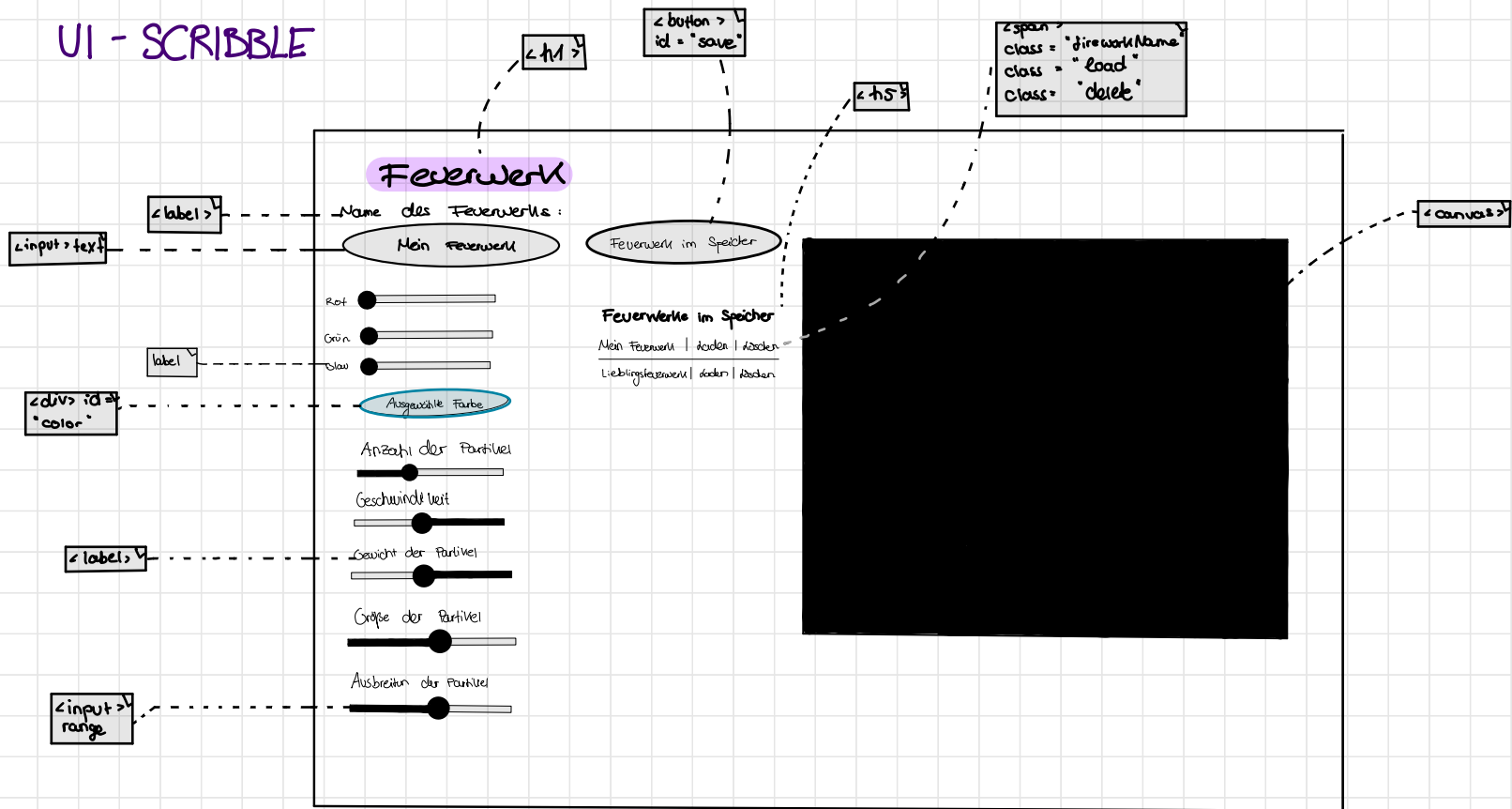
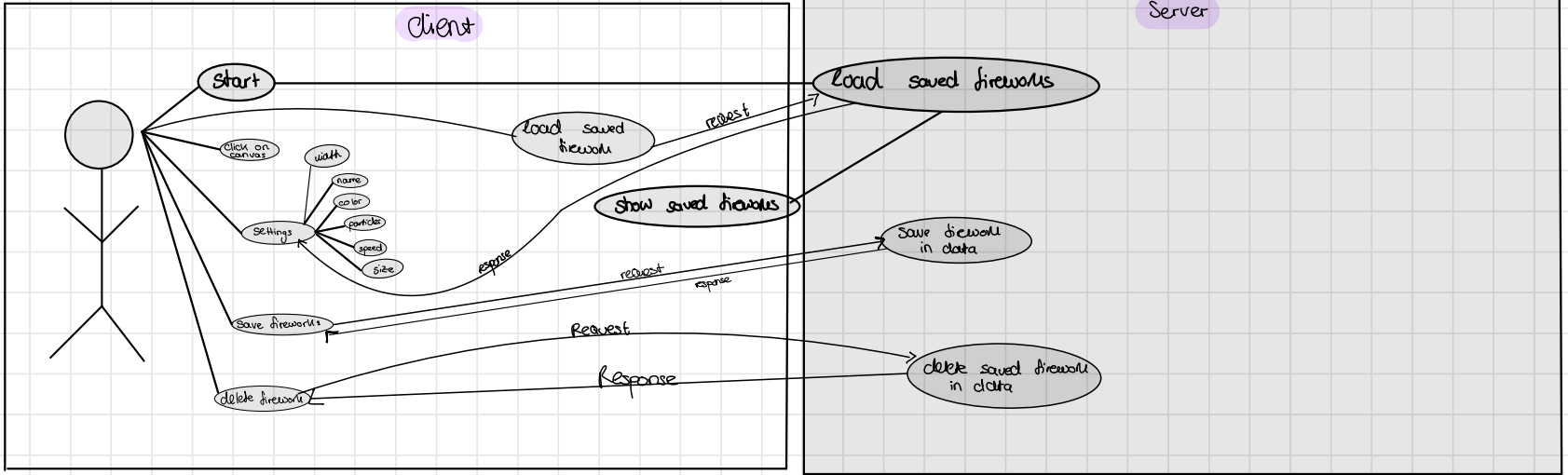


UI - SCRIBBLE



USE-CASE



CLASS-DIAGRAM

Firework

```
particles : Particle[];  
CreatedParticles : boolean;  
particleConfig : ParticleConfig;  
counter : number;  
ctx2 : CanvasRenderingContext2D;  
color : string;  
numberOfParticles : number;  
position : Vector;  
speed : number;
```

```
Constructor (config : FireworkConfig, particleConfig : ParticleConfig)
```

```
draw () : void
```

```
update () : void
```

Vector

```
x : number
```

```
y : number
```

```
Constructor (x : number, y : number)
```

```
set (x : number, y : number)
```

```
scale (factor : number)
```

```
add (addend : Vector)
```

```
random (minlength : number, maxlength : number)
```

```
copy () : Vector
```

Particle

```
alive : boolean;
```

```
lifetime : number = 20;
```

```
position : Vector;
```

```
velocity : Vector;
```

```
ctx2 : CanvasRenderingContext2D;
```

```
color : string;
```

```
downForce : number;
```

```
size : number;
```

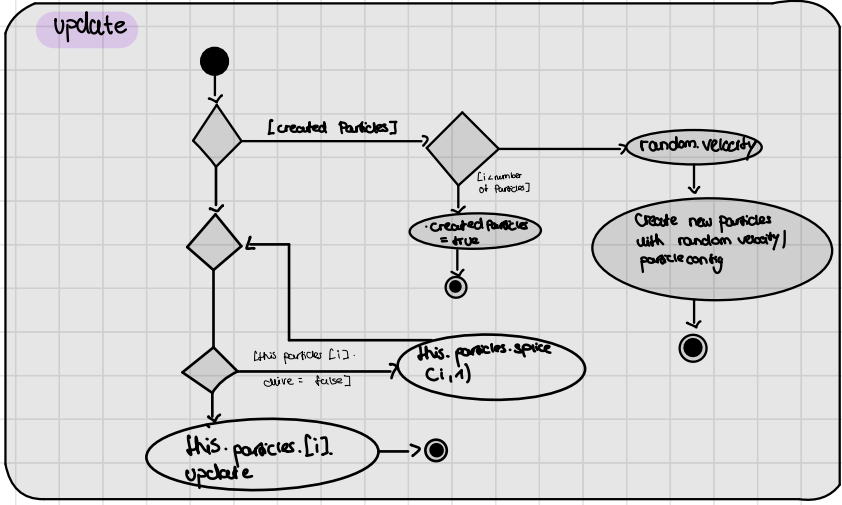
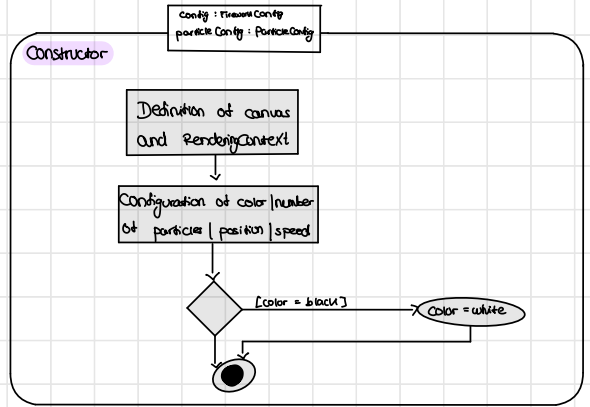
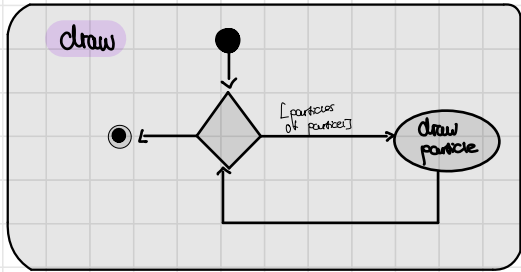
```
Constructor (config : ParticleConfig, position : Vector,  
            startVelocity : Vector)
```

```
update () : void
```

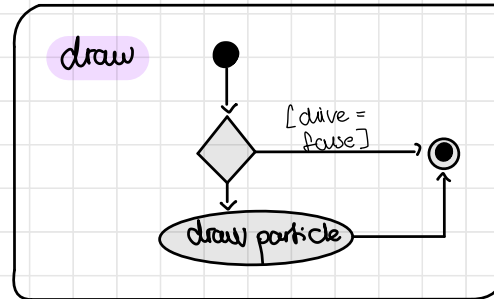
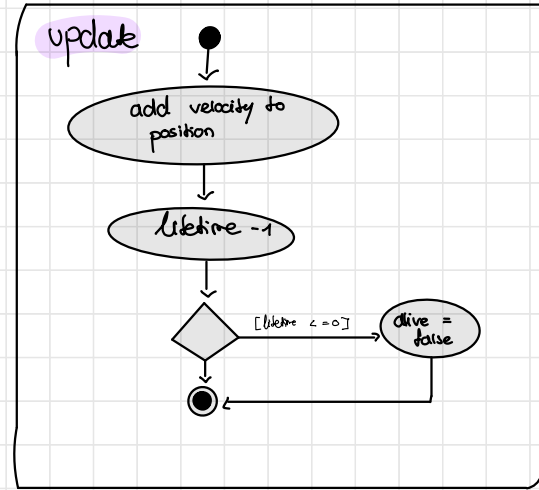
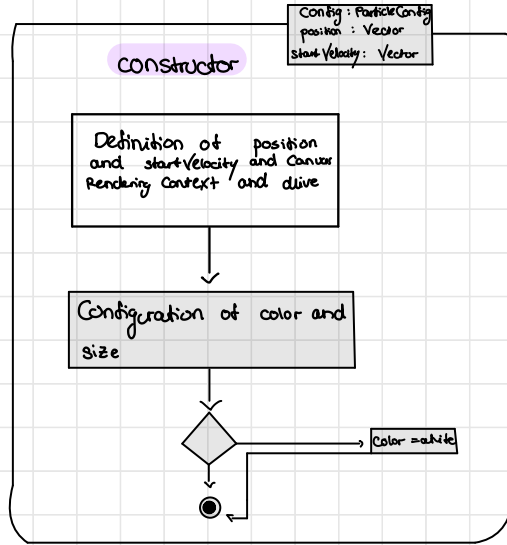
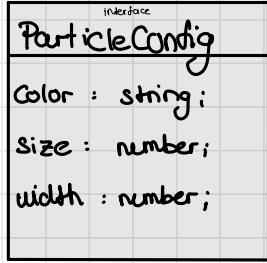
```
draw () : void
```

ACTIVITY - DIAGRAM

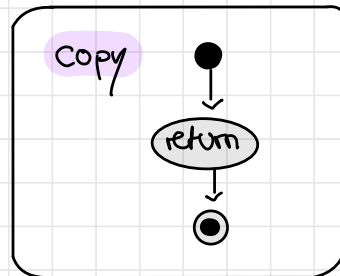
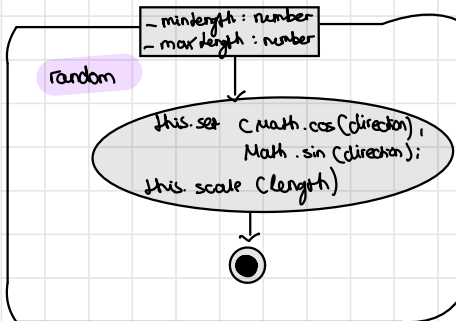
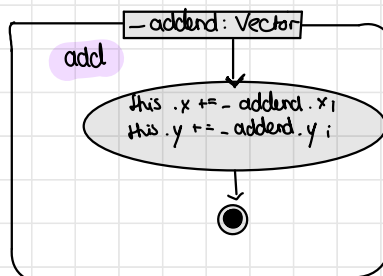
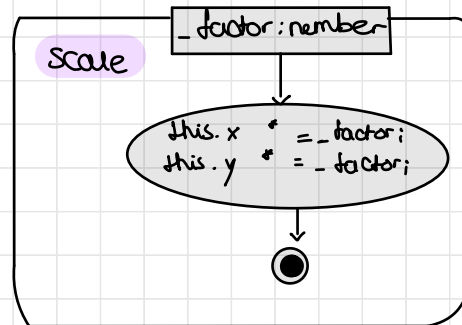
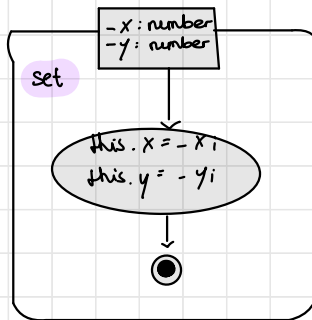
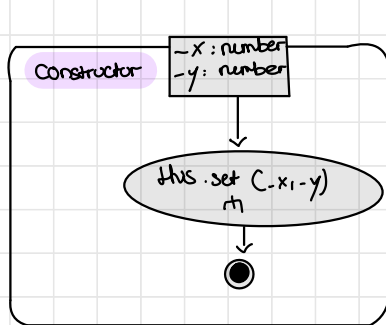
- Firework



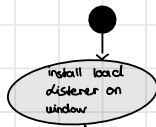
- Particle



~ vector



~ main



```

let canvas = HTMLCanvasElement
let ctx = canvas.getContext(2d)
let fireworks = fireworks [] = []
let rect = DOMRect;

let name: string = "Main fireworks"
let color: string = "#ff0000"
let red: number = 255;
let green: number = 0;
let blue: number = 0;

let numberOfParticles: number = 50;
let speed: number = 5;
let size: number = 1;
let width: number = 10;

+
and define input-Elements
  
```

