

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

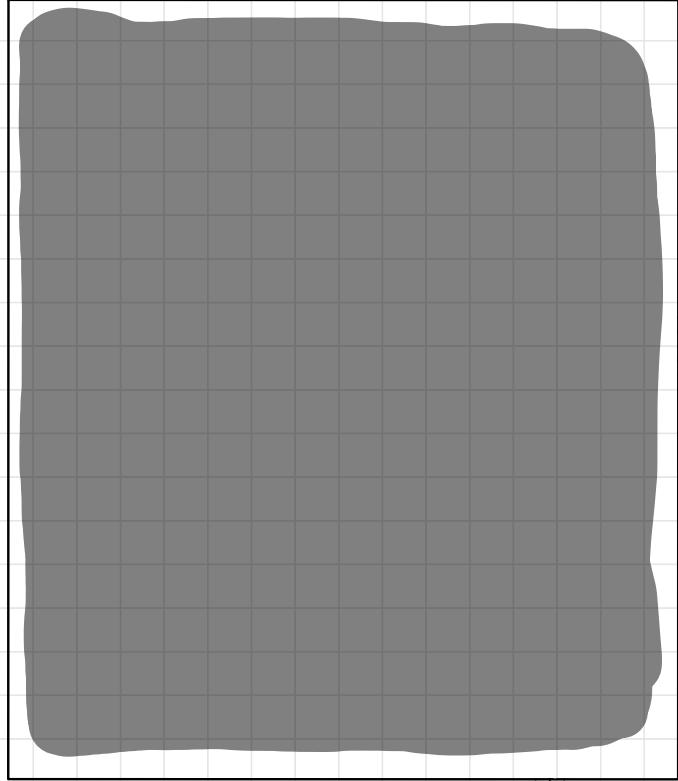
Create

Save current Firework

<div>
id = "create"
>change

with id

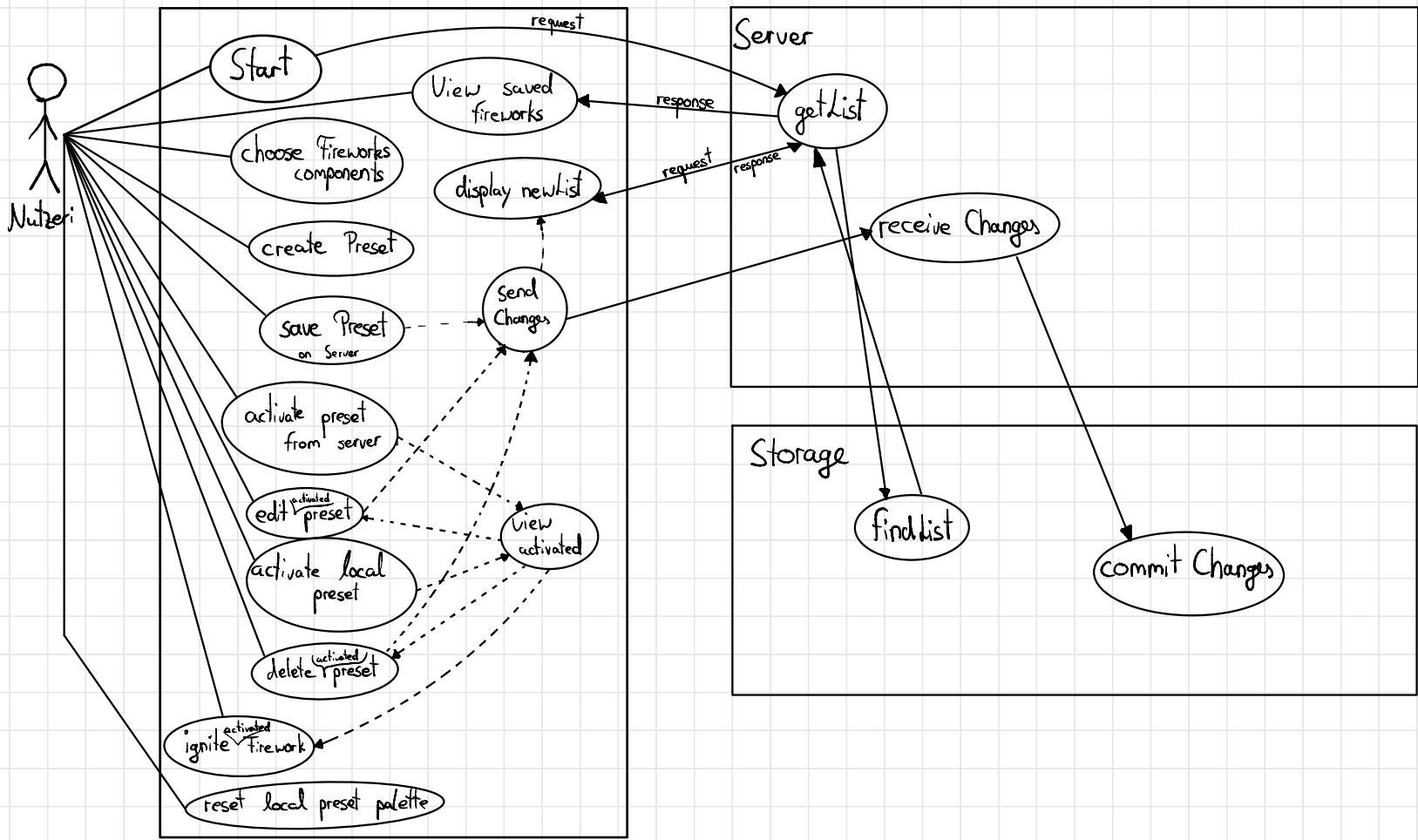
<canvas>
width = "800"
height = "55"
>click



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

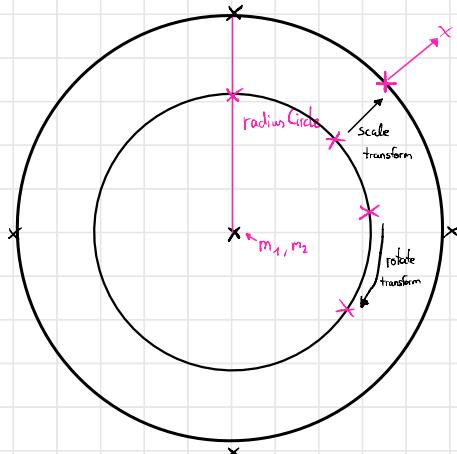
id = "listElement" + index
>click

Use-Case - Diagram



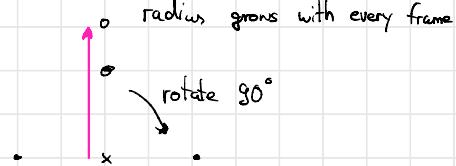
Firework: Canvas

Circle



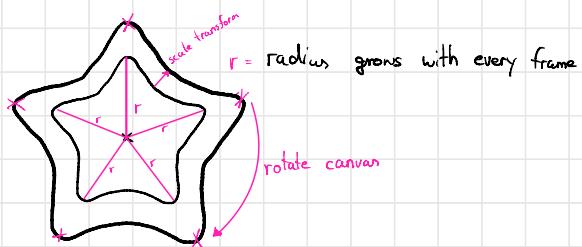
radius grows with every frame

Cross

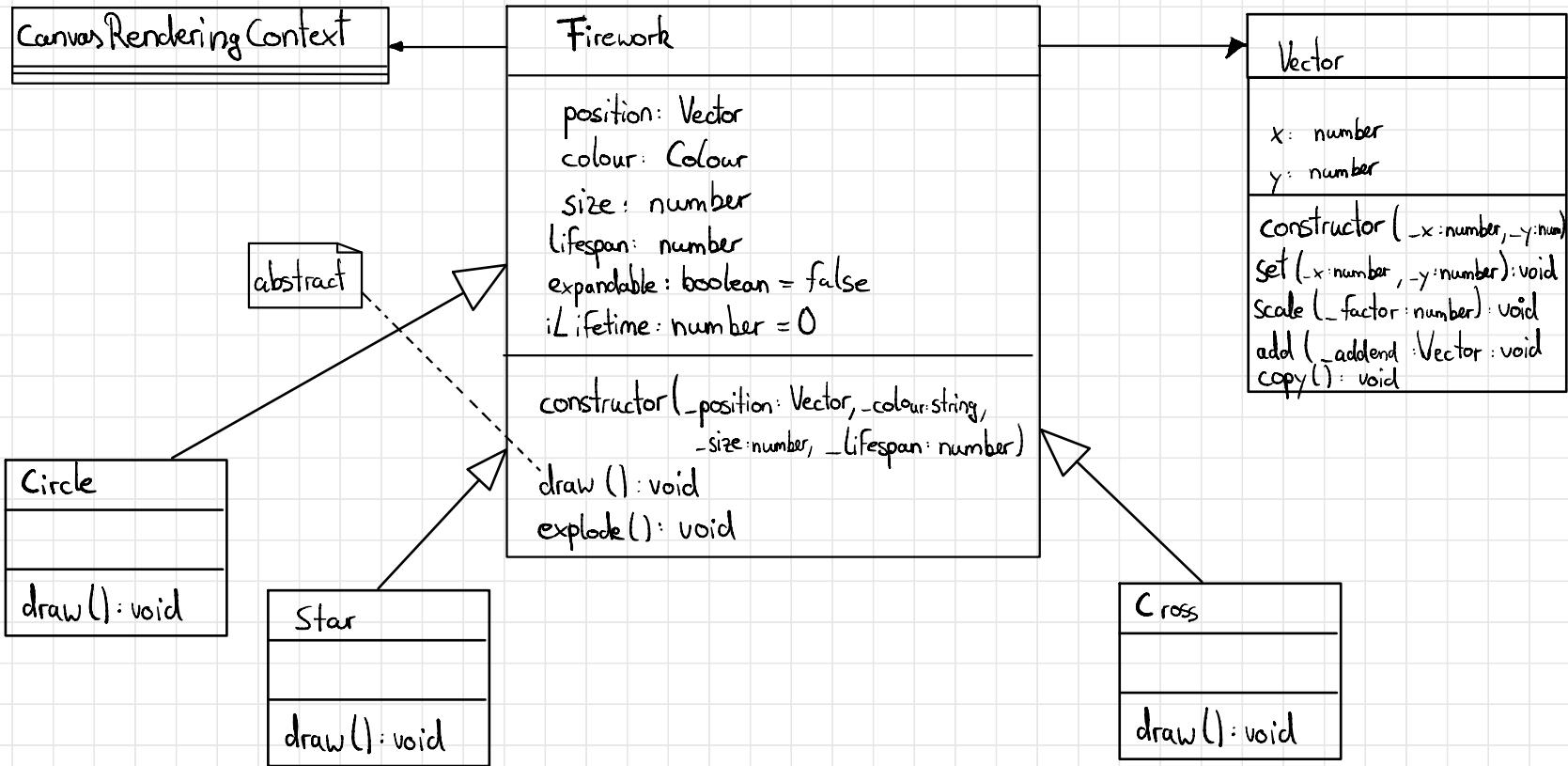


radius grows with every frame

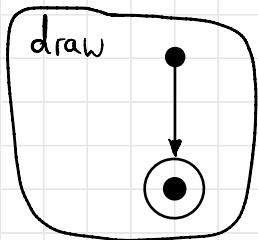
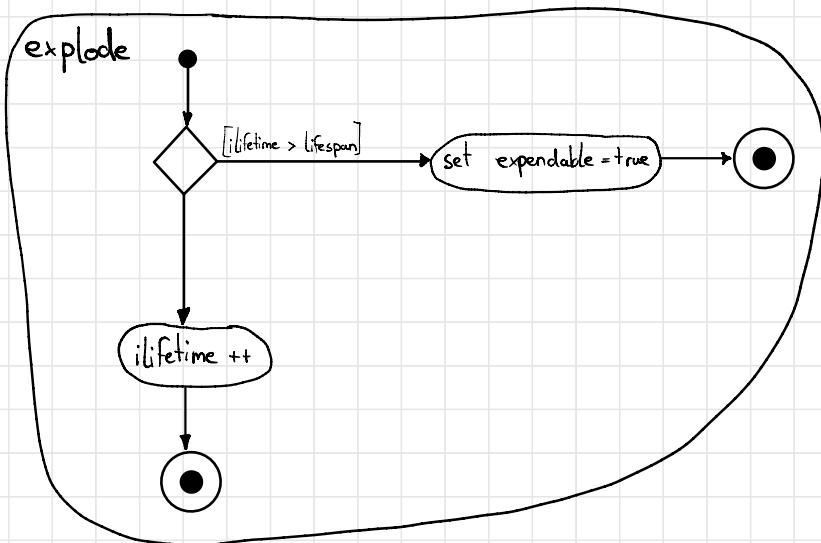
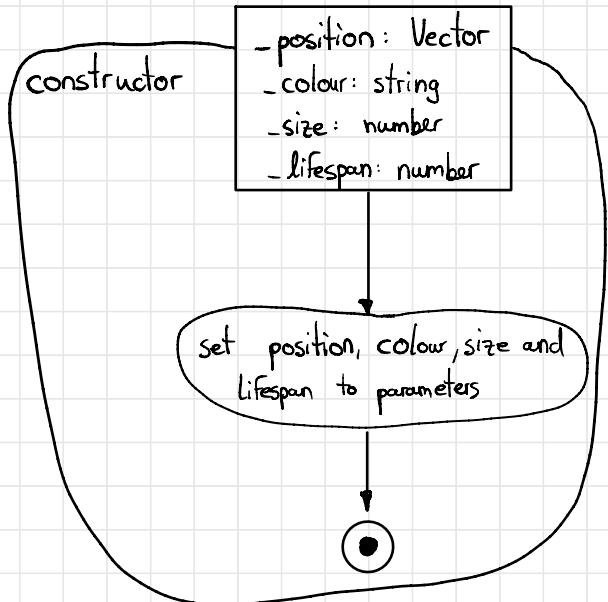
Star



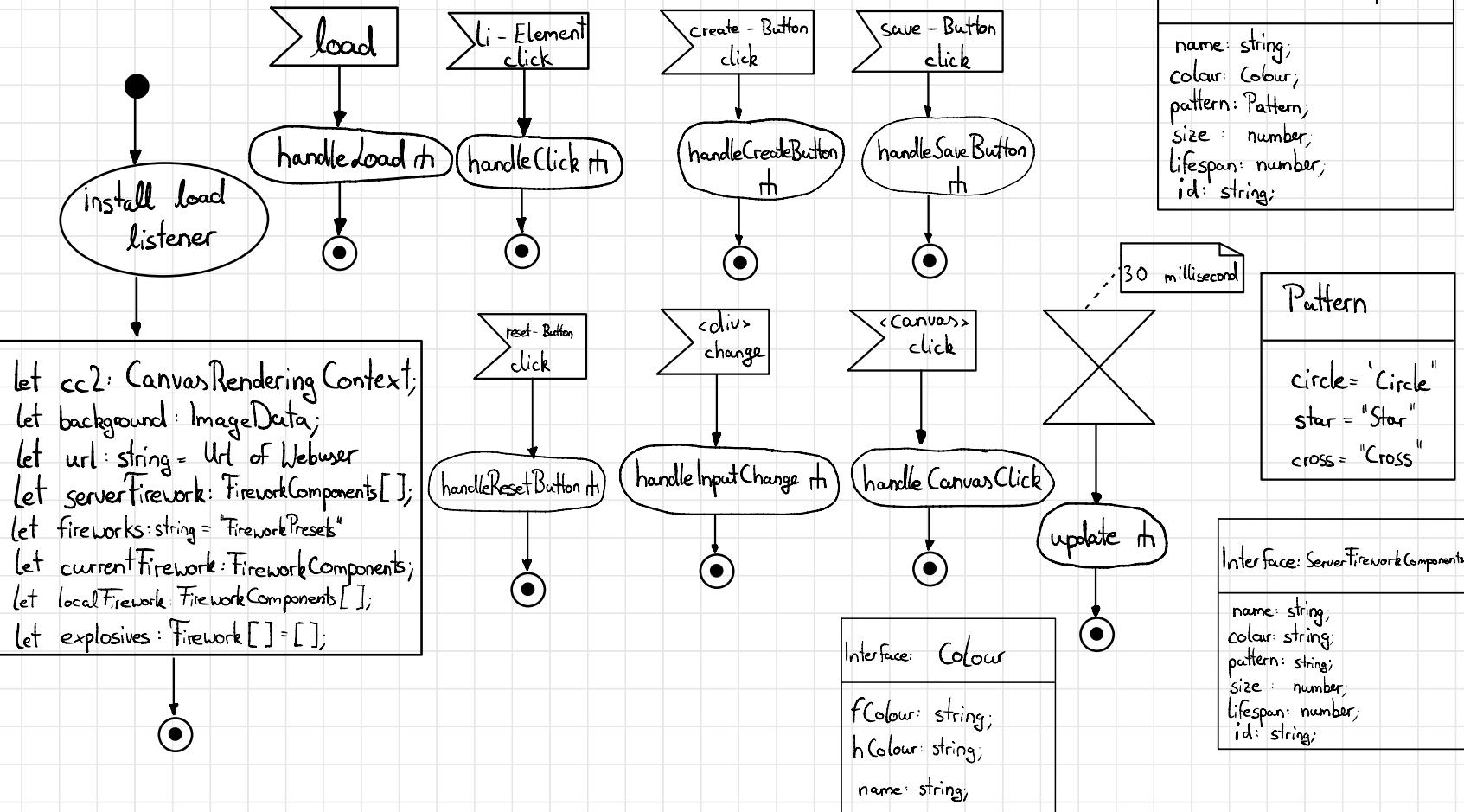
Firework: Class - Diagram

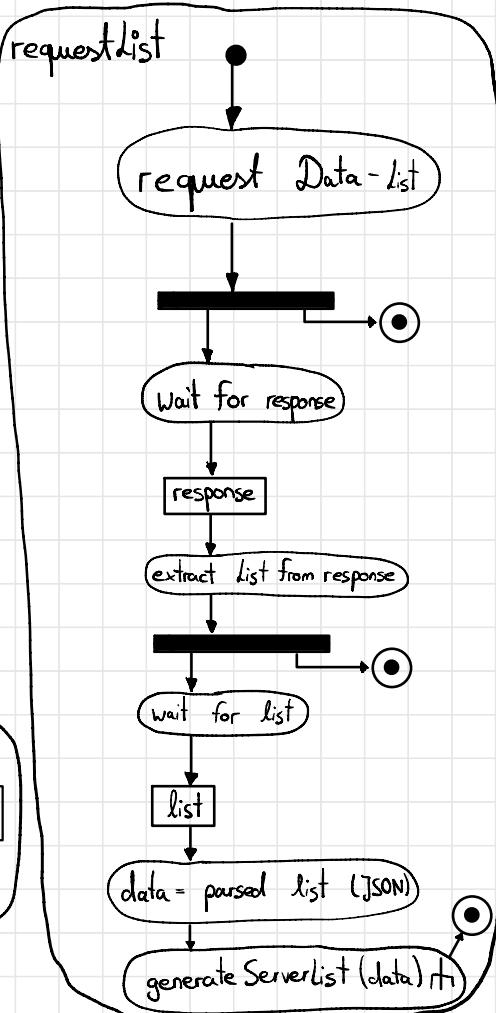
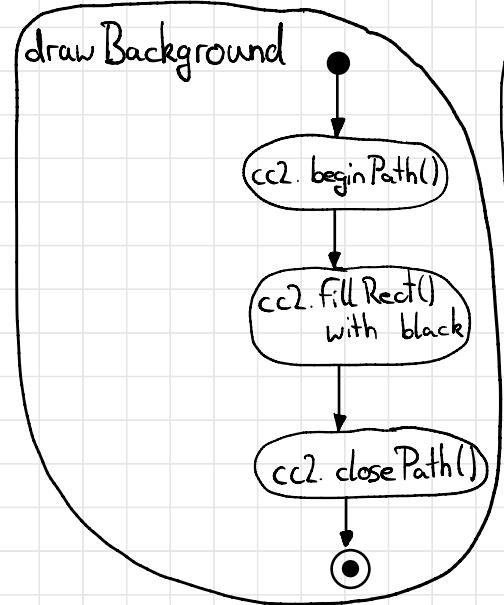
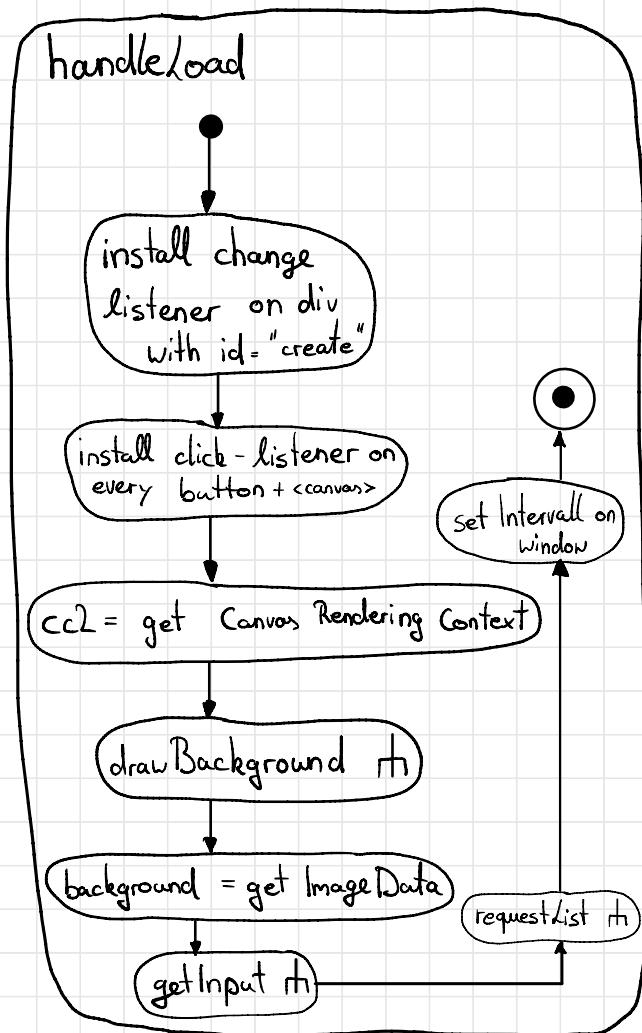


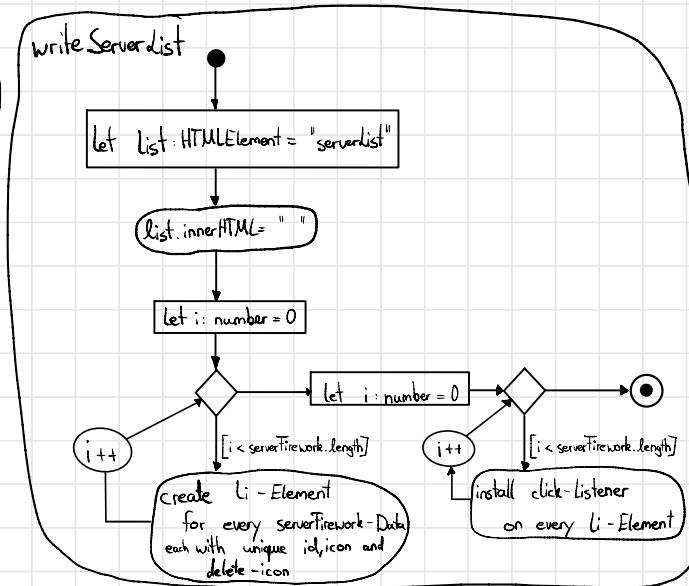
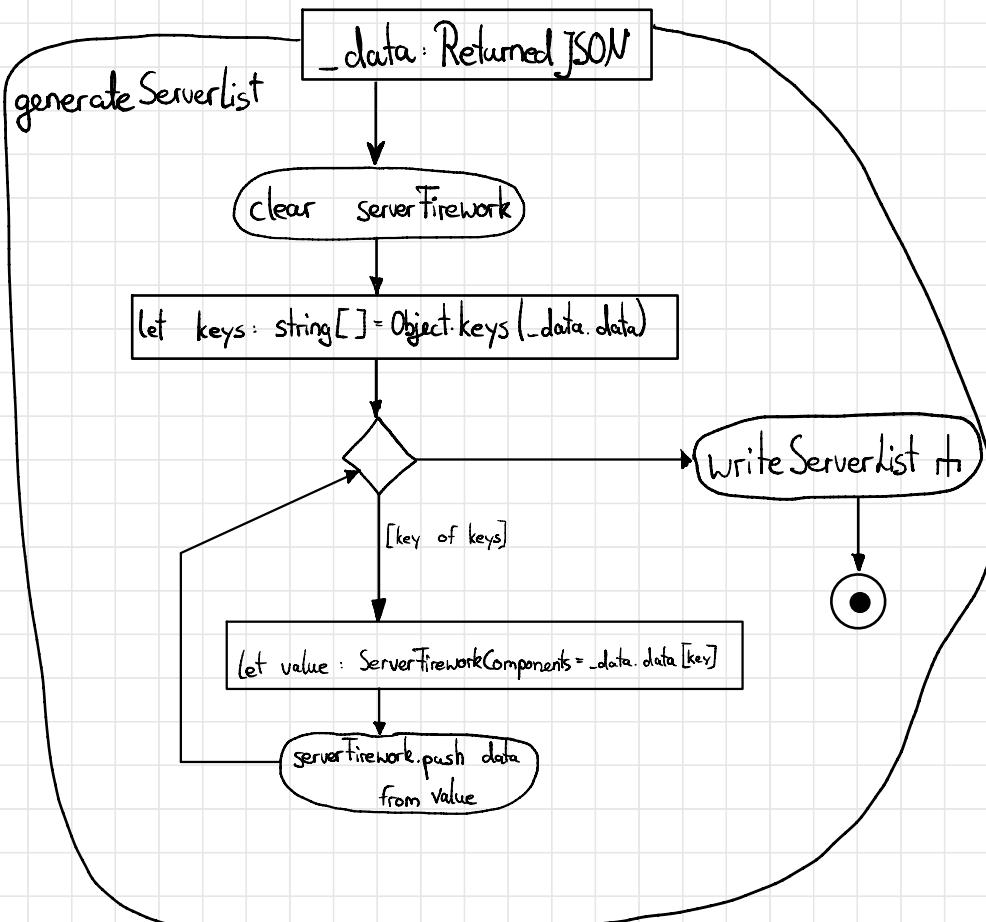
Firework : Activity Diagram - Firework

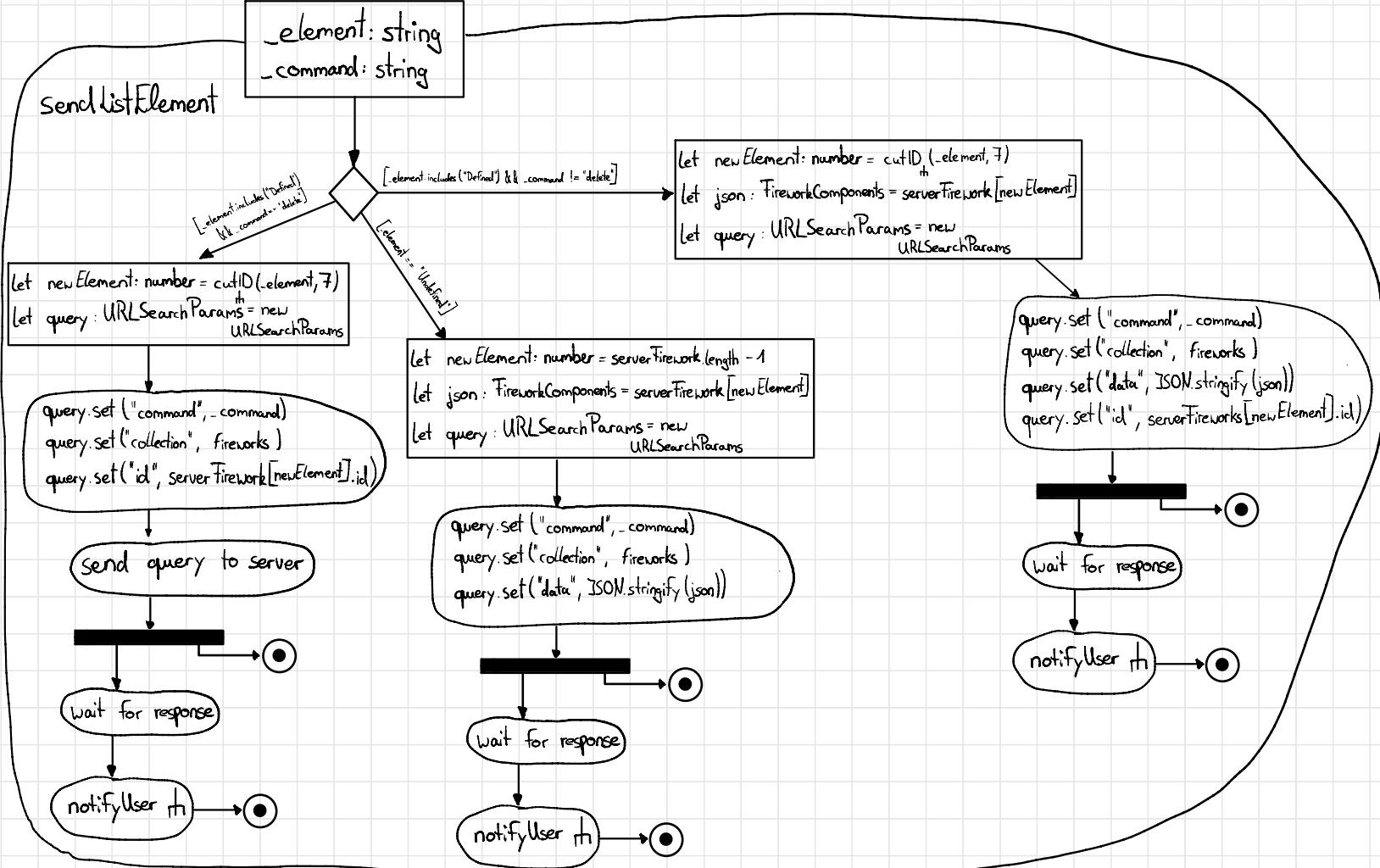


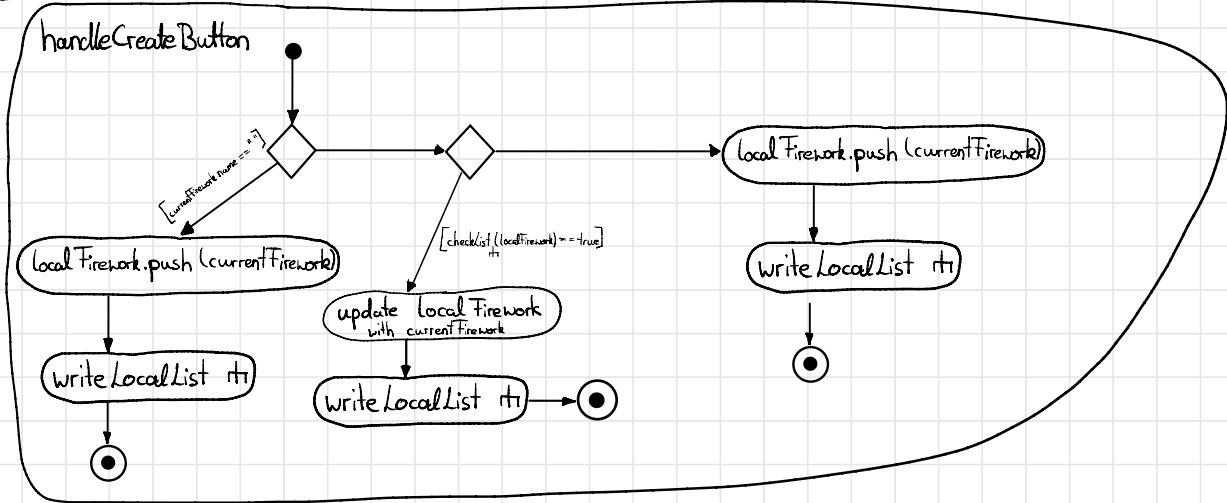
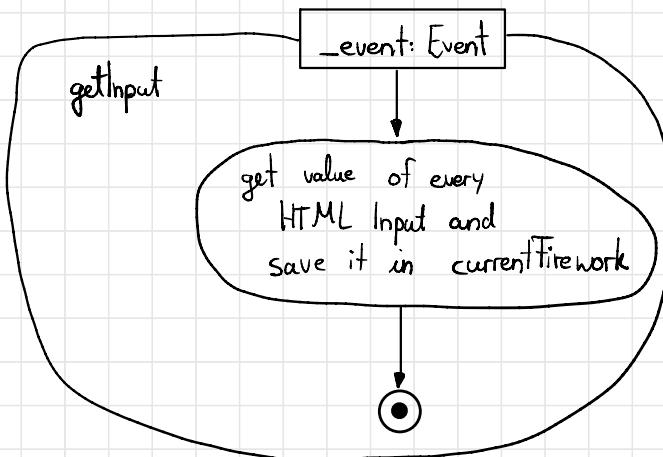
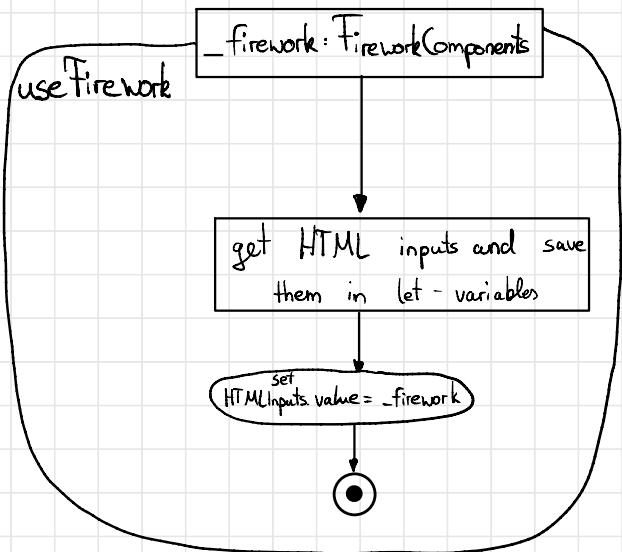
Activity Diagram: Firework



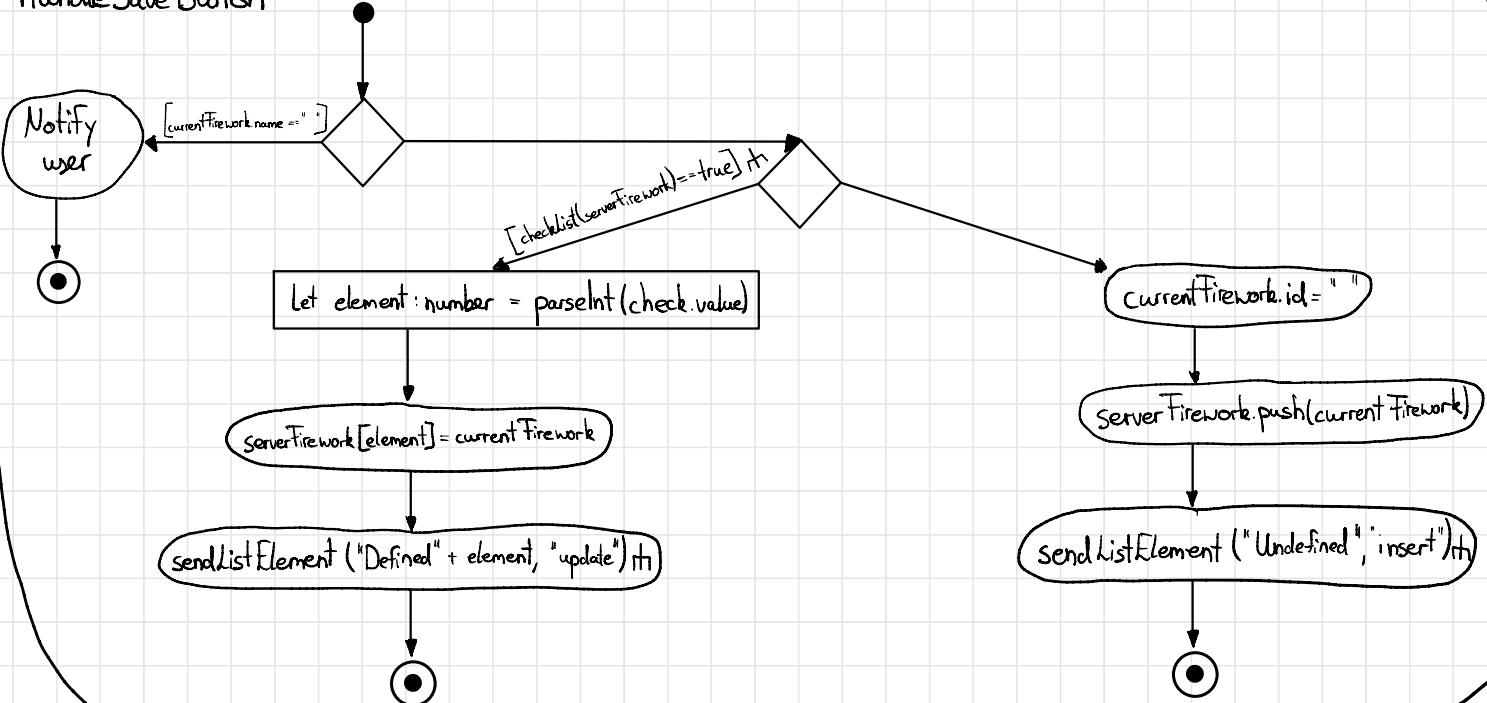








handleSaveButton



writeLocalList

```
let list: HTMLElement = "local list"
```

```
list.innerHTML = "
```

```
let i: number = 0
```

```
[i < localFirework.length]
```

create Li - Element
for every LocalFirework - Data
each with unique id!

```
i++
```

```
[i < localFirework.length]
```

install click-listener
on every Li - Element

deleteLocal

```
- place: number  
- quantity: number
```

```
localFirework.splice(-place, -quantity)
```

```
writeLocalList()
```

handleClick

_event: MouseEvent

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

[id includes "local"]



[id includes "server"]
[id includes "sensor"]
[id includes "delete"]

let newID: number = cutID(id, 6)

let newID: number = cutID(id, 6)

let newID: number = cutID(id, 3)

deleteLocal(newID, 1) +

useFirework((localFirework[newID]) +)



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

useFirework(serverFirework[newID]) +

handleCanvasClick

_event: MouseEvent

let hotspot = new Vector
with Event - Position

Create Firework out
of currentFirework
on hotspot



