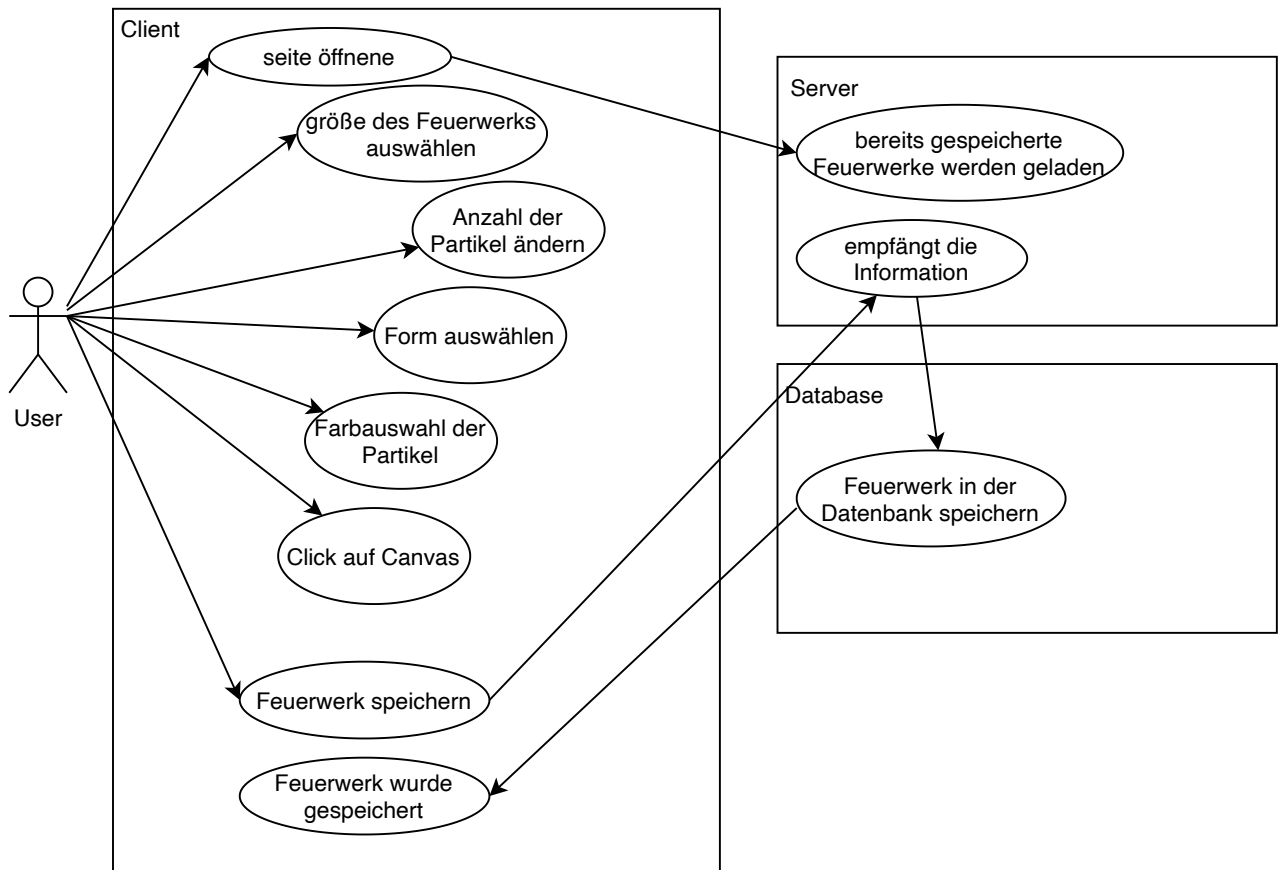
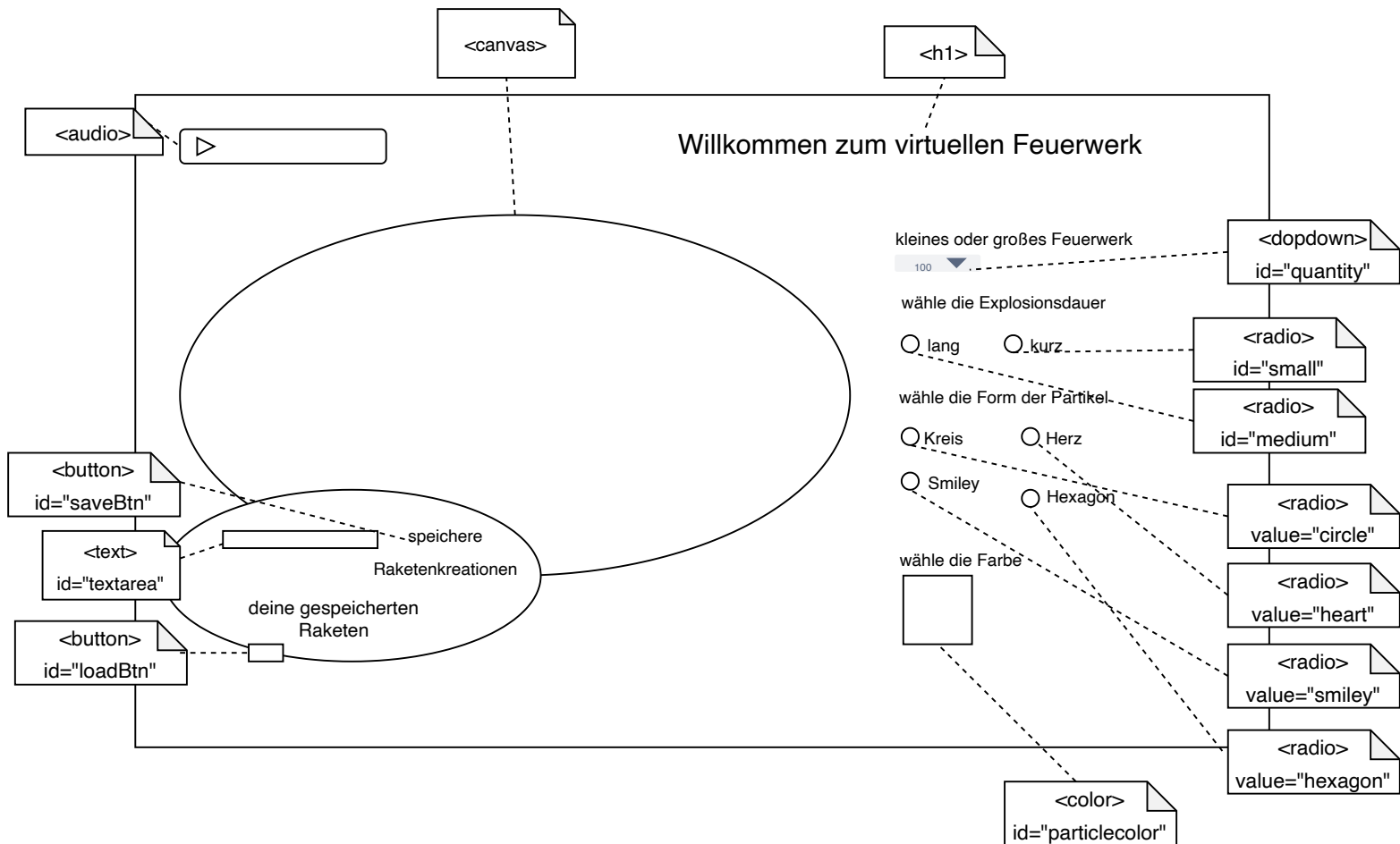


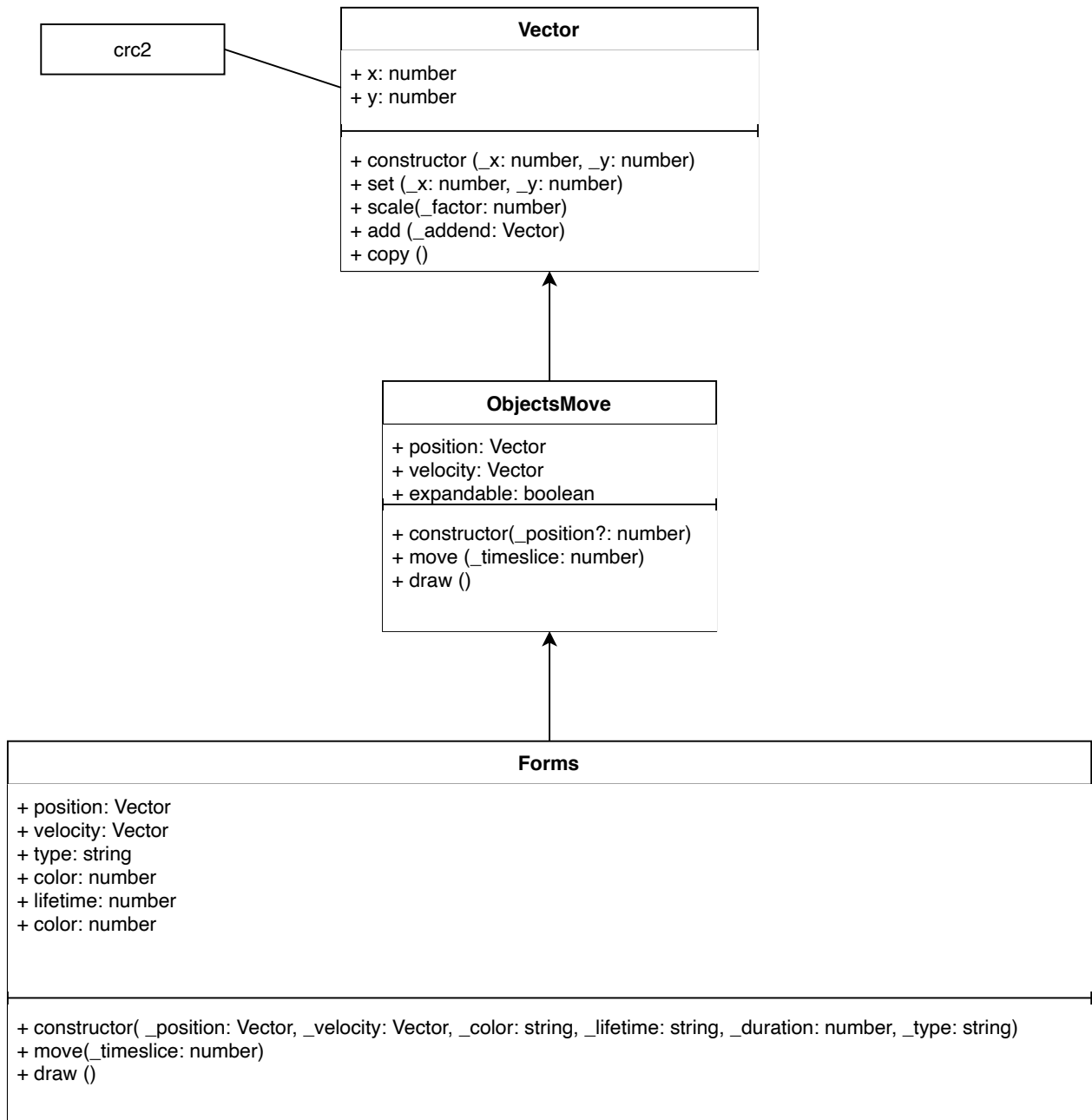
Use-Case-Diagramm



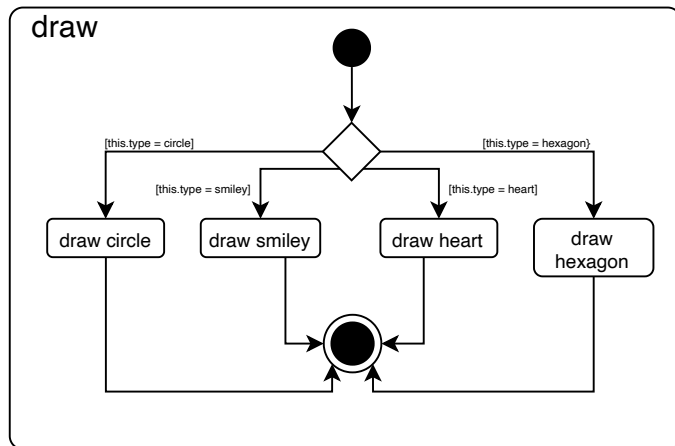
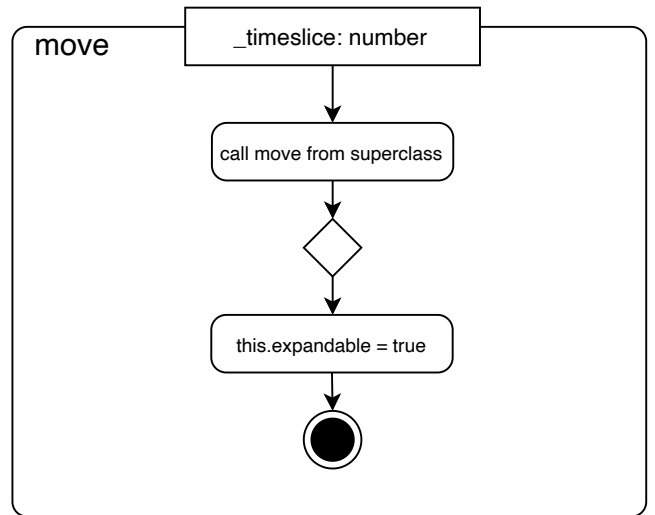
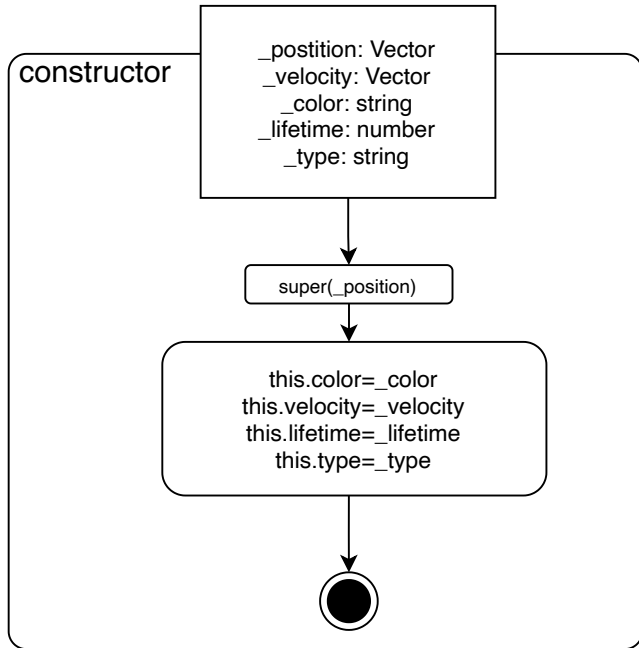
User Interface Diagram



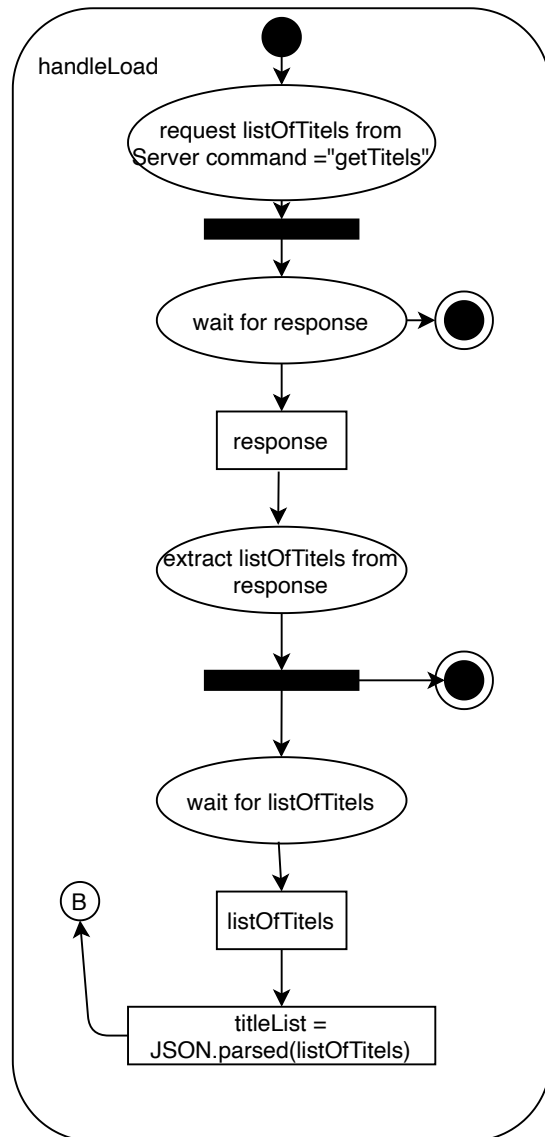
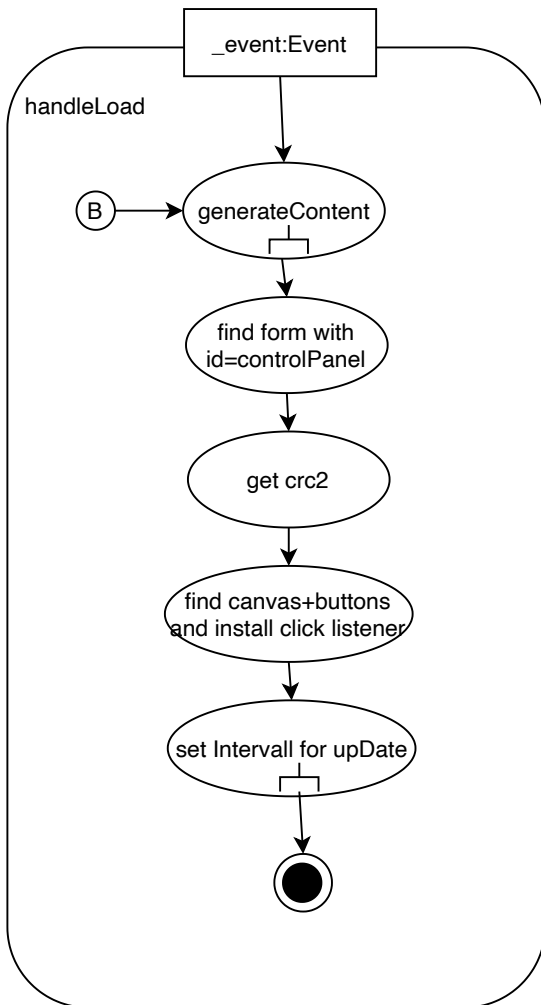
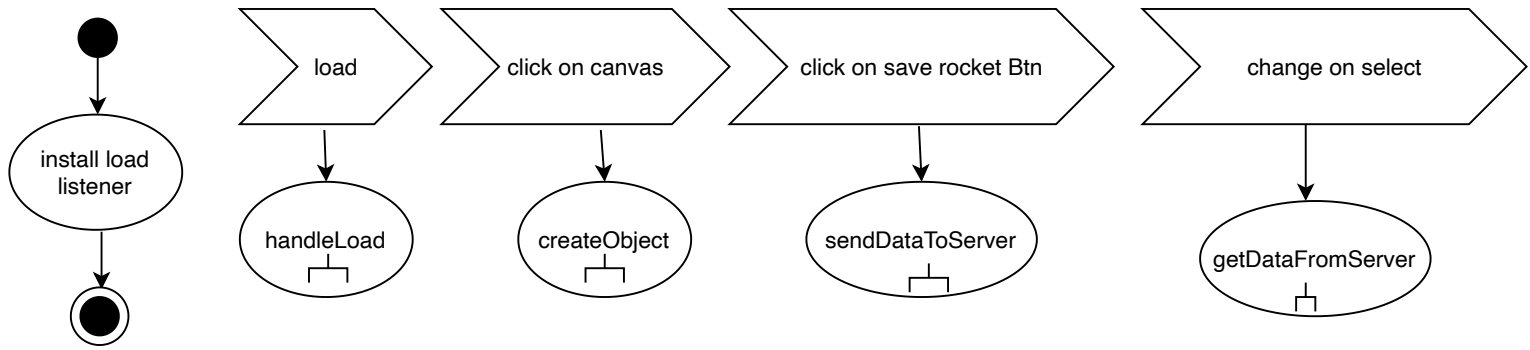
Klassendiagramm

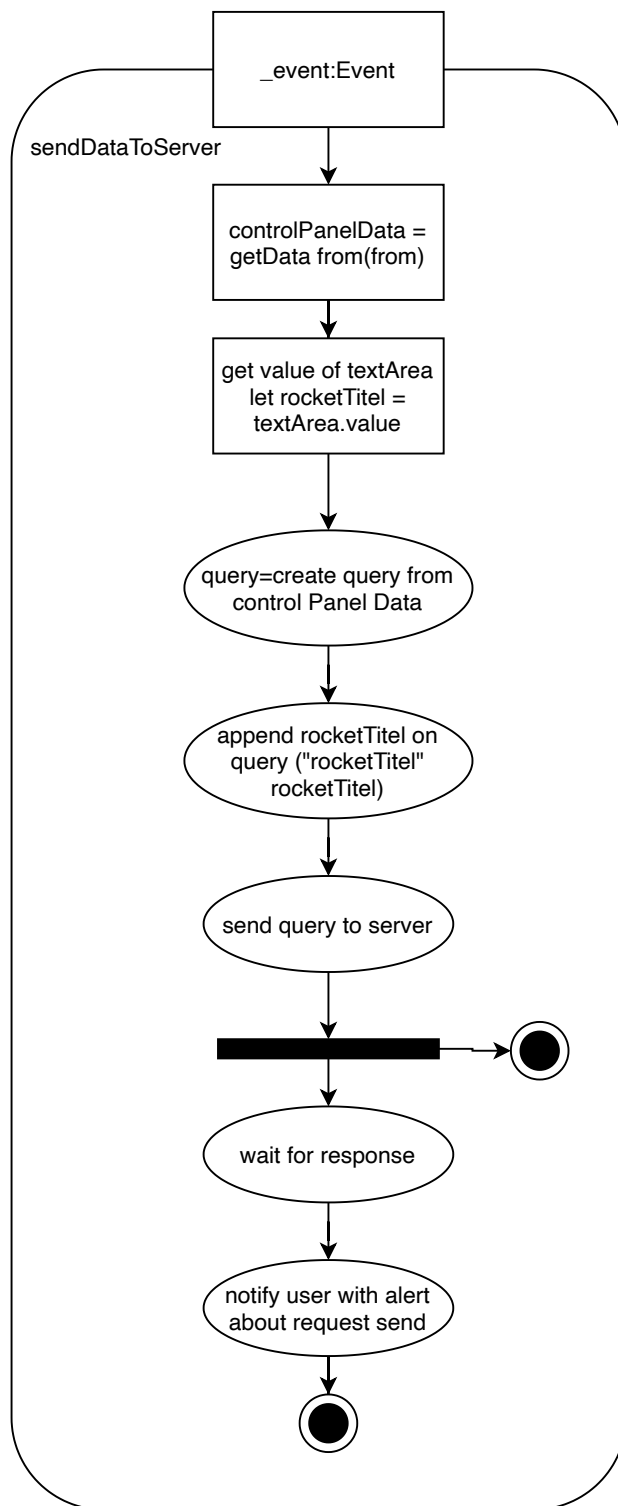
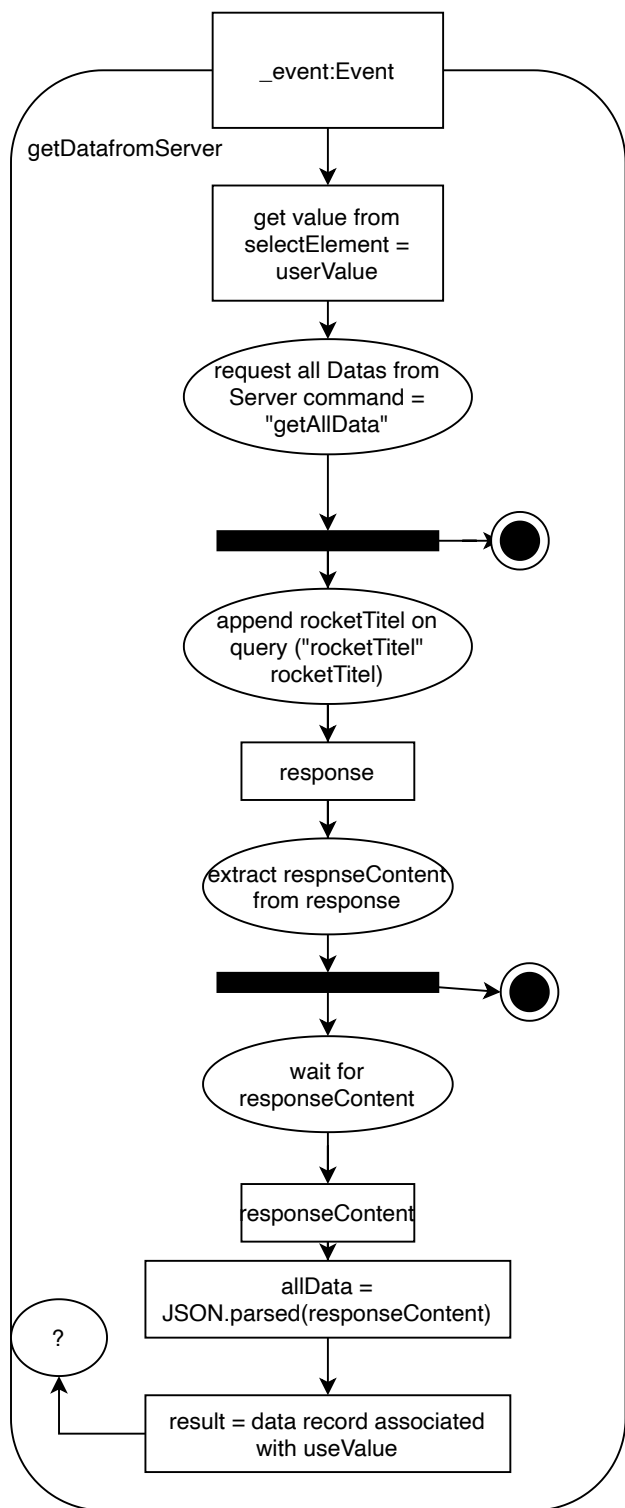


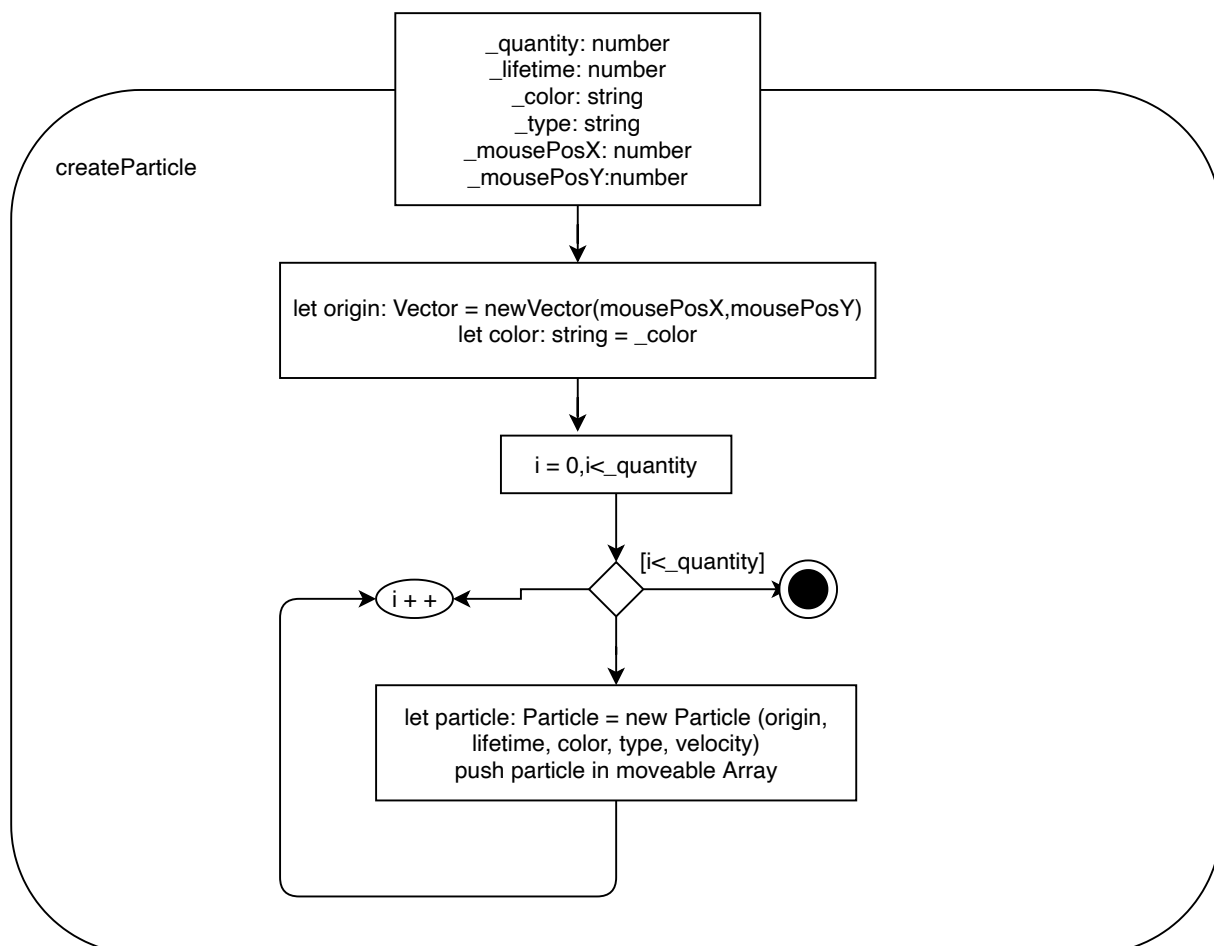
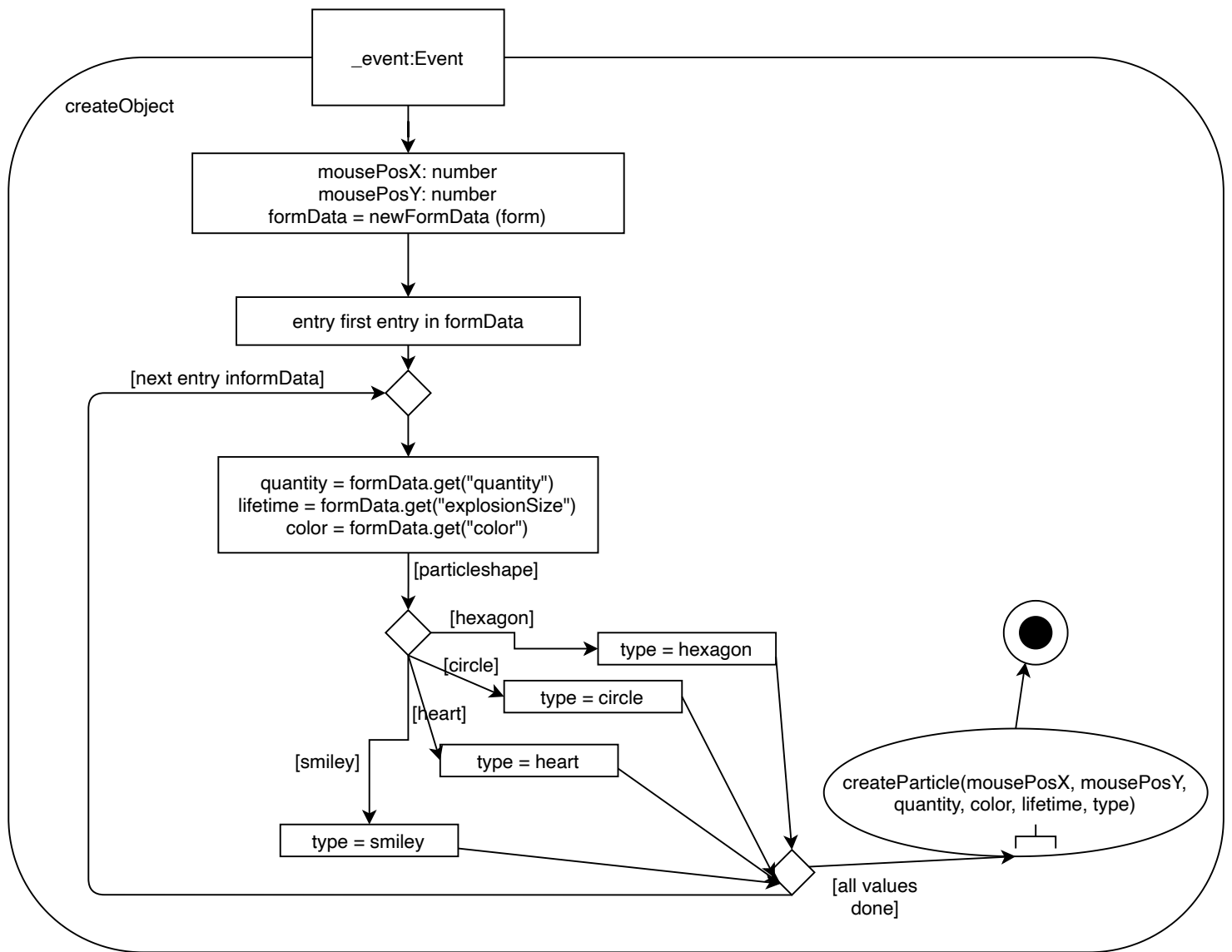
Aktivitätsdiagramm Forms.ts



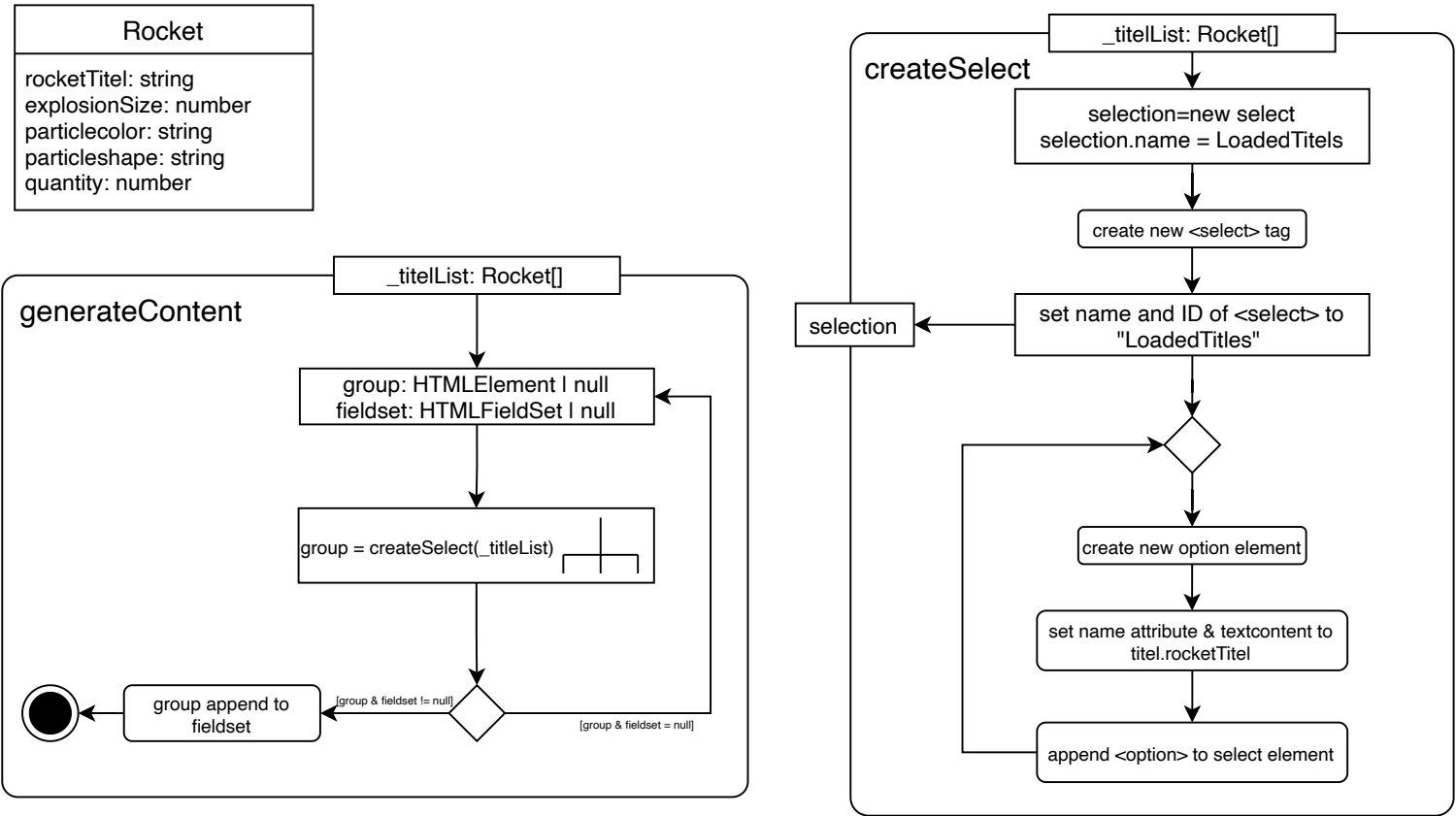
Aktivitätsdiagramm main.ts



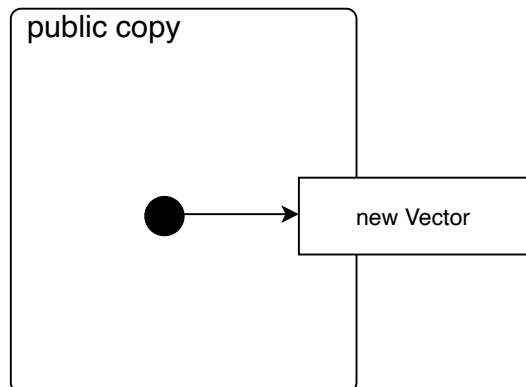
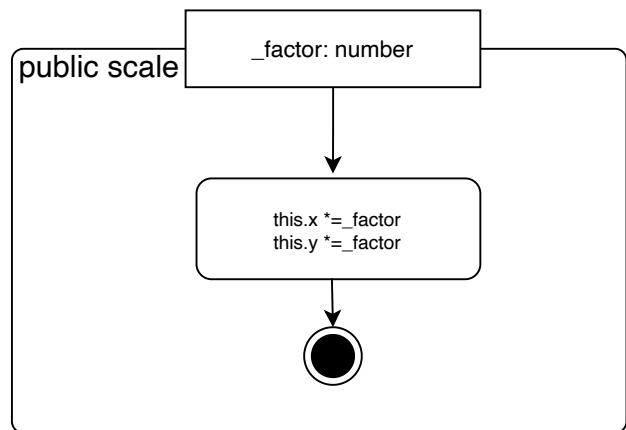
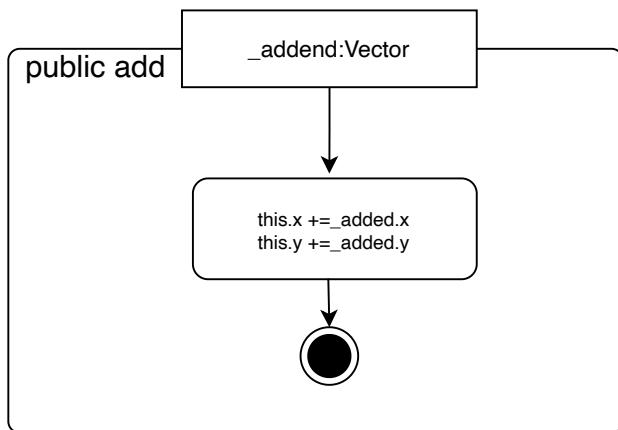
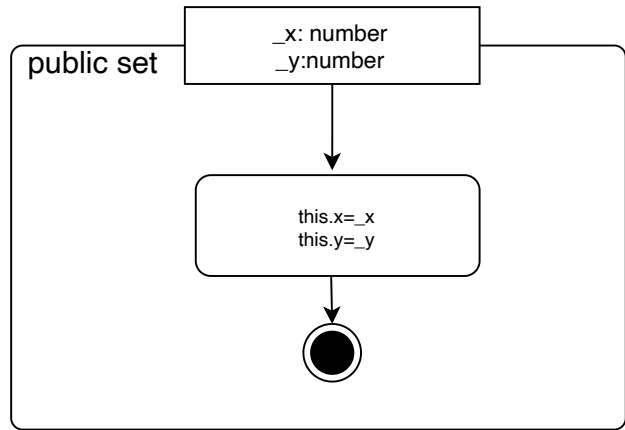
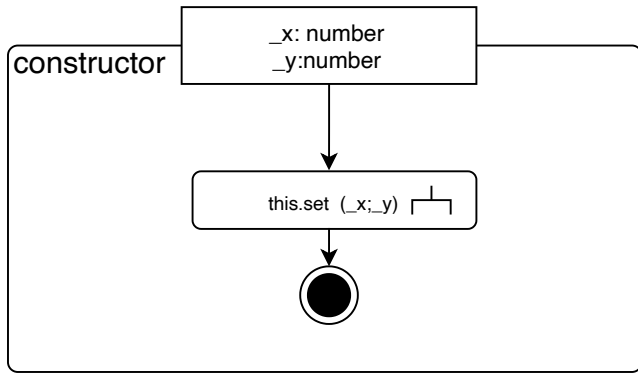




Aktivitätsdiagramm generateRockets.ts



Aktivitätsdiagramm Vector.ts



Abschlussarbeit – Anleitung zur Interaktion

EIA2

Theresa Josephine Gut

MKB 4

Interaktion:

1. Zuallererst kann man die Musik links oben anschalten (für das feeling...)



2. Anschließend kann man die verschiedenen Parameter für die gewünschte Feuerwerksexplosion auswählen. Als aller erstes wählt man die Anzahl der Partikel der Explosion aus (klein oder mega) in dem man von 100 auf 1000 geht.



3. die weiteren Parameter können auch ausgewählt werden, wie die der Explosionsdauer (ob sie lange „glühen“ soll oder eben kürzer.



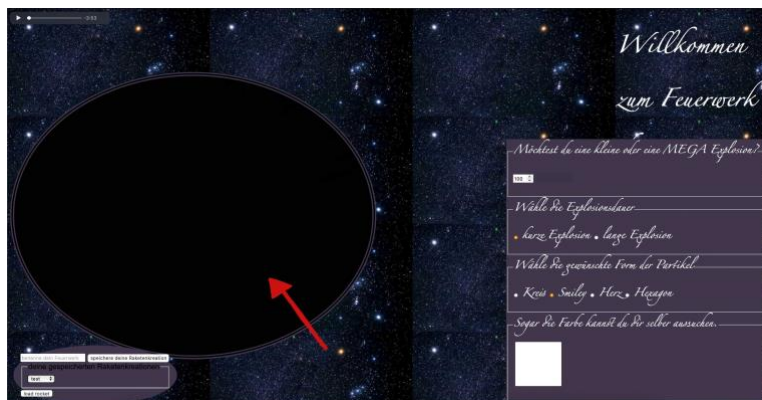
4. Anschließend kommt das beste: Die Auswahl der Form der Partikel. DU kannst zwischen verschiedenen Parametern auswählen.



5. Zuletzt kannst du noch die Farbe auswählen.



6. Dann ist es soweit! Du kannst in dem schwarzen, runden Feld deine Raketen starten lassen.



7. Jetzt kannst du dein Feuerwerk auf dem Server speichern



Das war's. Viel Spaß mit dieser Anwendung! ☺

Abschlussarbeit - Anleitung MongoDB und Heroku

EIA2

Theresa Josephine Gut

MKB 4

Heroku:

1. Bevor Heroku installiert wird, sollte Node mit dem Befehl `npm install @types/node` auf der obersten Ebene des Projektes installiert werden.
2. Lege ein Nutzerkonto auf Heroku an oder logge dich in einem bereits existierenden Account ein. Wichtig: Die primary Language muss node.js sein.
3. Erstelle eine App und verbinde diese mit dem zutreffenden Github-Repository.
4. In der package.json Datei, die bei der Installation von node kreiert wurde, unter „start“ den relativen Pfad zur server.js datei angeben.
5. Ob alles geklappt hat testest du mit dem Befehl `npm start`, ein „build succeeded“ sollte im Heroku-Logs auftauchen.
6. Im Client muss die URL-Variable angepasst werden. Hierzu muss der Pfad zur App an die Stelle der deklarierten Variable URL in der main.ts eingesetzt werden. Zu diesem Pfad kommst du unter „Open App“.

Mongo DB Initialisierung:

1. Installiere zuallererst MongoDB in Node. Gebe hierzu im Terminal auf der obersten Ebene den Befehl `npm install @types/mongodb` sowie `npm install mongodb` ein.
2. Füge in den server.ts datei nun das Mongo-Modul mit dem Befehl `import * as Mongo from „mongodb“;`
3. Erstelle einen neuen Account oder nimm ein bereits erstelltes Benutzerkonto
4. Lege ein neues Cluster an, mache dies mit der IP-Adresse 0.0.0.0/0 von überall aus sichtbar
5. Lege eine neue Collection an, in die du eine Datenbank (in meinem Fall „eia“) und eine Collection (bei mir „score“) erstellst.
6. Um Mongo mit deinem Server verbinden zu können musst du den Link, den du bei Connect - > Connect your Application finden kannst, in die server.ts Datei einbinden (in meinem Fall: `let serverPage: string = "https://eia-endabgabe.herokuapp.com/";`).

THAT'S IT ☺