

EIA2

ENDABGABE 20/21

Name: Anna Sophia Wagner
Matrikelnummer: 263690

#Feuerwerk

konzipiere & produziere eine Anwendung, mit der die Nutzer Raketenladungen zusammenstellen & per Mausklick oder Touch zur Explosion bringen können.

- Nutzer: kann verändern & testen
- Kreationen auf Server speicherbar & wieder abrufbar
- Endergebnis: per Touch / Mausklick erstelltes Feuerwerk interaktiv & live auf Screen zu inszenieren

Funktionsanalyse

Nutzererlebnis

Nutzer soll sich durch spielerische & interaktive Art ein Feuerwerk zusammenstellen können.

- Spaß / Freude
- Spielerie → viele Möglichkeiten zum Austesten
- Einstellungen behalten / Speichern
(selbst erstelltes Feuerwerk → Stolz)
- Weiterleiten an Freunde & Familie
→ gemeinsame Freude & Zeit

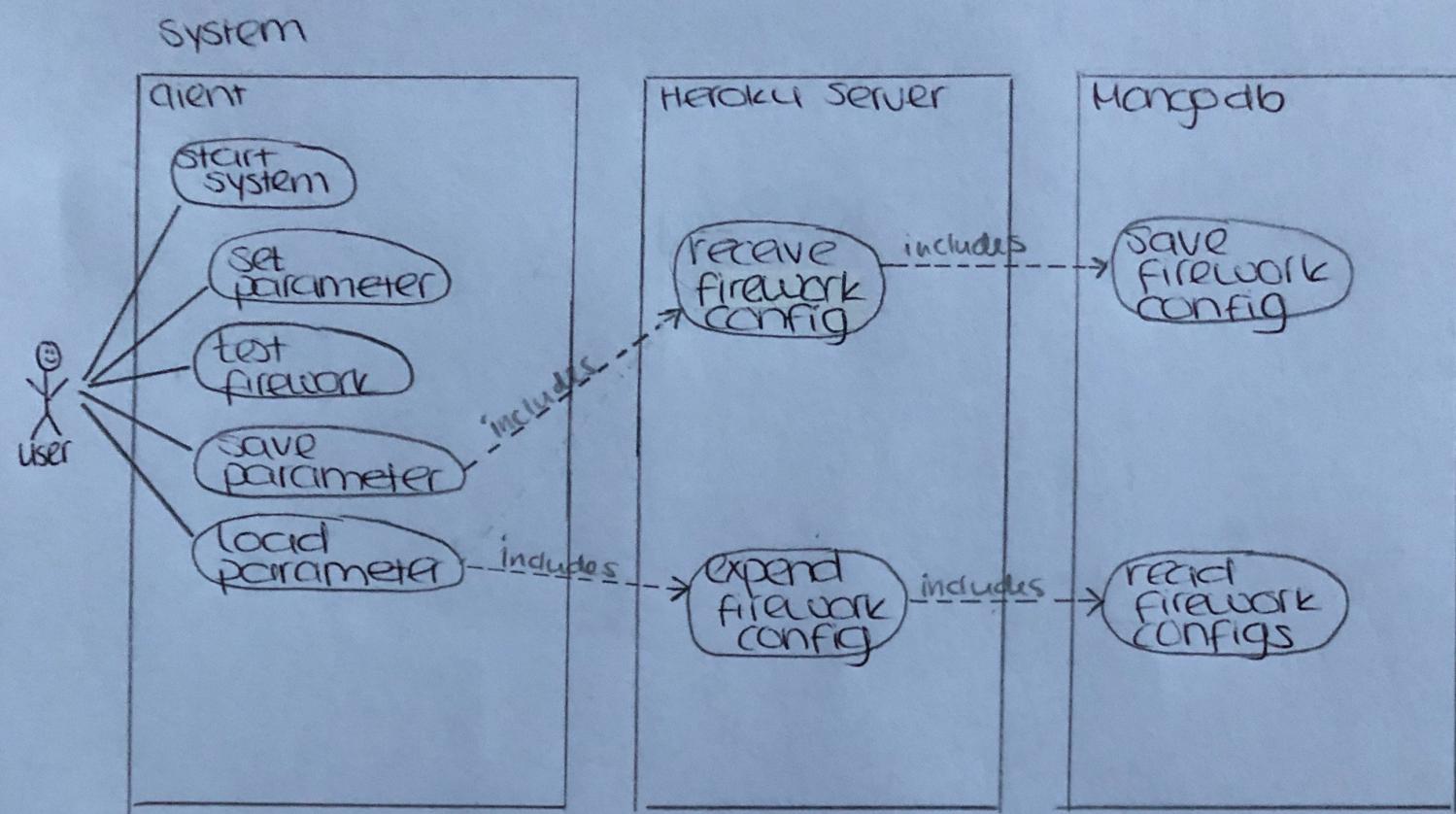
Akteure

- Nutzer / Client
- Heroku Server
- MongoDB Server

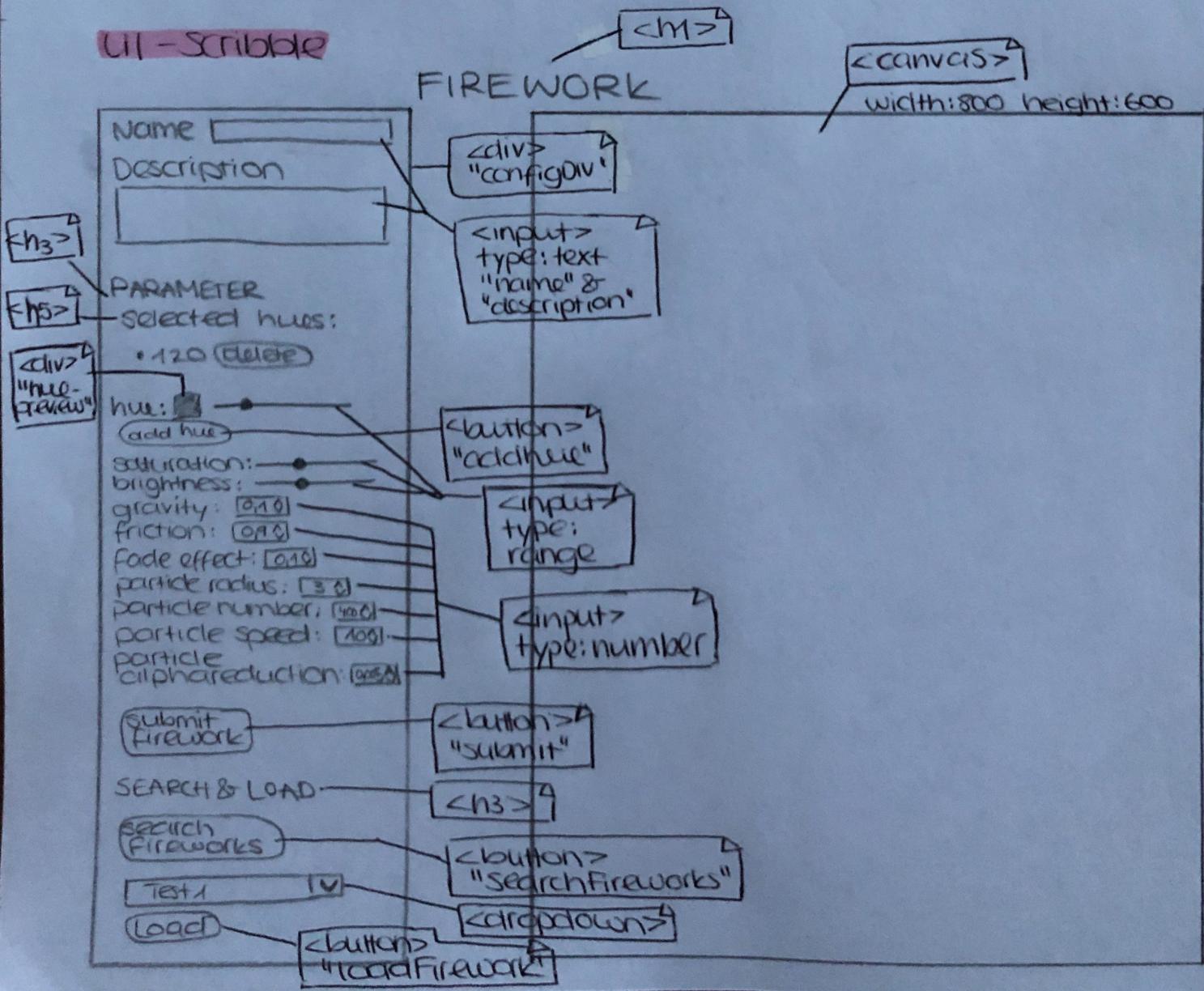
Plattform

Die Anwendung soll für PC's produziert werden, da man einen größeren Bildschirm hat (als bei Smartphones) und somit ein besseren Ersatz für ein echtes Feuerwerk als Nutzer erlebt. Zudem erreicht man so auch Nutzer der älteren Generationen.

use-case - dicigramm

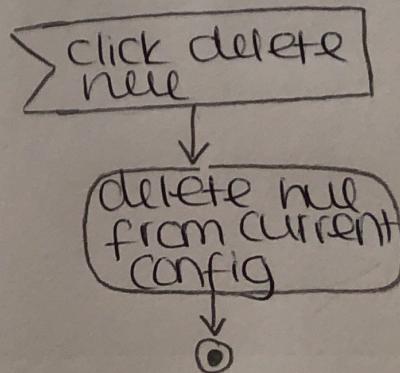
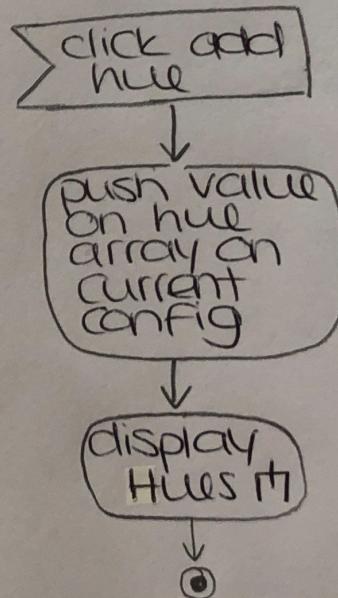
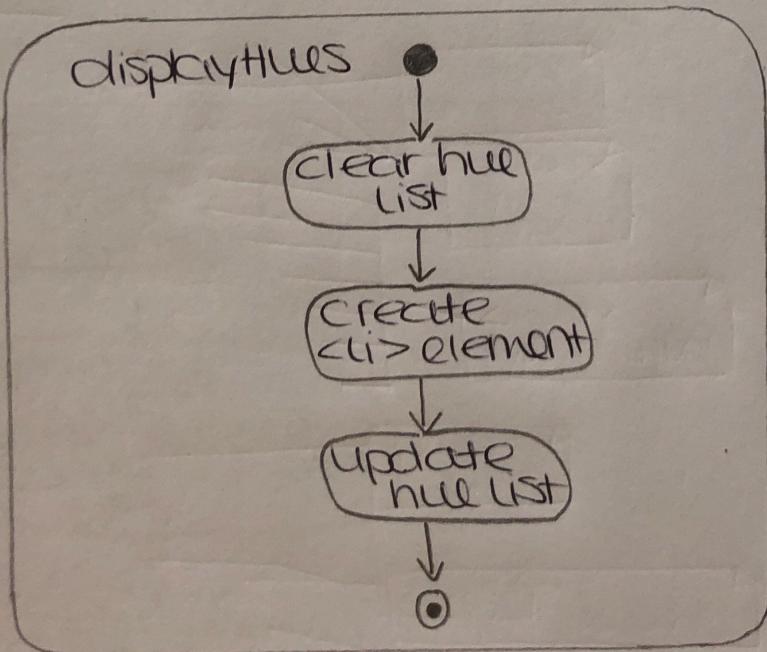
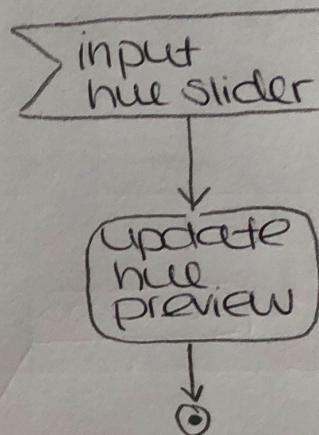
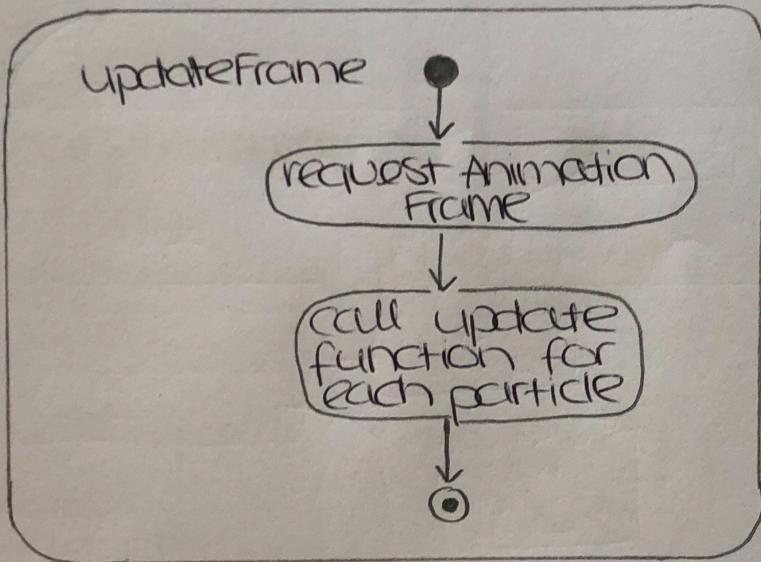
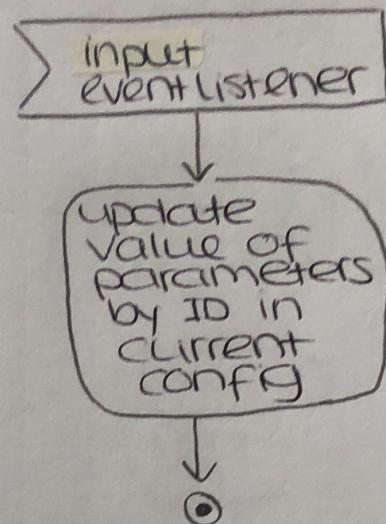
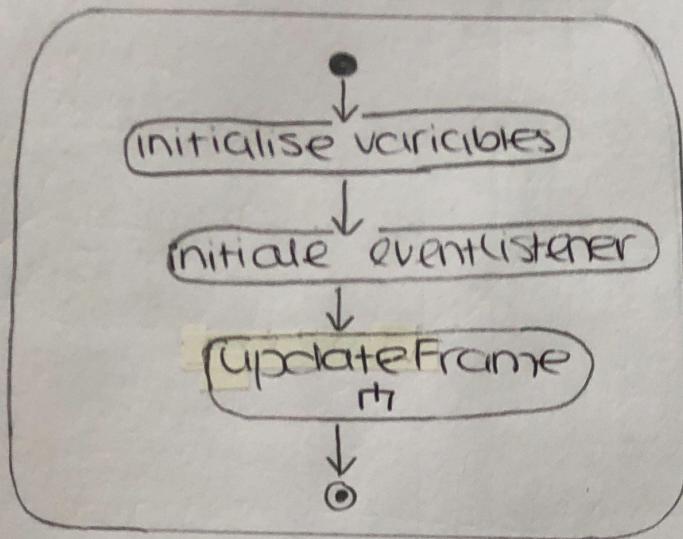


UI - Scribble



activity diagram

main



➤ Click submit button

Send current config as JS to server (Heroku)

➤ Click search firework button

Fetch configs from server

Add configs to available configs

Update dropdown

➤ Click load button

Update current config to selected config

Load current config

Load current config

Update UI controls to current config

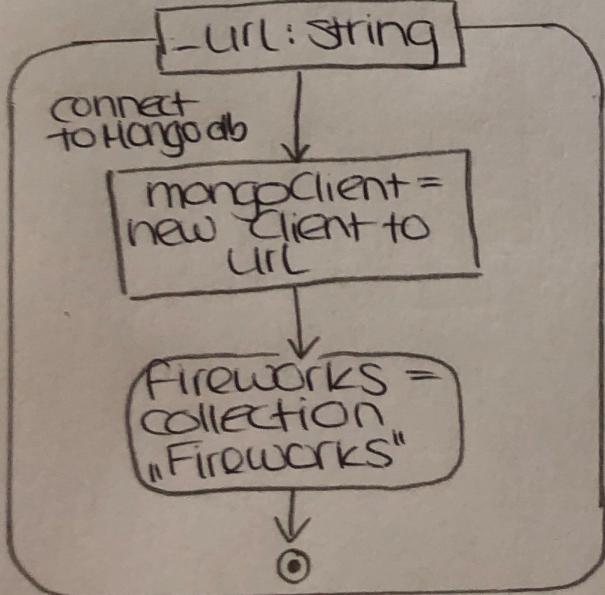
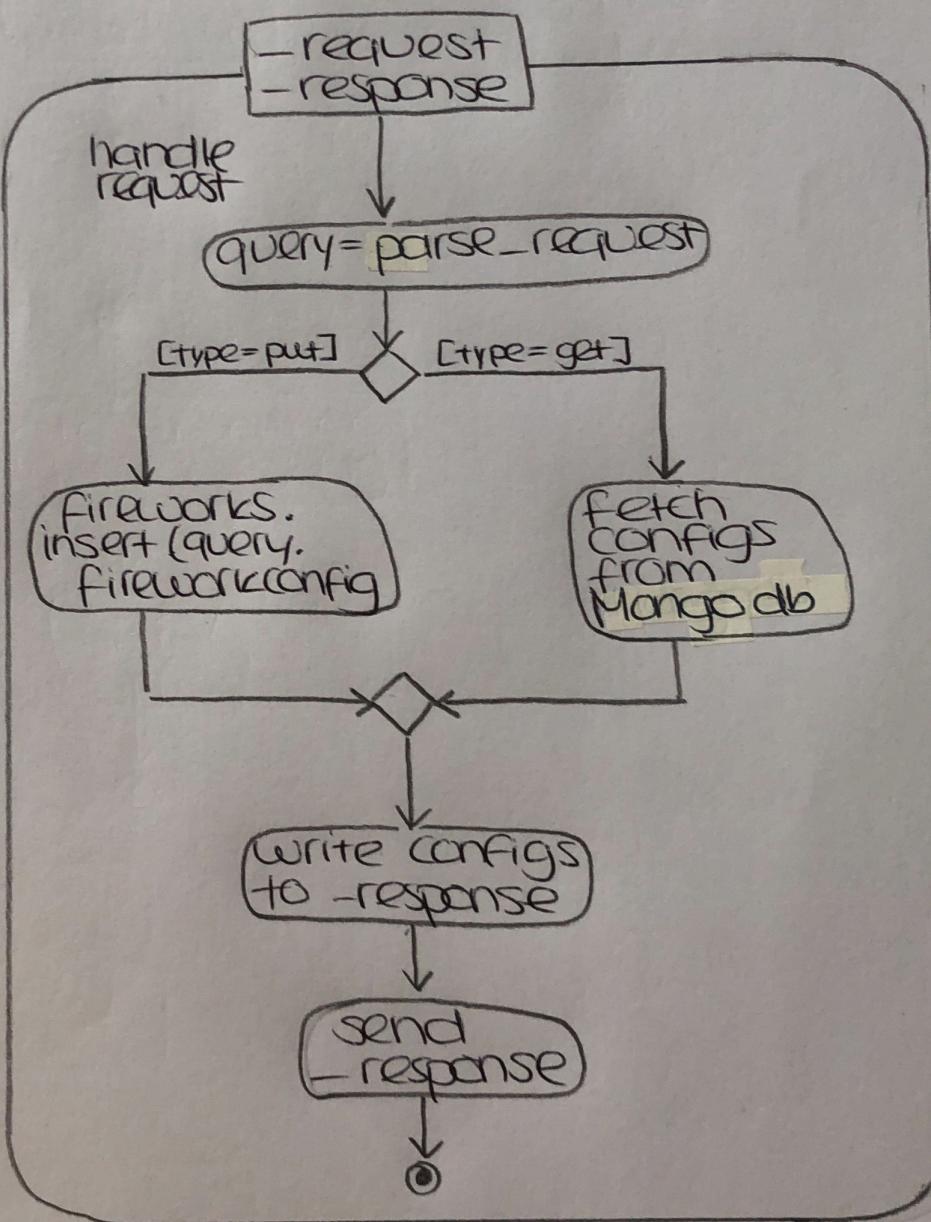
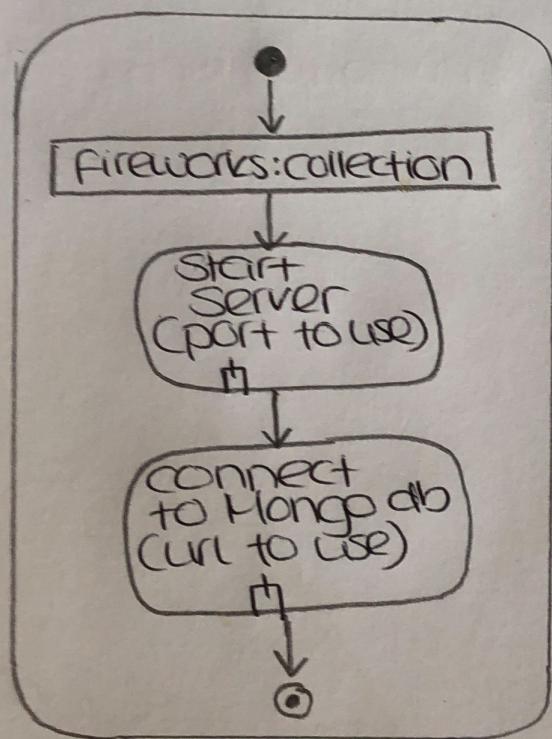
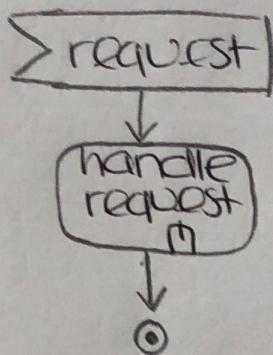
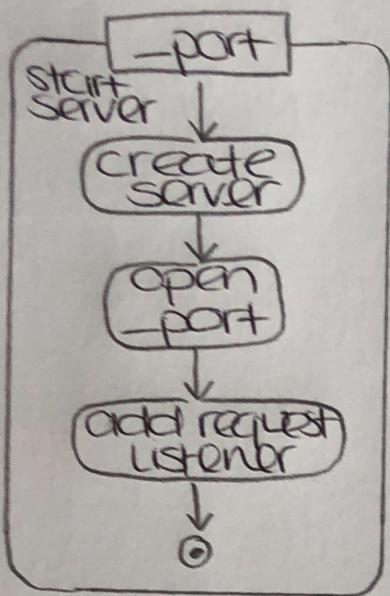
Display hues

➤ Click canvas

Generate particles with current config parameters

Draw particle with current config parameter

Server



class diagram

export class Particle

```
public alpha: number;  
private pos: Vector;  
private radius: number;  
private color: string;  
private velocity: Vector;  
private gravity: number;  
private friction: number;  
private alphaReduction: number;
```

```
constructor(pos, radius,  
color, velocity, gravity,  
friction, alphaReduction)
```

```
public draw();  
public update();
```

canvas rendering context

export class FireworkRocket

```
private pos: Vector;  
private config: FireworkConfig;  
private particles: Particle[] = [];
```

```
constructor(pos, config);
```

```
public update();
```

```
public isBurnedOut();
```

export type FireworkConfig

```
- id?: string;  
name: string;  
description: string;  
hues: number[];  
saturation: number;  
brightness: number;  
gravity: number;  
friction: number;  
fadeEffect: number;  
particleRadius: number;  
particleNumber: number;  
particleSpeed: number;  
particleAlphaReduction: number;
```

export class Vector

```
public x: number;  
public y: number;
```

```
constructor(x, y);
```