

# Feuerwerk

---

EIA 2 - ENDABGABE

# FUNKTIONALE ANALYSE

## Erlebnis und Interaktion des Nutzers

Da das Silvesterfeuerwerk dieses Jahr in Deutschland ausgefallen ist bietet diese Anwendung dem Nutzer die Möglichkeit, das Feuerwerk virtuell nachzuholen. Der Nutzer kann sich sein Feuerwerk individuell nach verschiedenen Kriterien wie z.B. Explosionsgröße und Farbe zusammenstellen und auf dem Canvas explodieren lassen. Zudem gibt es die Möglichkeit die erstellten Raketen zu speichern und gespeicherte Raketen zu Laden. Dazu muss klar erkennbar sein, dass es sich bei den Einstellungen der Raketen um interaktive Buttons und Menüs handelt, die er anklicken kann. Dies wird durch eine klare Trennung von Canvas und Einstellungsbereich sichergestellt. Zudem werden alle Einstellungsmöglichkeiten sehr klassisch dargestellt, wodurch die User Experience erheblich verbessert wird. Die Buttons und Regler sind in einer logischen Reihenfolge gegliedert und optisch klar voneinander getrennt, um den Nutzer nicht zu verwirren. Generell ist die Anwendung sehr klar und einfach gehalten.

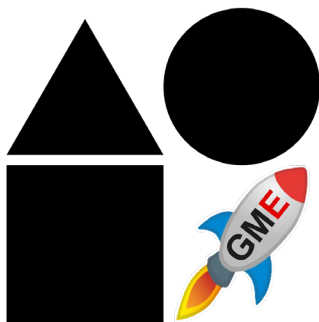
## Plattform für die Anwendung

Als Plattform wird der PC verwendet. Durch die Möglichkeit den Canvas groß darzustellen wird dem Nutzer so ein Intensives Erlebnis geboten. Er kann präzise seine Raketen konfigurieren und diese auf dem Canvas direkt nebendran darstellen lassen. Außerdem ist es angenehmer am PC mehr Zeit zu verbringen wodurch die Anwendung auch ausgiebig benutzt werden kann.

## Was der Nutzer während des Interaktionsverlaufs sieht

### Raketenpartikel

4 verschiedene, in Farbe und Größe variable Formen, können auf dem Canvas platziert werden.



### Audio

Der Partikel GME spielt einzigartige, bis jetzt unbekannte Soundaufnahmen der Nutzer des WallStreetBets Discord Servers ab.

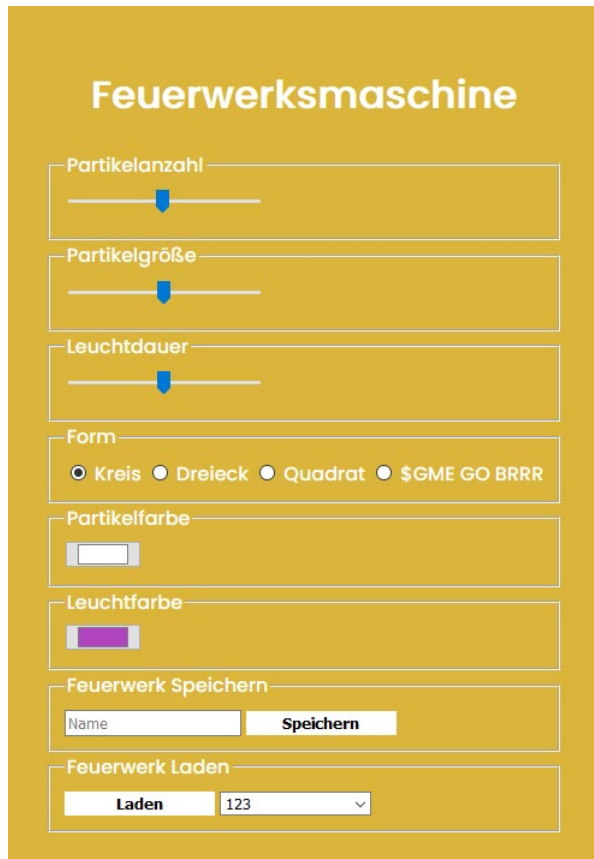


### Farben

Die Partikel können in der Haupt- und Leuchtfarbe in allen Farben des RGB Spektrums angepasst werden.



# ANLEITUNG INTERAKTION ANWENDUNG



Auf der linken Seite lassen sich alle Einstellung über das Feuerwerk treffen. Über die Schieberegler lassen sich die Partikelanzahl, die Partikelgröße und die Leuchtdauer bestimmen. Standardmäßig sind die Regler mittig eingestellt und lassen sich somit nach links und rechts verschieben. Unter den Schieberegler befindet sich die Einstellung der Partikelform. Dort kann der Nutzer über Radiobuttons seine gewünschte Form auswählen. Beim Punkt Partikelfarbe lässt sich die Hauptfarbe der Partikel über einen Color Picker bestimmen. Der zweite Color Picker bestimmt die Leuchtfarbe des Partikels die um ihn herum erscheint und flackert. Zur Auswahl stehen bei beiden Farbeinstellungen alle Farben des RGB Spektrums. Der Nutzer hat jederzeit die Möglichkeit sein Konfiguriertes Feuerwerk mit einem Namen zu speichern. Über das Auswahlménü neben dem Laden Button lassen sich alle jemals gespeicherten Feuerwerke wieder abrufen.

Über den Canvas lassen sich per Mausklick an beliebiger Position die Raketen starten. An dieser Stelle noch eine Erklärung zu dem Hintergrund des Canvas und der „\$GME GO BRRR“ Partikelform. Hierbei geht es um das Subreddit r/wallstreetbets. Diese Community hatte es geschafft, die GameStop Aktie (GME) entgegen der Leerverkäufe großer Hedgefonds in die Höhe zu treiben. Um das zu feiern können die Nutzer hier ebenfalls GME Raketen zum Mond schießen.



# INSTALLATIONSANLEITUNG: HEROKU UND MONGODB

## Heroku

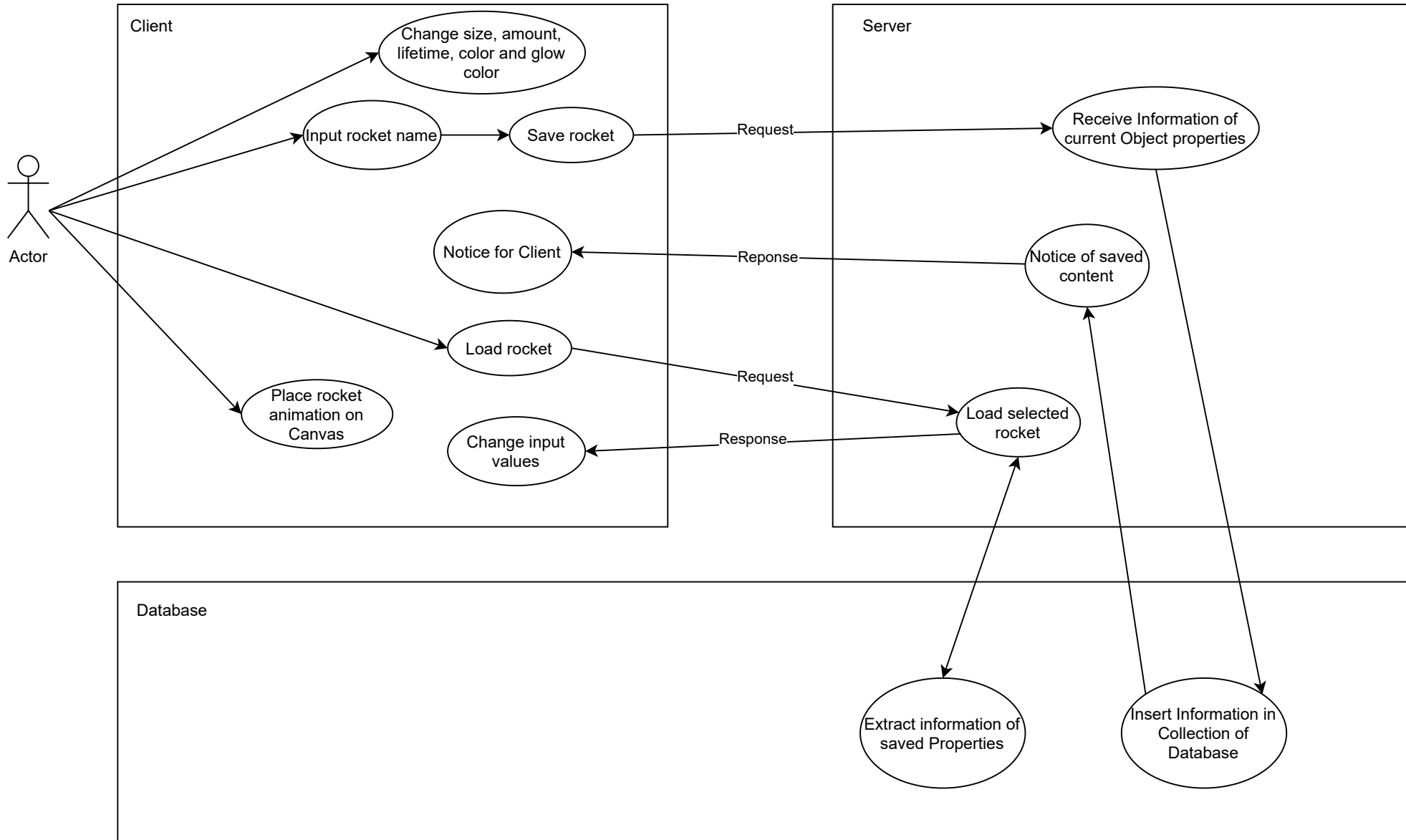
1. Gehe auf die Webseite [www.heroku.com](https://www.heroku.com) und melde dich mit deinem Konto an. Falls kein Nutzerkonto vorhanden ist, registriere dich kostenlos.
2. Erstelle die als nächstes über den Punkt „New“ > „create new app“ eine neue App.
3. Jetzt muss GitHub Repository verknüpft werden. Dazu Unter „Deploy“ „Connect to GitHub“ auswählen und das EIA2 Repository verbinden.
3. Nun muss in der, auf der obersten Ebene liegenden „Package.json“ Datei, der im Eintrag „start“ enthaltende Pfad angepasst werden. Dieser verweist dann auf die „server.js“ Datei. Damit die Daten auch an den richtigen Server gesendet werden, muss in der ts Datei „Main.ts“, in der 4. Zeile der Link ausgetauscht werden. Hier wird der Link, aus der in Punkt eins erstellten Heroku-App gebraucht: <https://eia2-2020-2021.herokuapp.com>
5. Zuletzt muss das Repository nur noch über die Heroku Seite deployed werden.
6. Die fertige Anwendung lässt sich über die „Index.html“ öffnen.

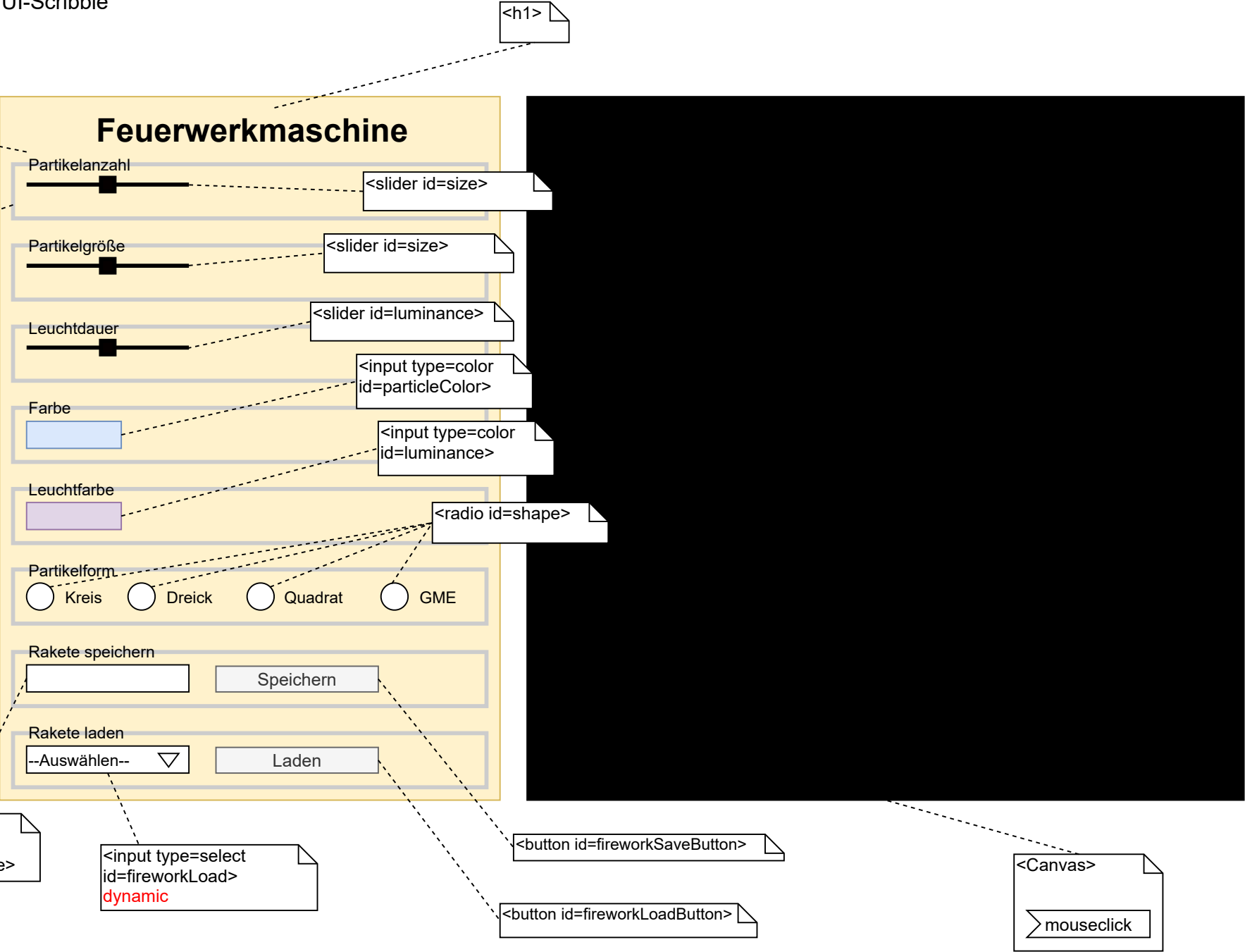
## MongoDB

1. Gehe auf die Webseite [www.mongodb.com](https://www.mongodb.com) und melde dich mit deinem Konto an. Falls kein Nutzerkonto vorhanden ist, registriere dich kostenlos.
2. Erstelle über einen Klick auf „Connect“ einen User. Teile diesem einen Namen und Passwort zu und erlaube uneingeschränkten Zugriff von überall.
3. Erstelle eine Datenbank mit dem Namen „Fireworks“ erstellen
4. Erstelle in dieser Datenbank eine neue Collection mit dem Namen „drawings“. In der 12. Zeile der ts Datei „Server.ts“ den Wert der Variable `databaseUrl` mit der URL für den erstellten Nutzer ersetzen. Die Url lässt sich bei mongoDB über den Punkt Connect>Connect your application abrufen und sieht folgendermaßen aus "mongodb+srv://DerPapa:<password>@eia2-2020-2021-endabgab.a47ra.mongodb.net/<dbname>?retryWrites=true&w=majority"

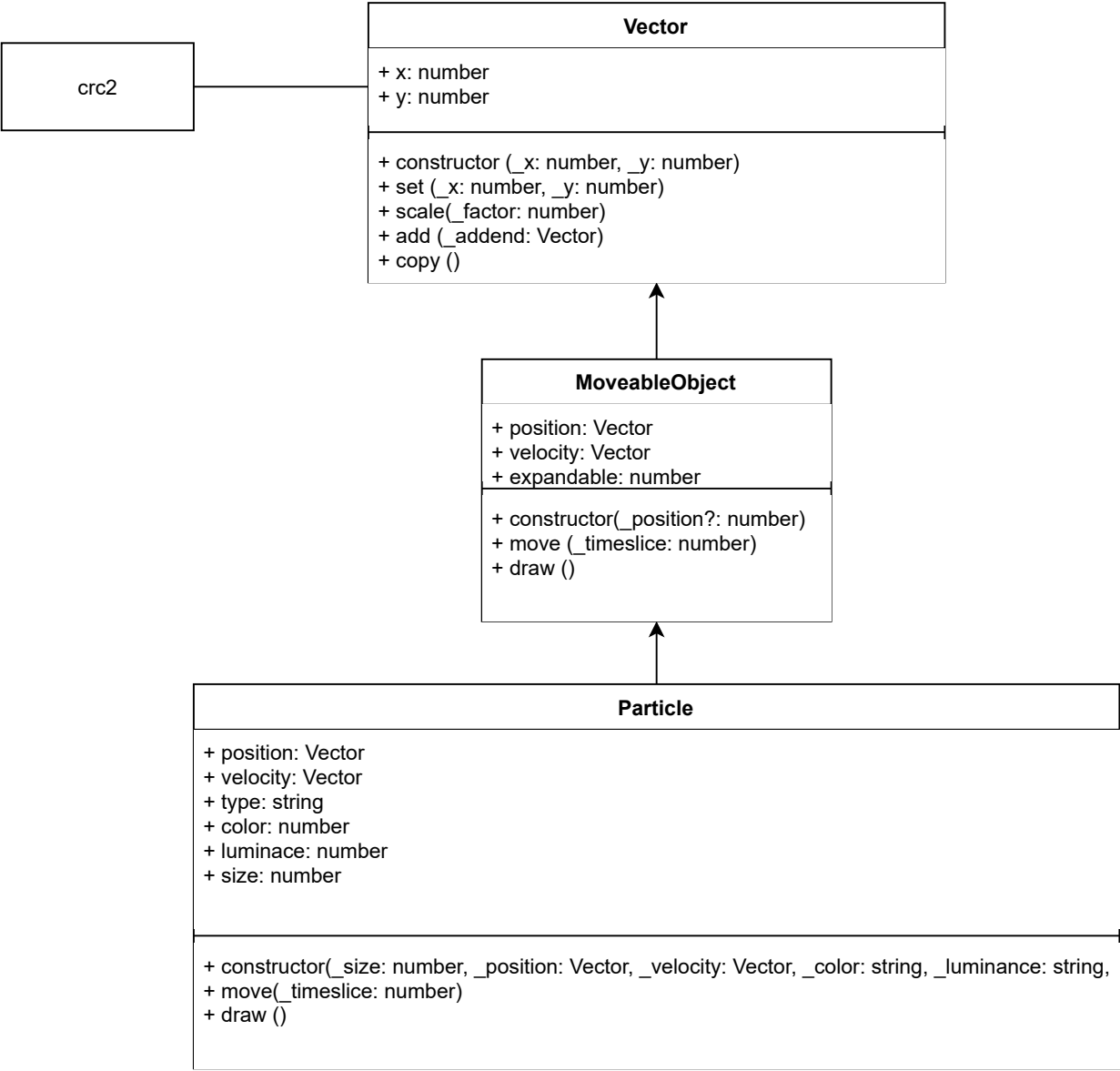
Die Platzhalter <password> und <dbname> müssen mit dem Passwort des Users und dem Namen der Datenbank ausgetauscht werden. Danach muss das Skript gespeichert, kompiliert und auf GitHub gepusht werden.

## Use-Case Diagram

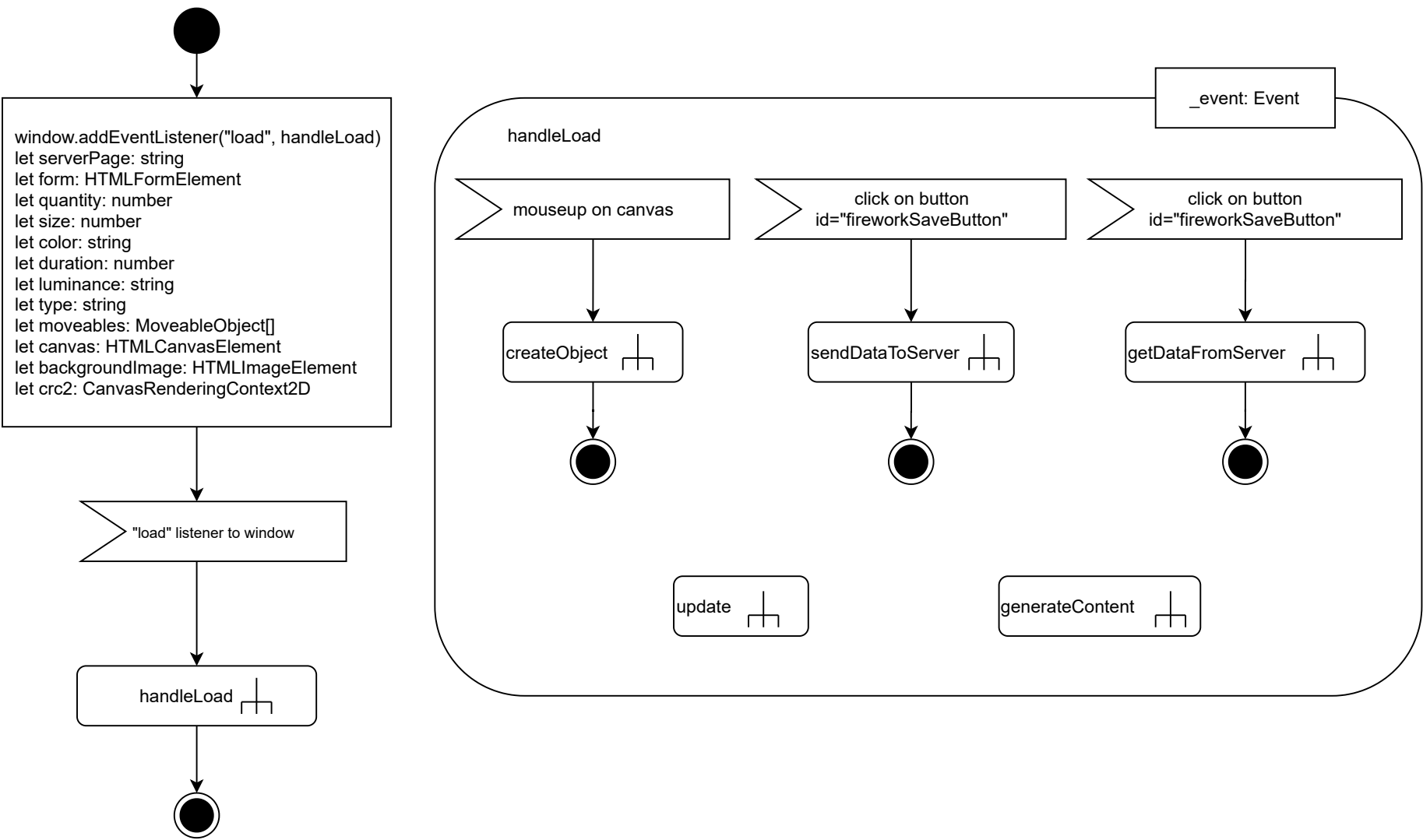




Class Diagram

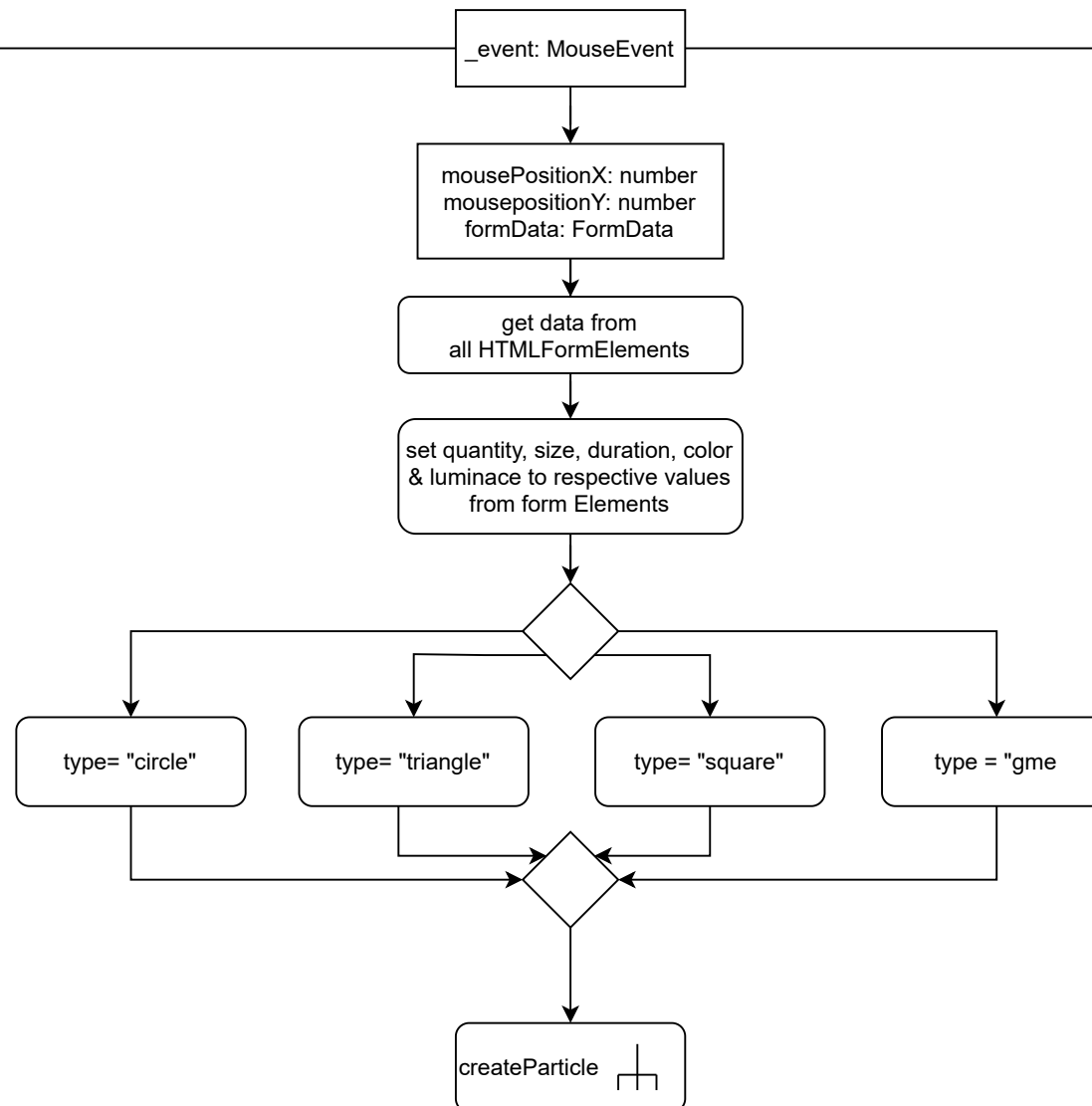


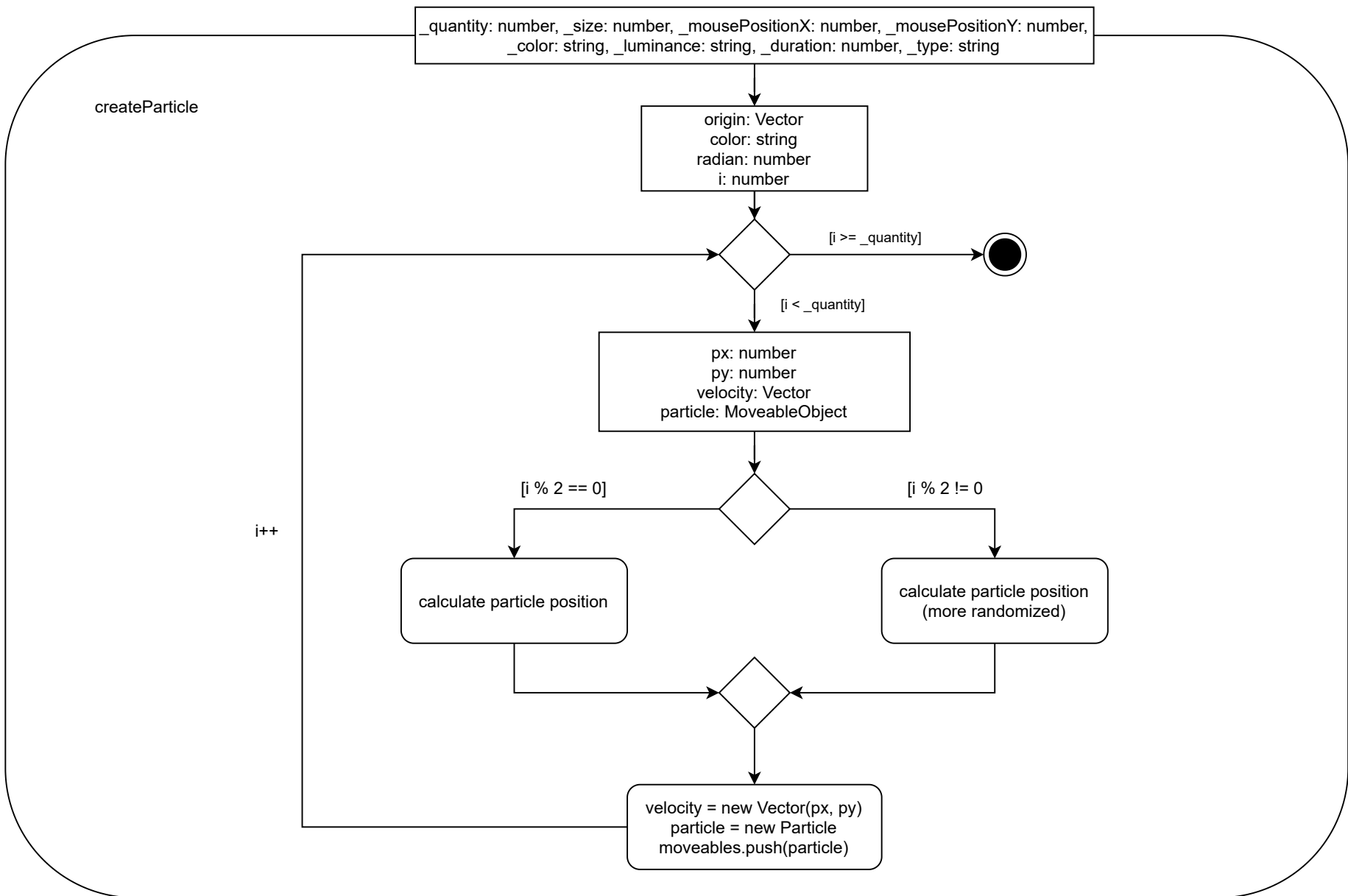
Activity Diagram - Main.ts

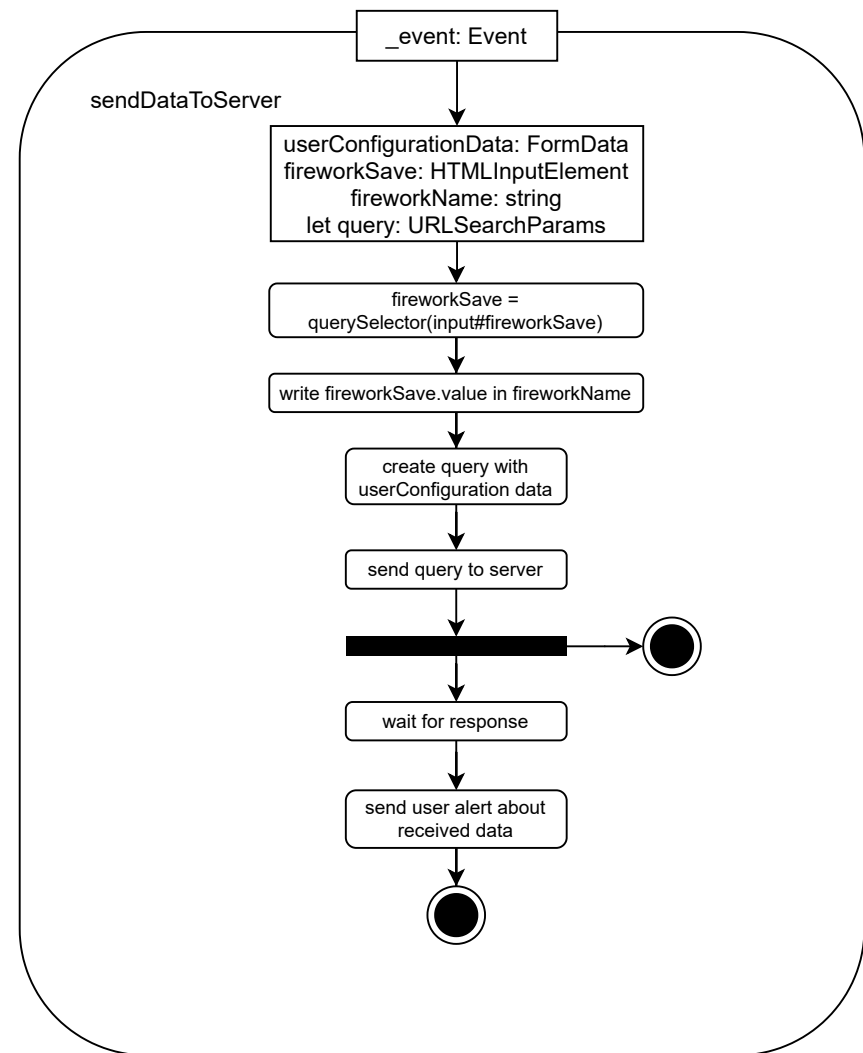
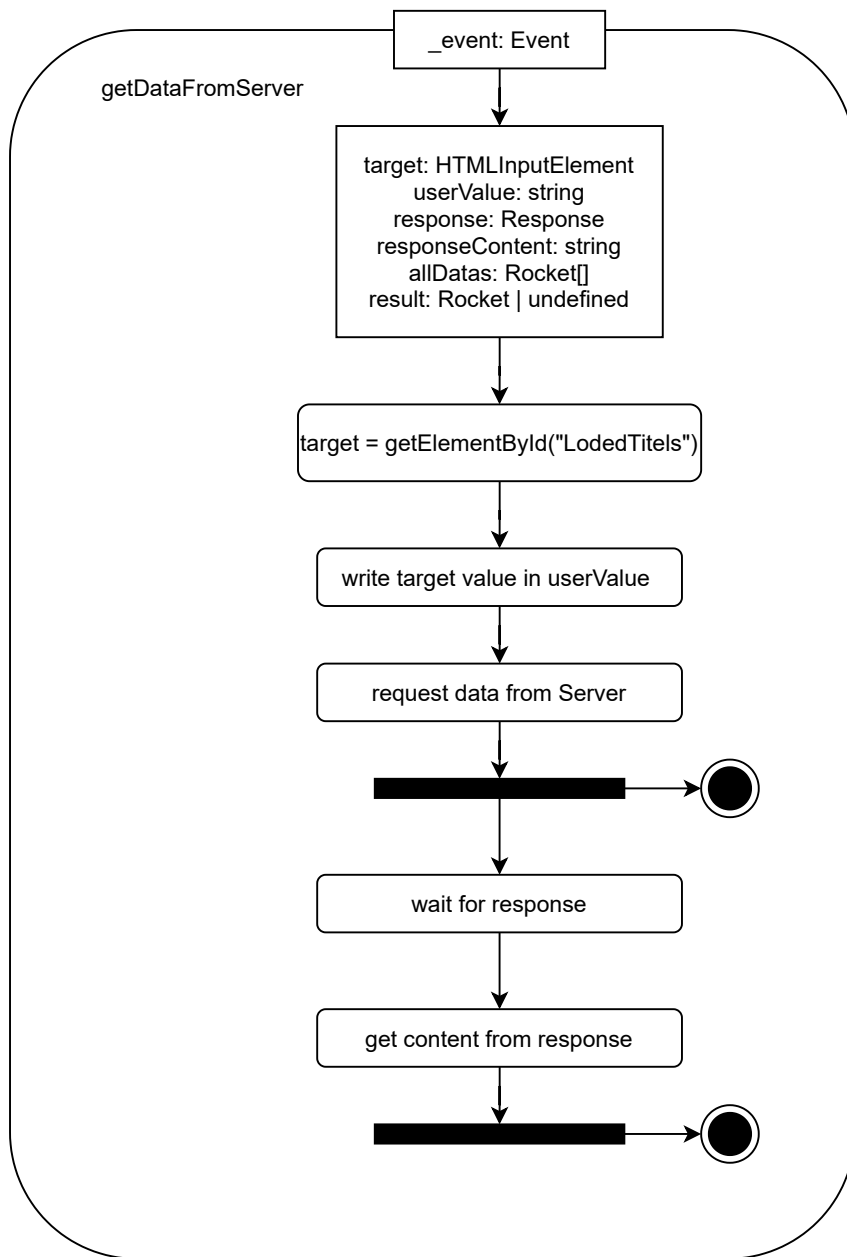




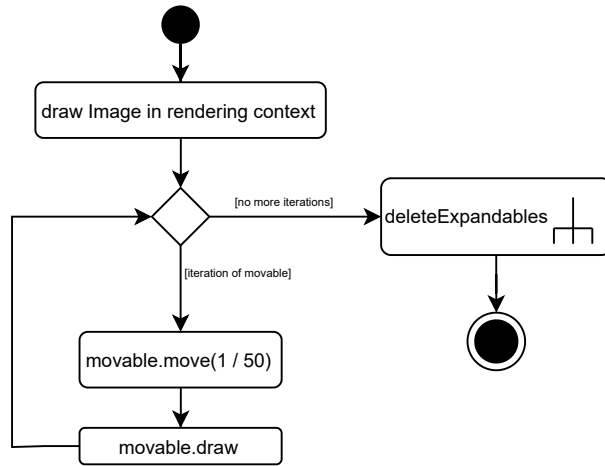
createObject



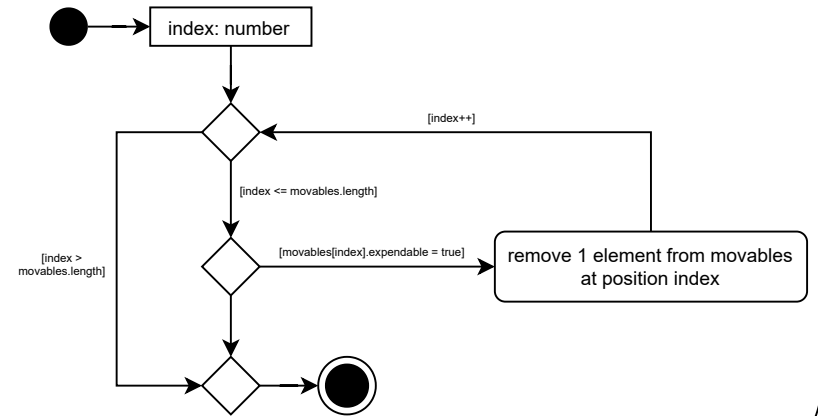




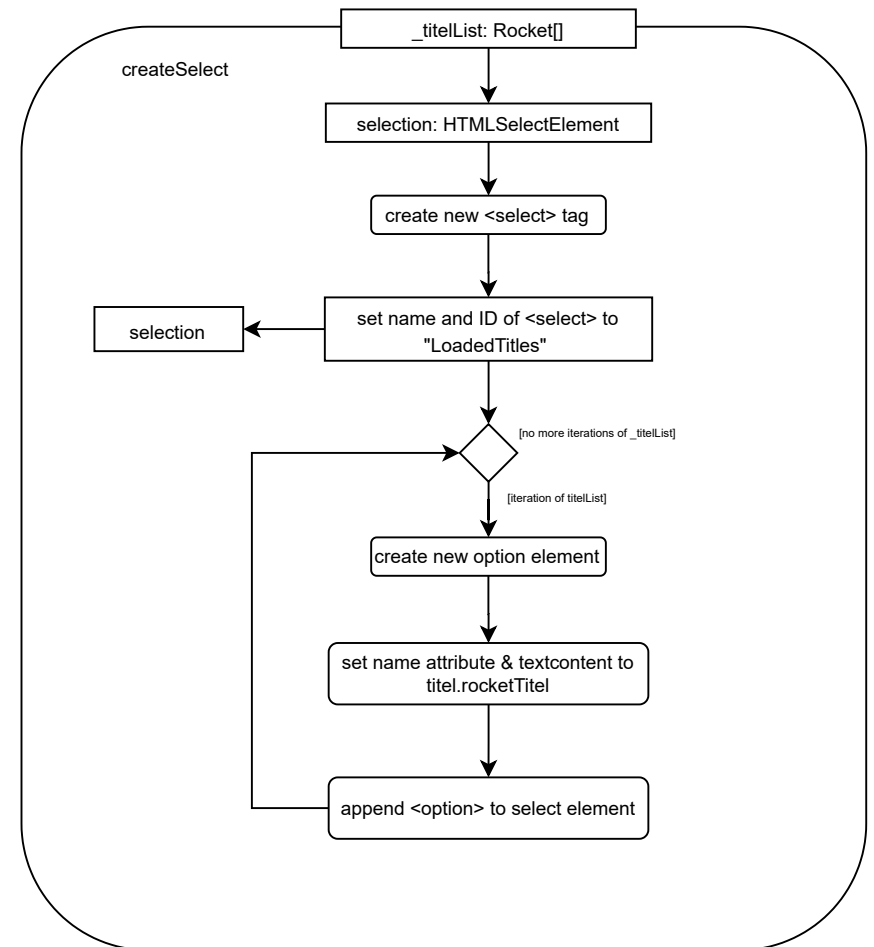
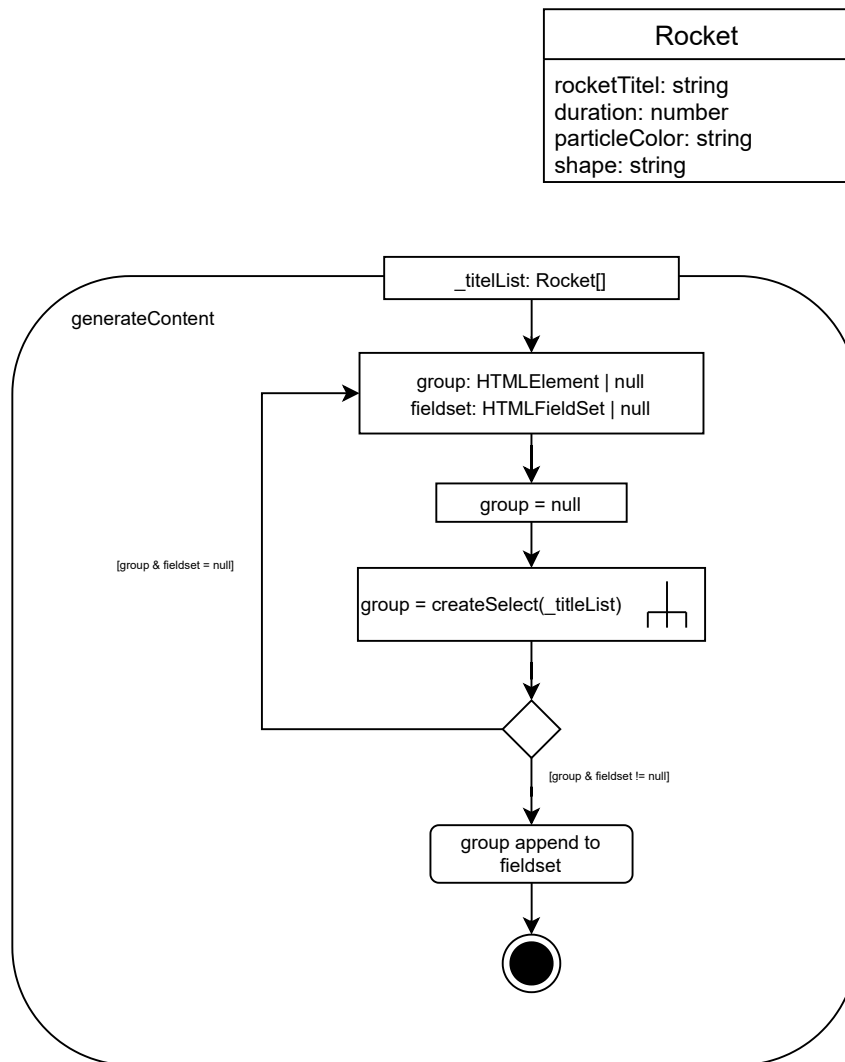
update



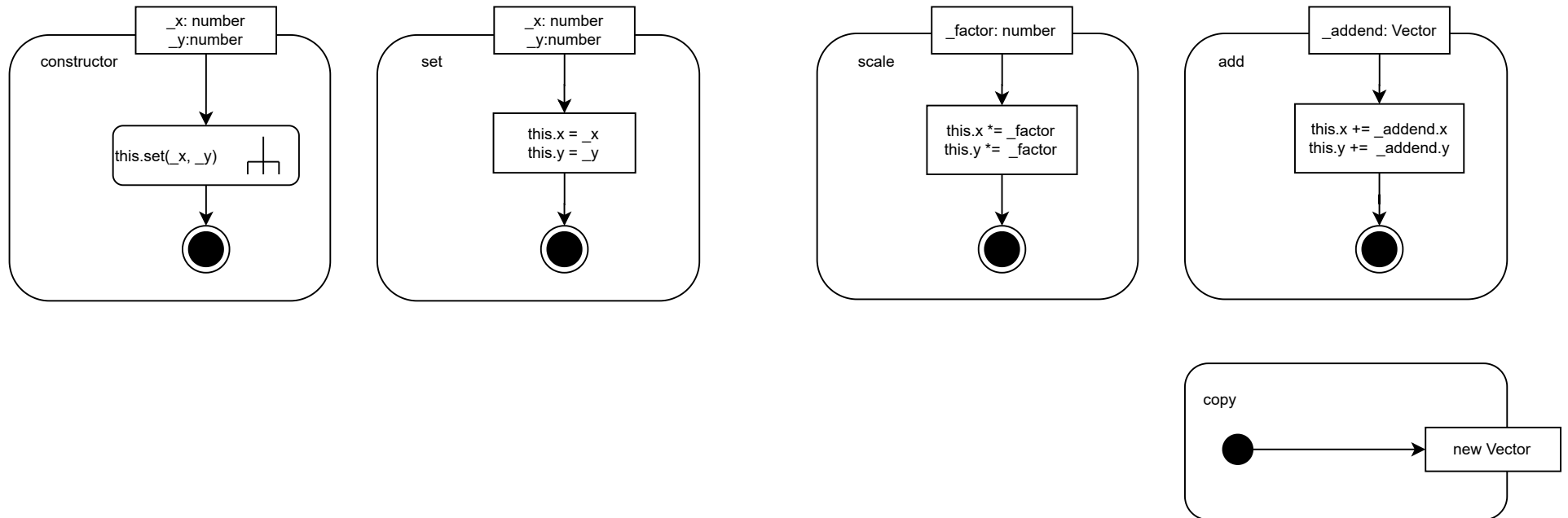
deleteExpandables



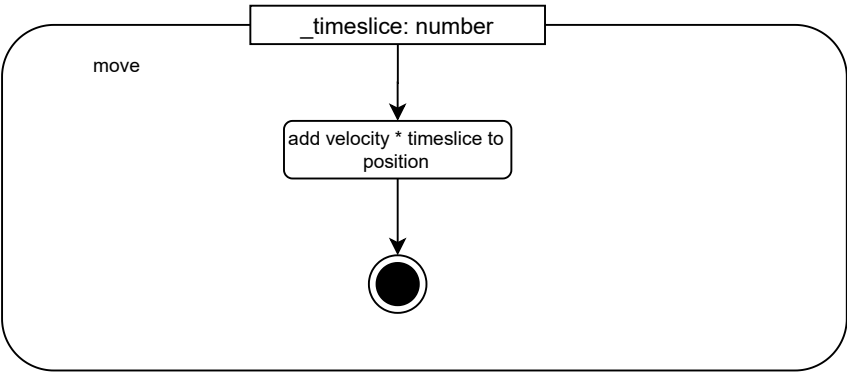
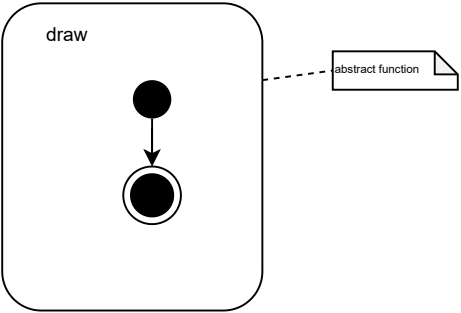
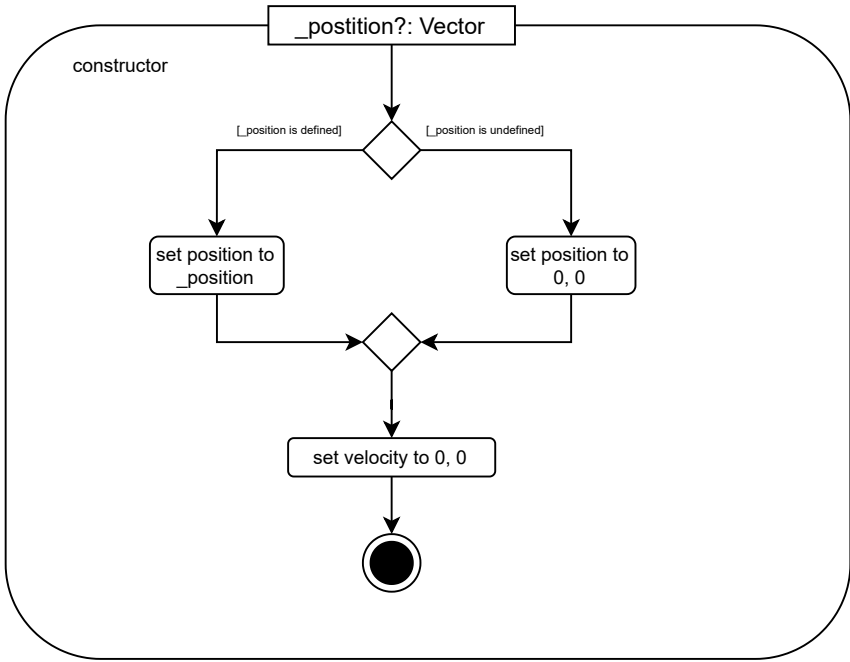
## Activity Diagram - generateContent.ts



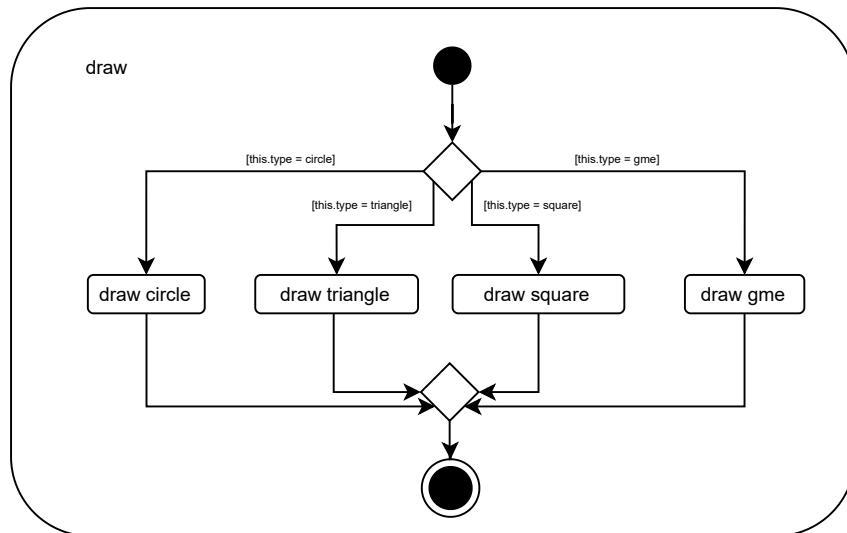
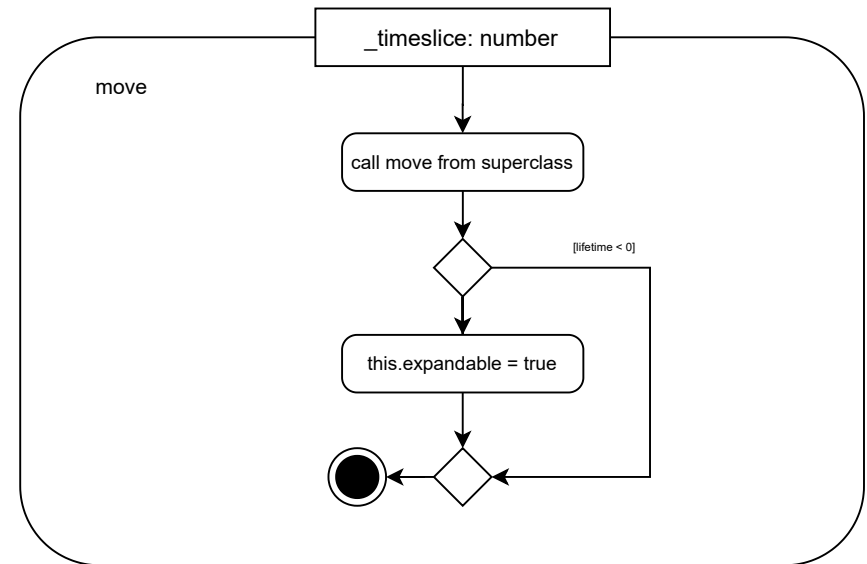
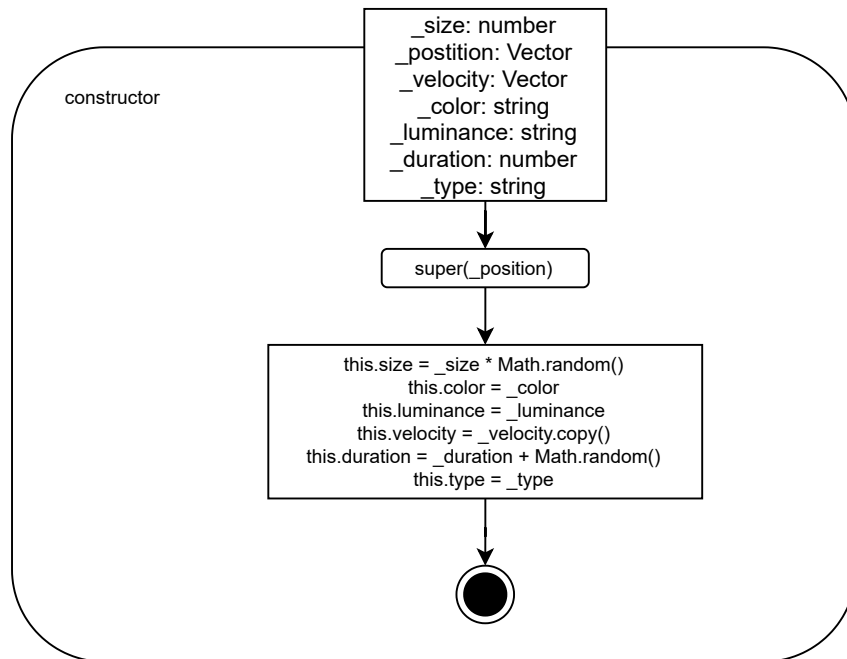
## Activity Diagram - Vector.ts



Activity Diagram - MoveableObject.ts



## Activity Diagram - Particle.ts





## Activity Diagram - Server.ts

