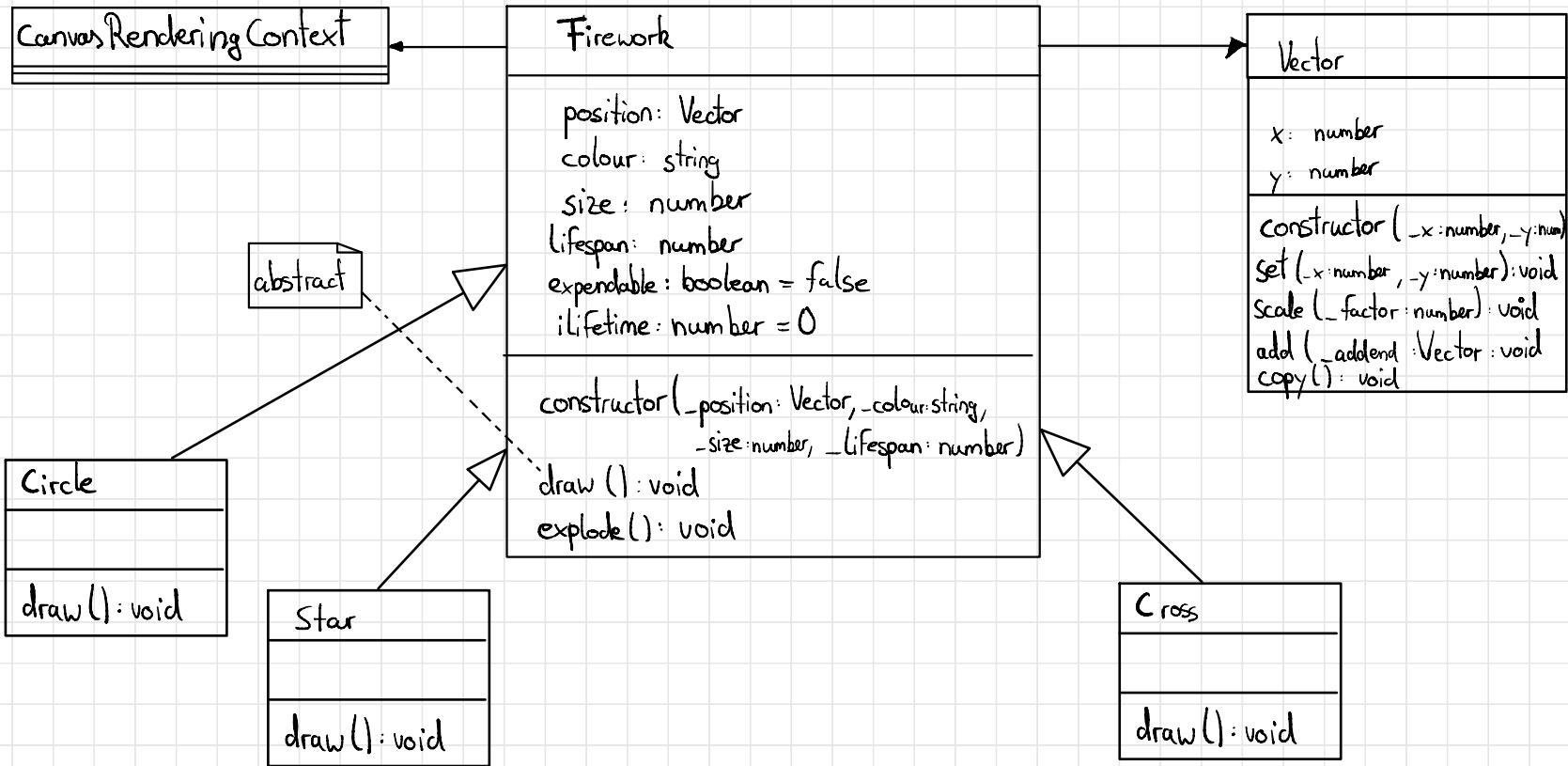
Circle



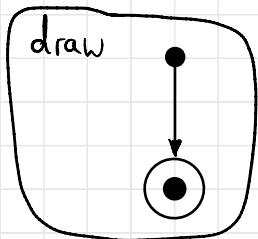
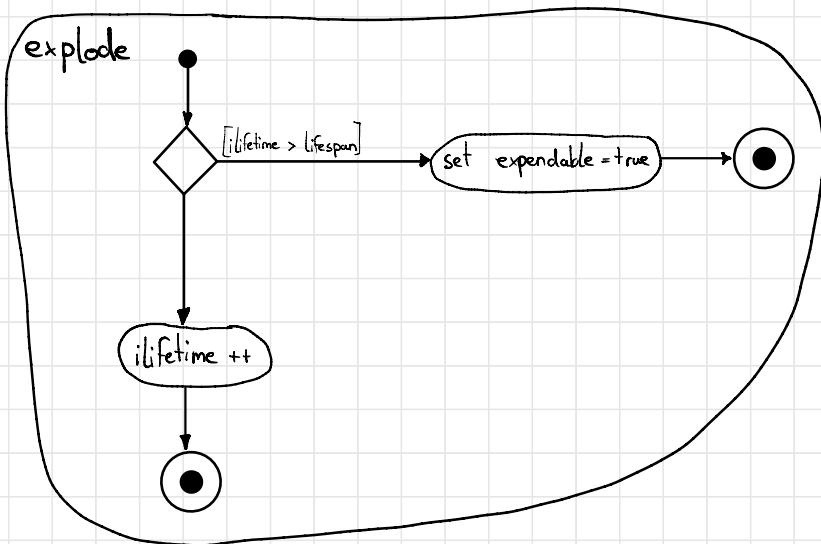
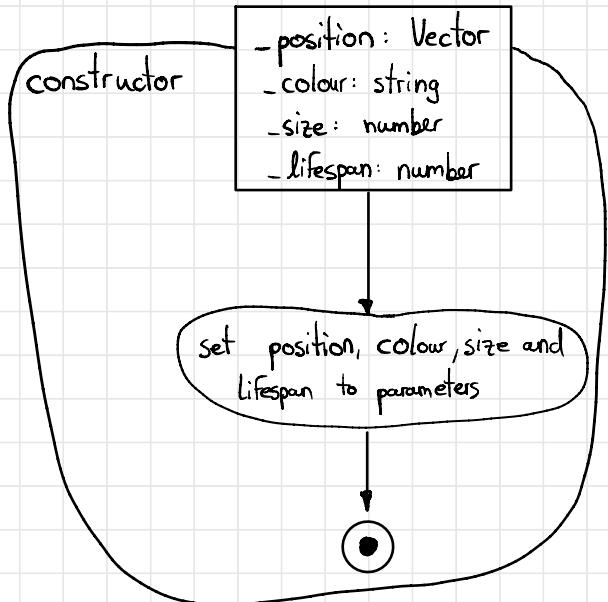
$$x = r \cdot \cos(\phi) + m_1$$
$$y = r \cdot \sin(\phi) + m_2$$

Radius + lifespan

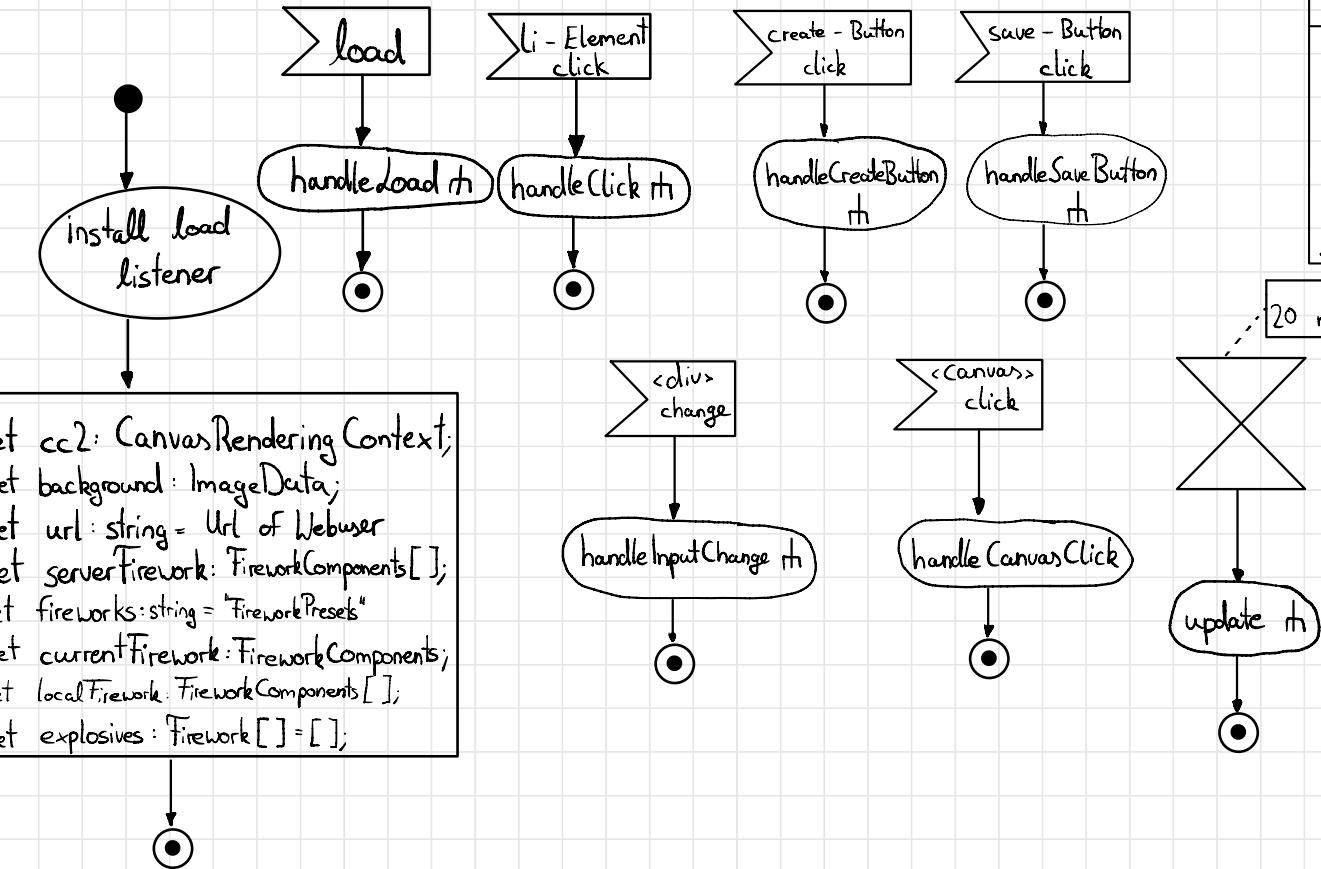
# Firework: Class - Diagram



# Firework : Activity Diagram - Firework



# Activity Diagram: Firework



**Interface: FireworkComponents**

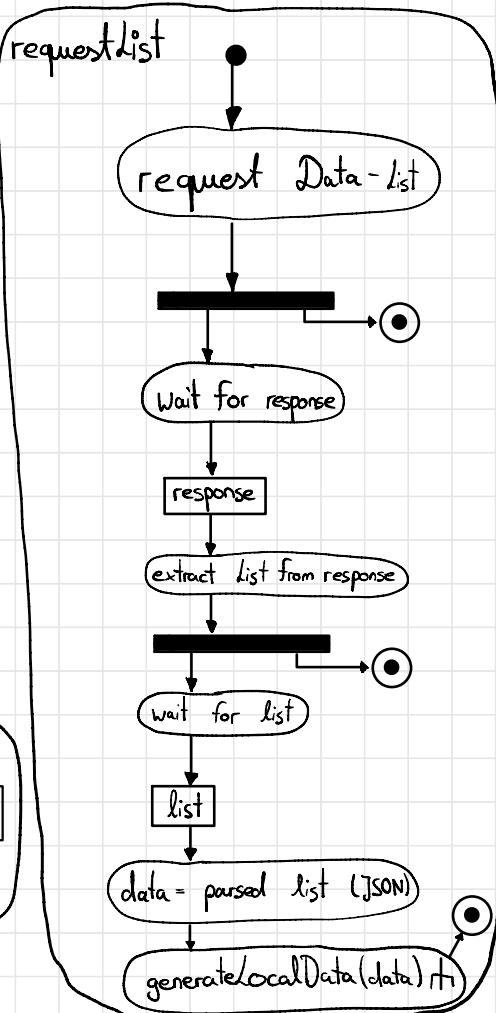
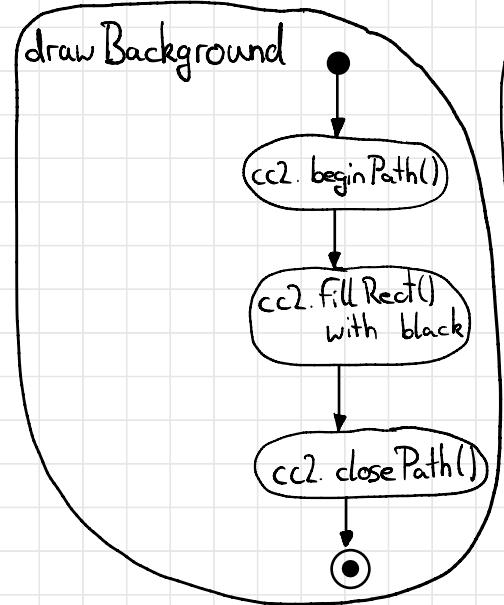
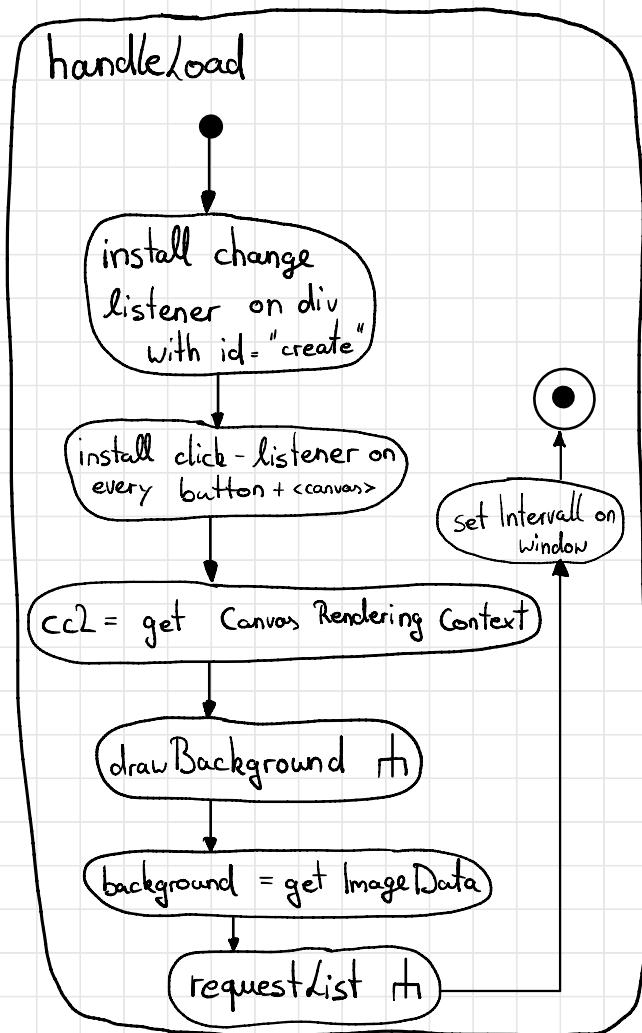
```

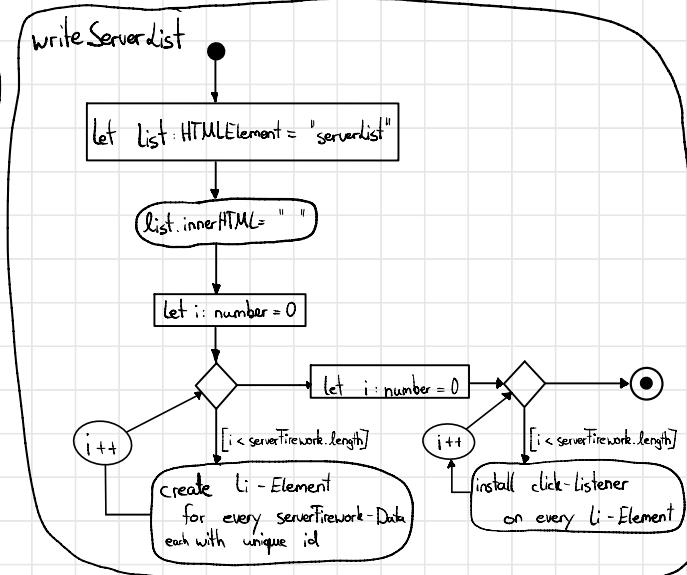
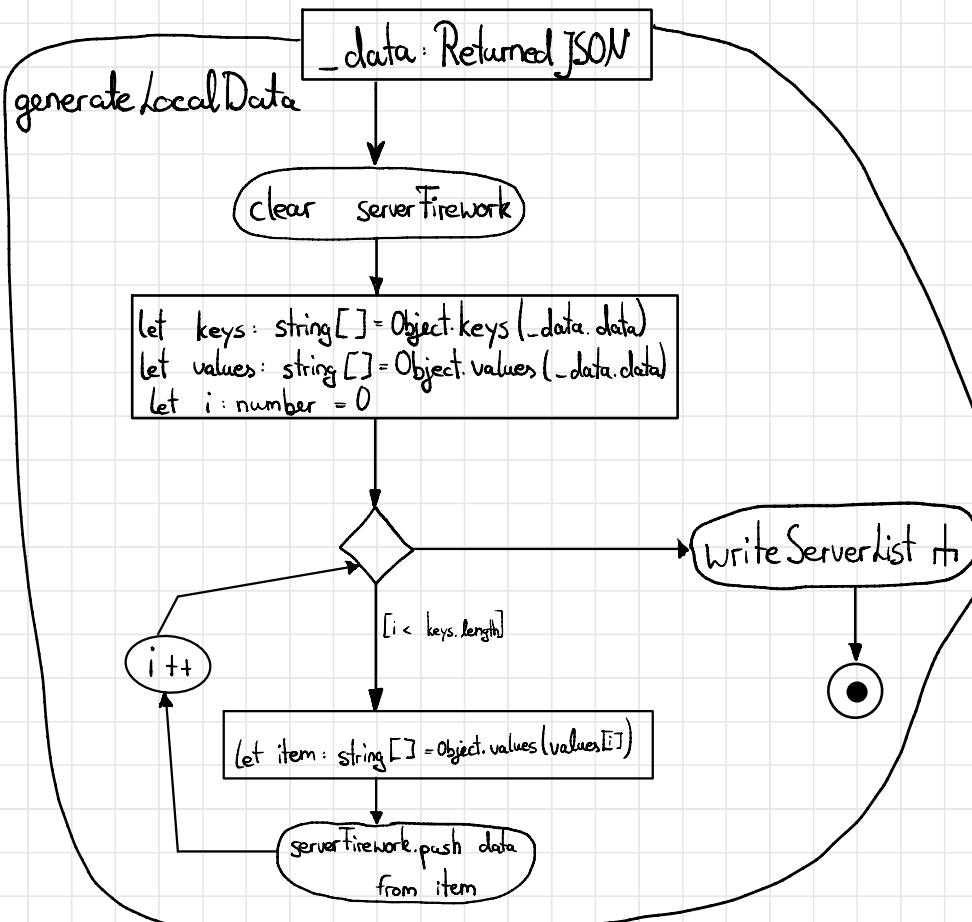
name: string;
color: string;
pattern: Pattern;
size: number;
lifespan: number;
id: string;
serverSafe: boolean;
    
```

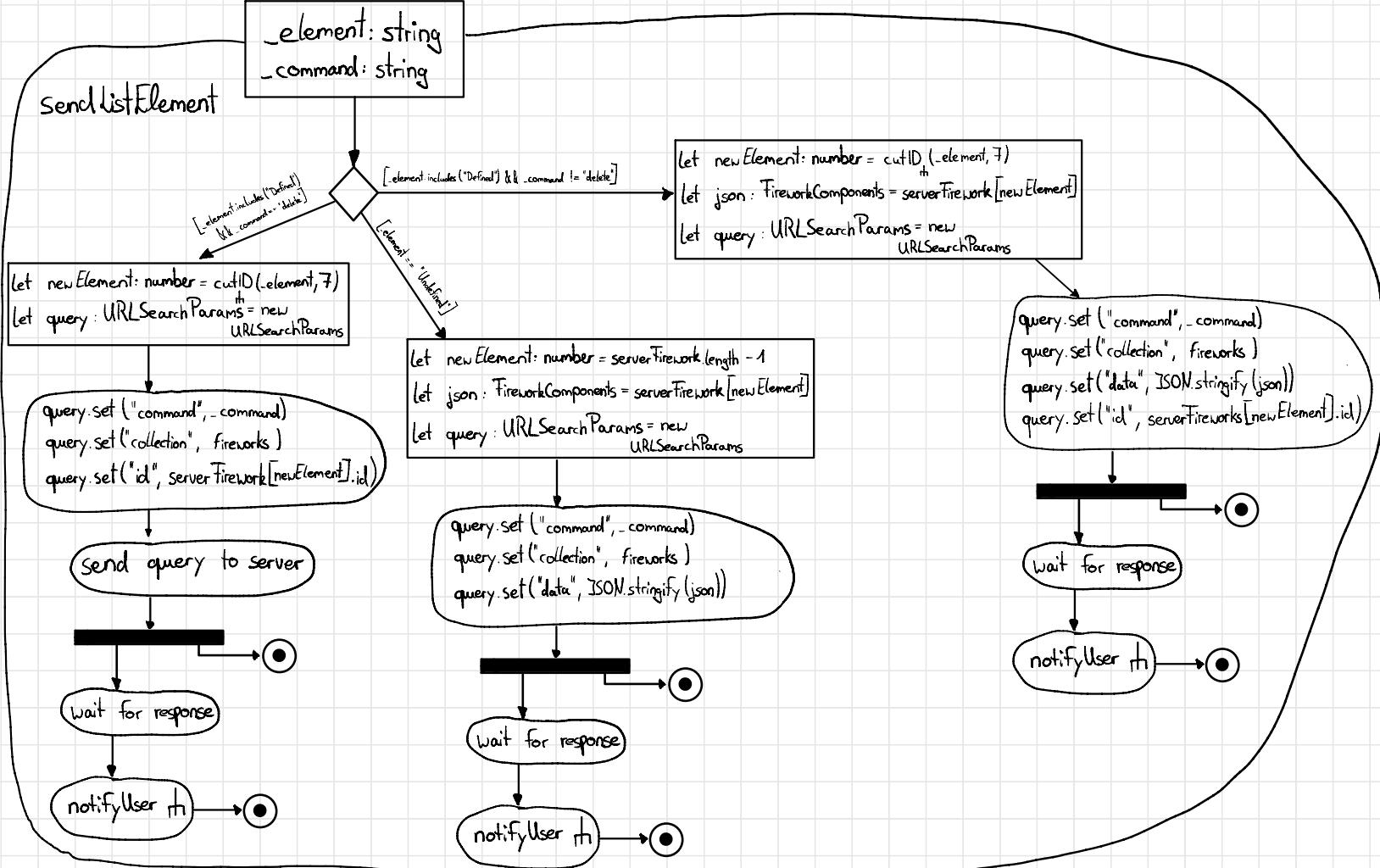
**Pattern**

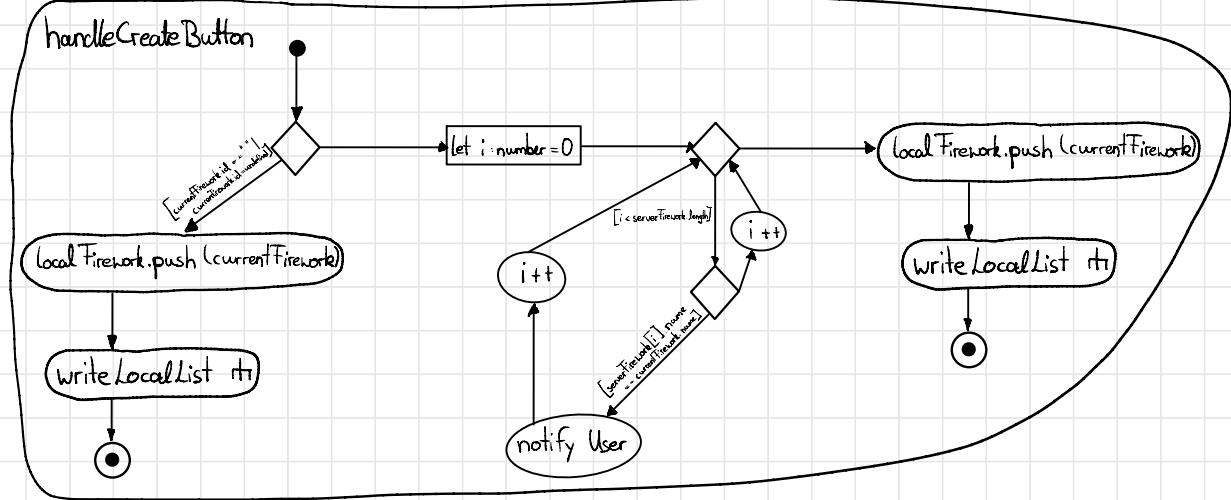
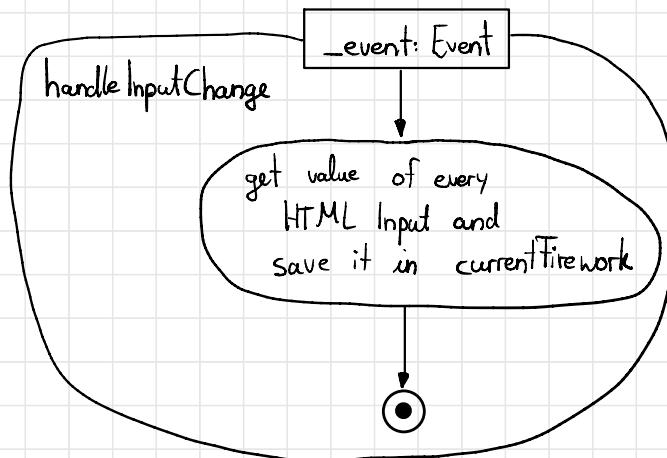
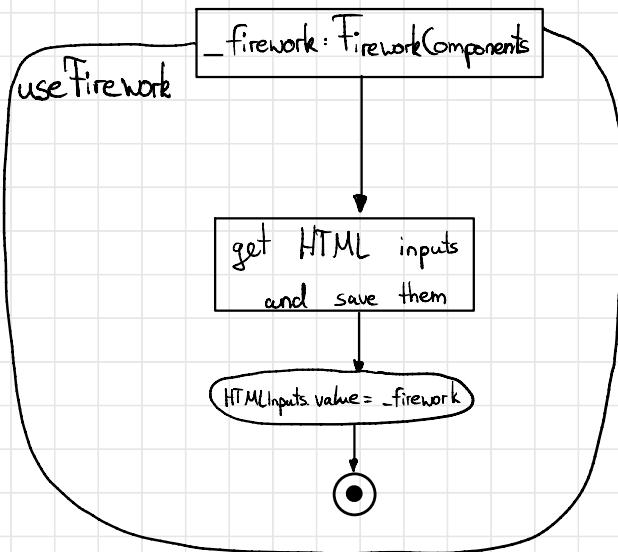
```

circle = "Circle"
star = "Star"
cross = "Cross"
    
```

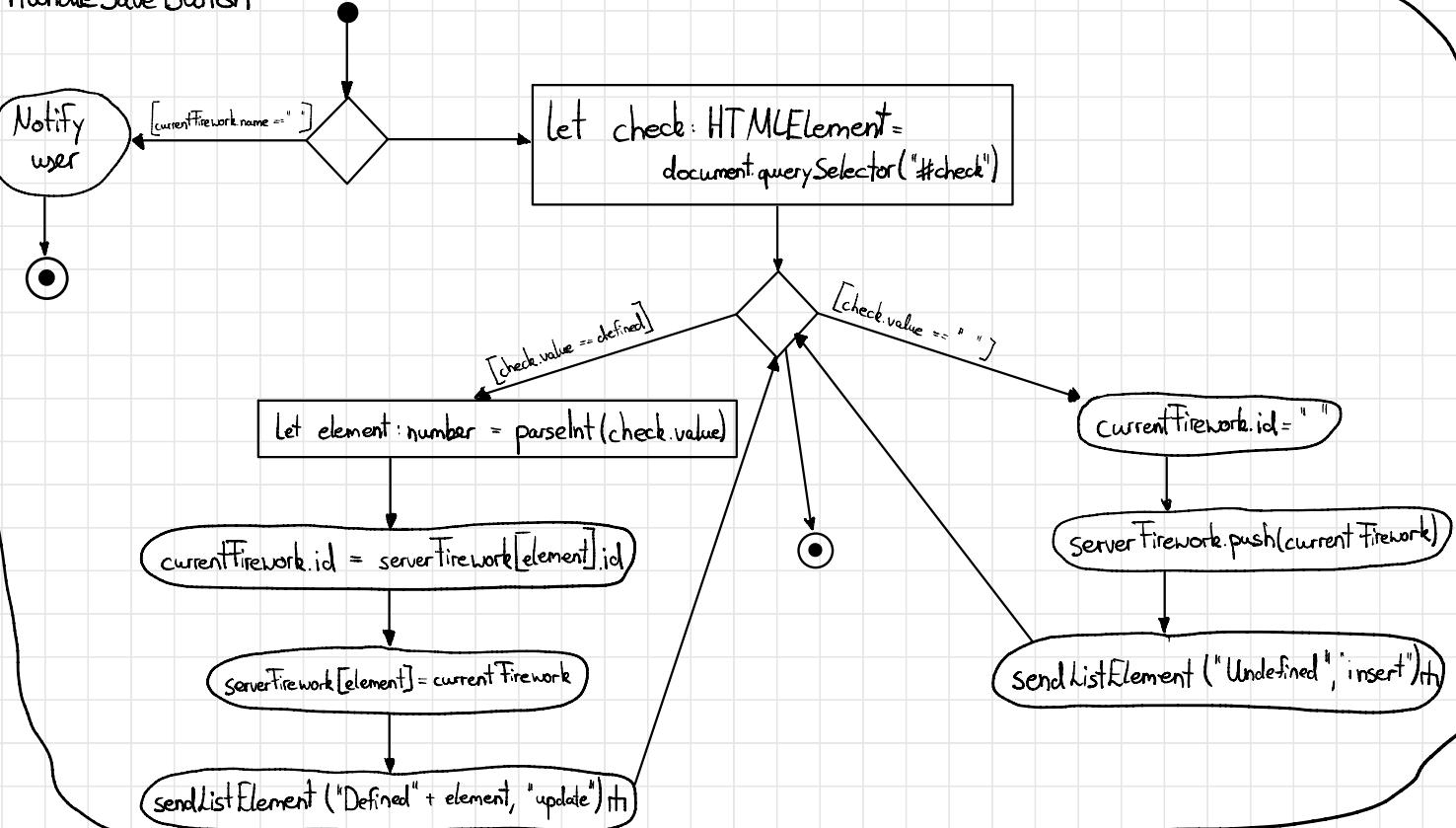


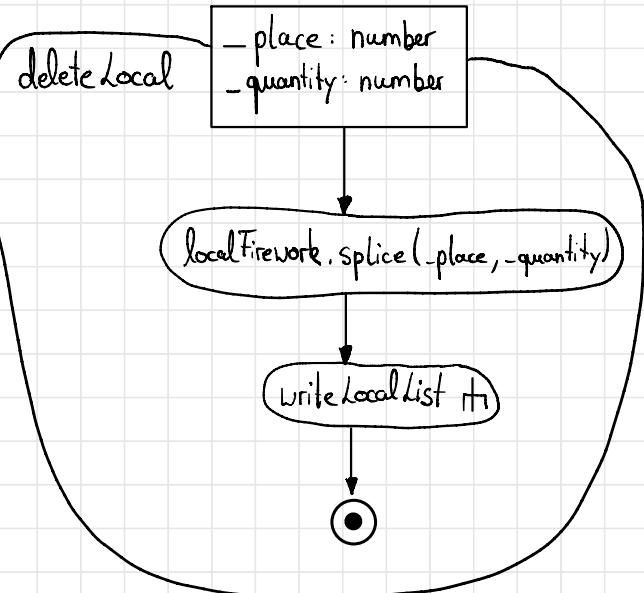
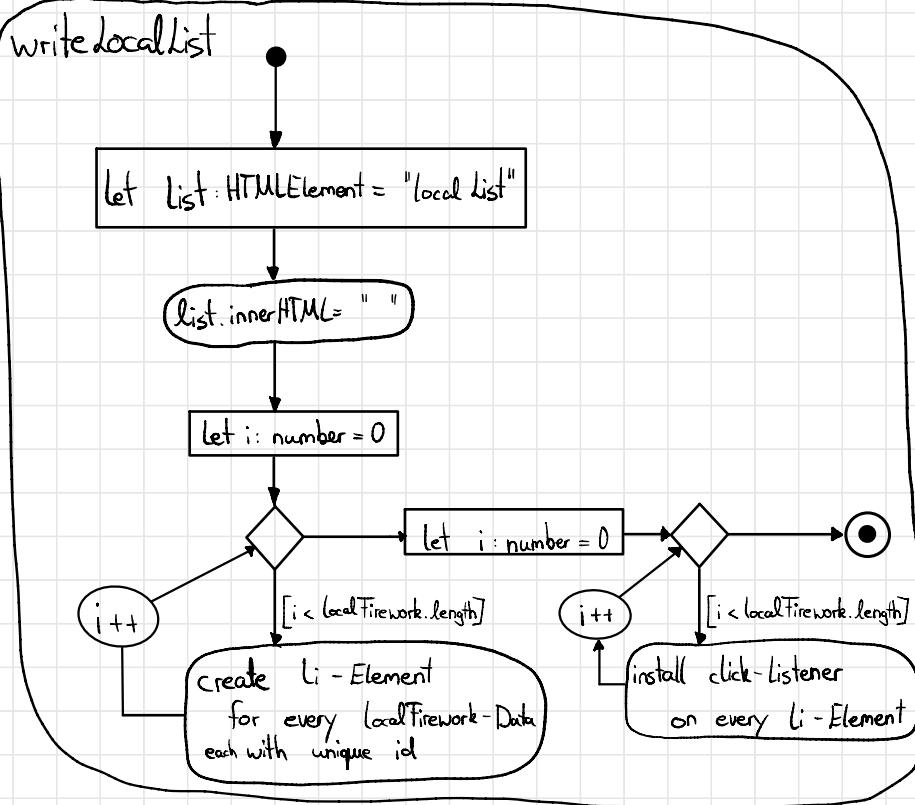






## handleSaveButton





handleClick

\_event: MouseEvent

let id: string = (\_event.target as Element).id

[id includes "local"]



[id includes "server"]  
[id includes "delete"]  
[id includes "use"]

let newID: number = cutID(id, 6)

let newID: number = cutID(id, 6)

let newID: number = cutID(id, 3)

deleteLocal(newID, 1) +

useFirework(localFirework[newID]) +

let newID: number = cutID(id, 6)

let newID: number = cutID(id, 3)

sendListElement("Defined" + newID, "delete") +

useFirework(serverFirework[newID]) +

handleCanvasClick

\_event: MouseEvent

let hotspot = new Vector  
with Event - Position

createFirework(hotspot) +

