

Funktionale Analyse - MagicCanvas

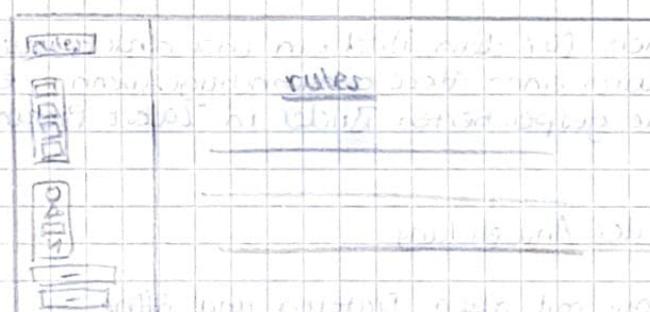
Plattform: PC

- mehr Platz für verschiedene Auswahlmöglichkeiten
- bessere Übersichtlichkeit
- bessere Testbarkeit
- Elemente können stärker in der Größe variieren
- geteilte Auswahl → auch Hover möglich

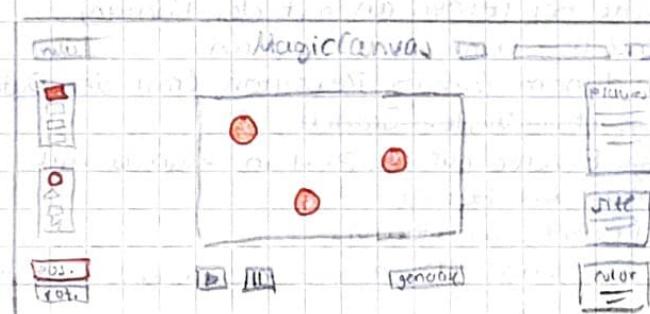
Interaktionsverlauf



1. Home



2. Rules - Overlay



3. Ausgewählt



4. Bild speichern

Infos die zum Nutzer gelangen

- alert, wenn ~~die Hochzeit~~ das Bild gespeichert und an den Server geschickt wurde
- Liste mit gespeicherten Bildern, zum Auswählen
- farbiger Feedback beim Auswählen der Farbe, Form und Animationform

Anleitung zur Interaktion mit der Anwendung

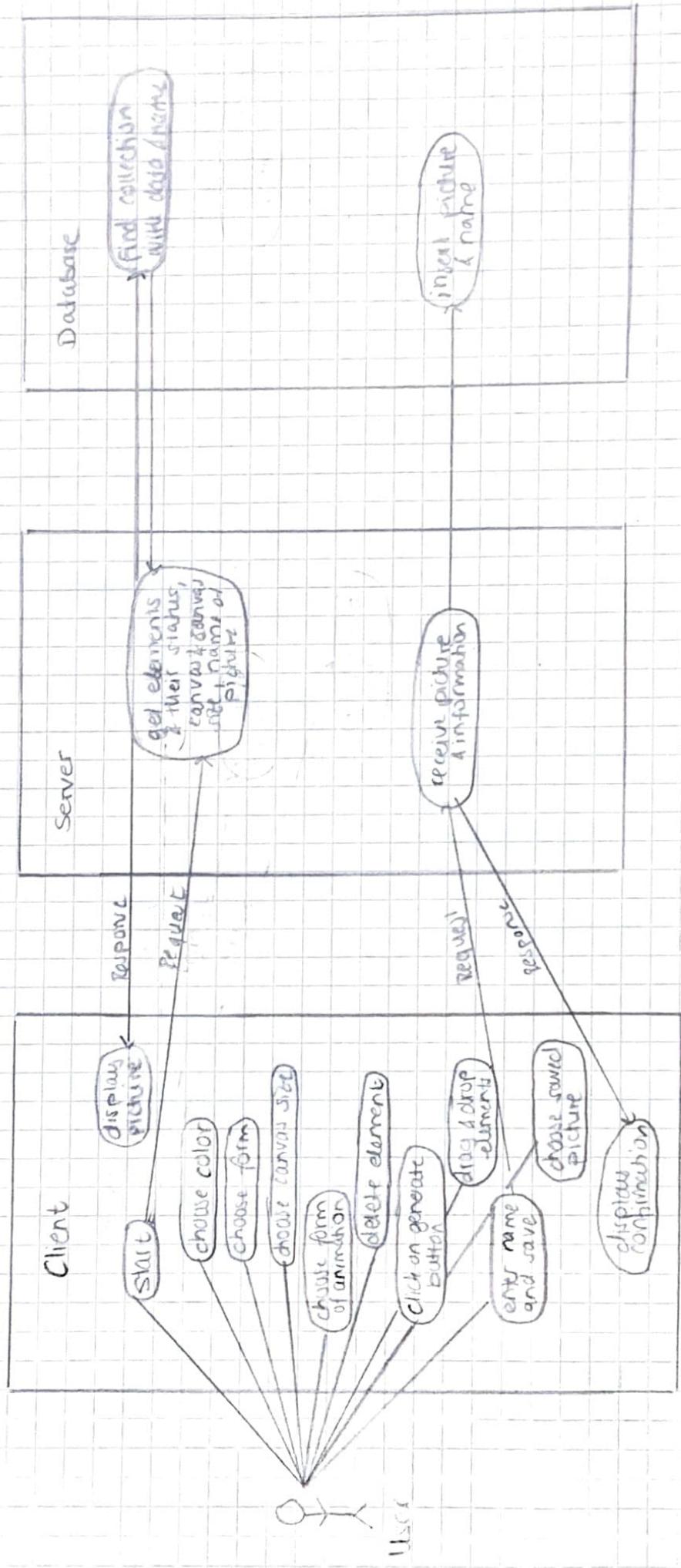
- 1) Wähle Canvas Größe
- 2) Wähle Hintergrundfarbe des Canvas
- 3) Wähle eine Farbe + Symbol
- 4) Wähle optional eine Animationsform (position oder rotate)
- 5) Drücke generate (Element wird dann links oben angezeigt)
- 6) Um die Animation zu starten: auf start drücken
- 7) Um Animation zu stoppen: auf stop drücken
- 8) Verschiebe das Element, in dem du drauf klickst und es an die gewünschte Stelle ziehst
- 9) Du kannst deine Leinwand mit einem Klick auf delete säubern
- 10) Gib einen Namen für dein Bild ein und drücke auf save. Du wirst durch einen Alert benachrichtigt, wenn es erfolgreich war.
- 11) Du findest deine gespeicherten Bilder in "Latest Pictures" oben rechts

Anleitung zur Installation der Anwendung

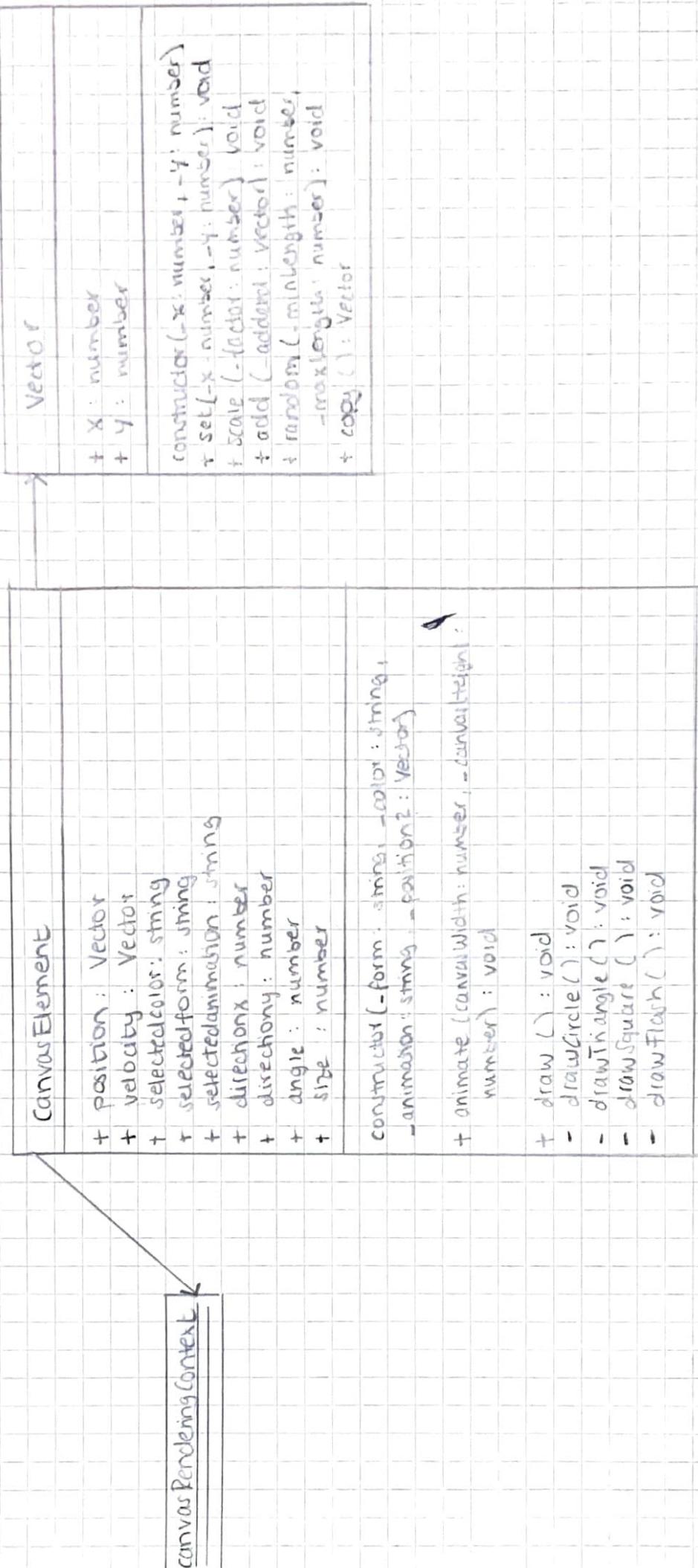
- 1) Pull GitHub Repository mit allen Dateien und öffne sie mit eigenem Programm (VS-Code)
- 2) Lege dir einen Account bei Heroku an mit der Primary Language - Node.js und lege eine neue App an
- 3) Verbinde Heroku mit deinem GitHub Repository (mit den Dateien der App), und deploye den Master Branch
- 4) Erstelle deine package Ersetze die appurl in Main.js mit deiner app url der App auf Heroku
- 5) Deploy den Master Branch auf Heroku
- 6) Lege einen Account bei MongoDB an und kreiere einen Cluster in Europe
- 7) Lege einen Testuser an und gebe ihm read und write Zugriff
- 8) Wähle connect your Application und ersetze den Connection String mit dem der databaseurl (jewer.js)
- 9) Lege eine Datenbank an
- 10) Erstelle ein Bild auf der Anwendung und speicher es unter einem eindeutigen Namen.
- 11) Finde den Namen des Bildes auf der rechten Seite

Magic Canvas -

Use Case Diagram



Magic Canvas - Class Diagram



Magic Canvas

- User interface

۲۷۱

250 units
160 miles
\\$100

Rules

Magic Canvas

```
<div>  
    id: '100'</div>
```

1

100

\triangle \leftarrow \leftarrow \leftarrow

1

120

102

107

4

4

Position
Potate

stop

generate

generate

3 3 3
id = password
3 3 3

Organization

$\text{P}_t = \frac{\partial \ln P}{\partial t}$

definition
id = "generate"

button
id:islaranimation

Magic Canvas

`<input type="button" value="Delete" />`

`<input type="text" id="name" value="Enter name" />`

`<input type="button" value="Save" />`

`<div> Latest pictures </div>`

`<div> <input type="radio" value="standard" checked="" /> standard
 <input type="radio" value="small" /> small
 <input type="radio" value="medium" /> medium
 <input type="radio" value="large" /> large </div>`

`<div> <input type="radio" value="white" checked="" /> white
 <input type="radio" value="black" /> black
 <input type="radio" value="orange" /> orange
 <input type="radio" value="yellow" /> yellow </div>`

`<div> <input type="radio" value="white" id="white" checked="" /> white </div>`

`<input type="button" value="Start" />`

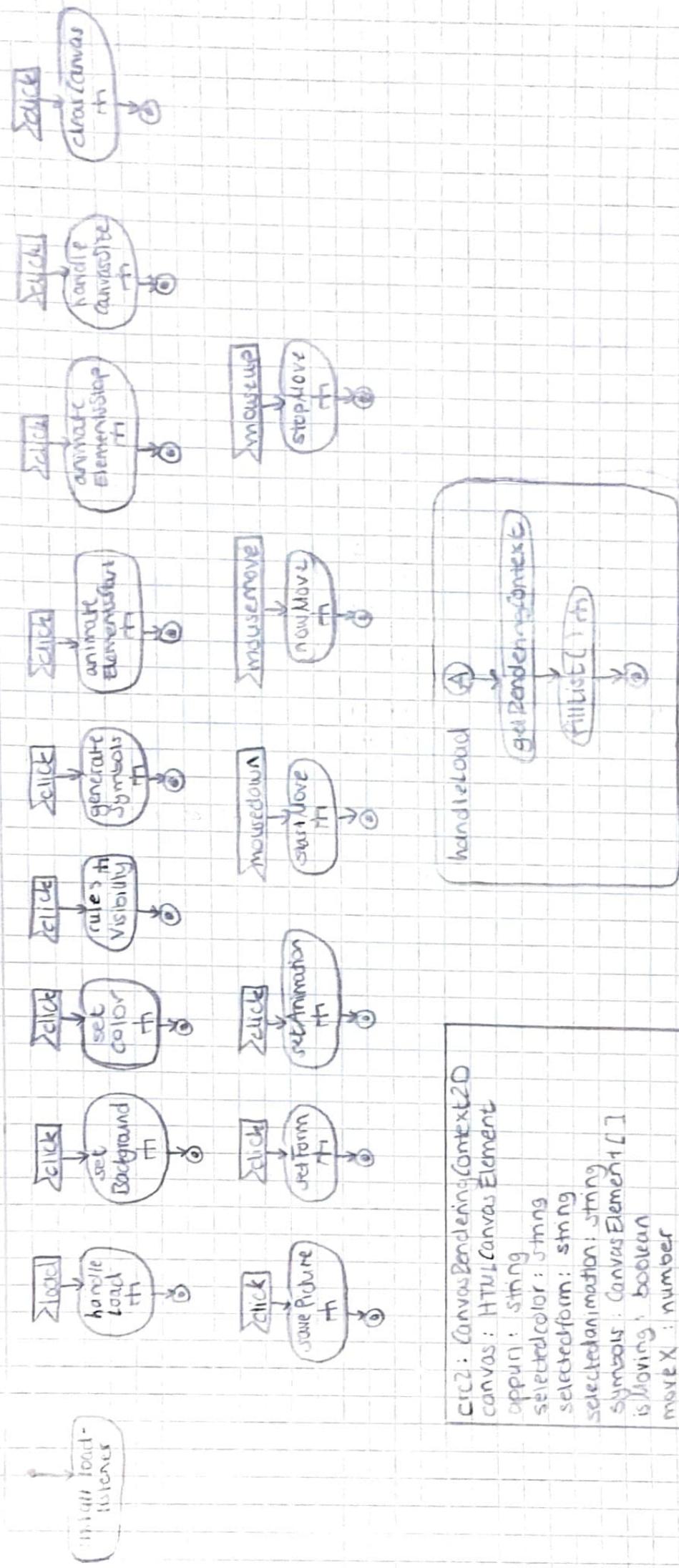
`<input type="button" value="Stop" />`

`<input type="button" value="Generate" />`

`<div> <input type="radio" value="startAnimation" checked="" /> startAnimation
 <input type="radio" value="stopAnimation" /> stopAnimation </div>`

Magic canvas

- Activity Diagram
- Main (A)

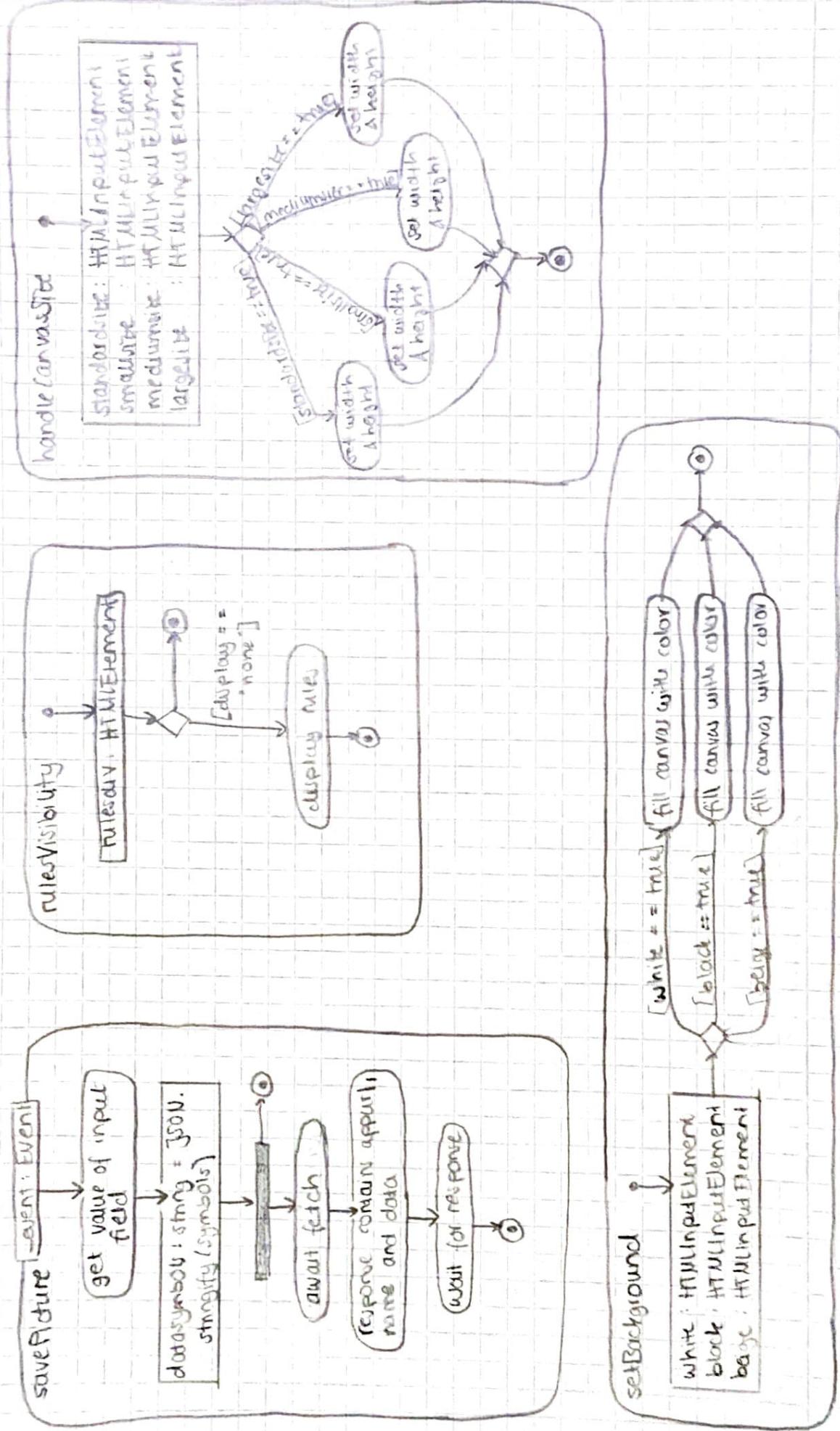


```

circ2 : CanvasBendingContext2D
canvas : HTMLCanvasElement
oppurn : string
selectedcolor : string
selectedform : string
selectedanimation : string
symbols : CanvasElement[ ]
isMoving : boolean
moveX : number
moveY : number
draggedElementIndex : number
timeOut : any
animationRunning : boolean
xpos : number
year : number
index : number
  
```

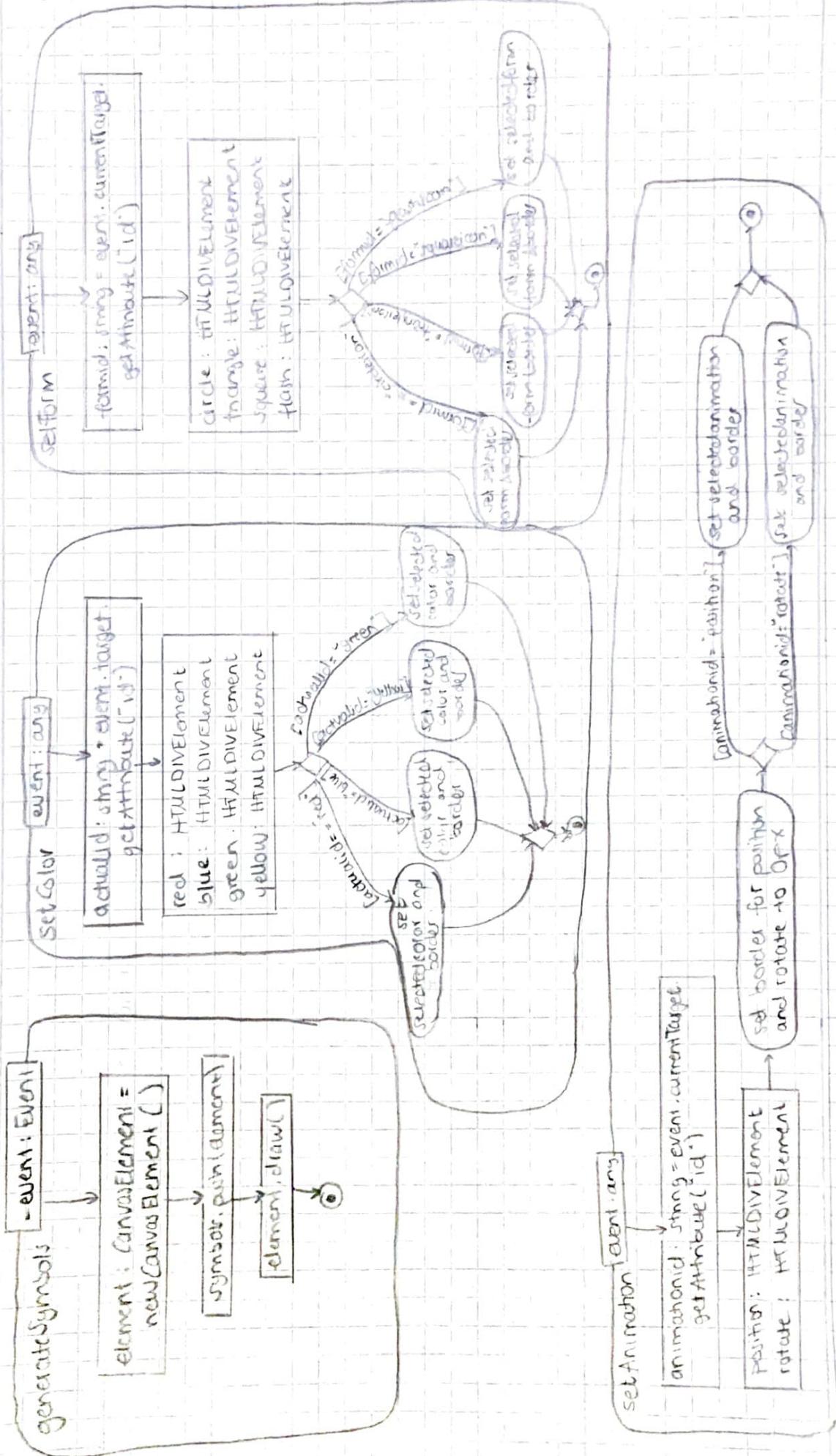
Magic Canvas

- Activity Diagram - Main (2)



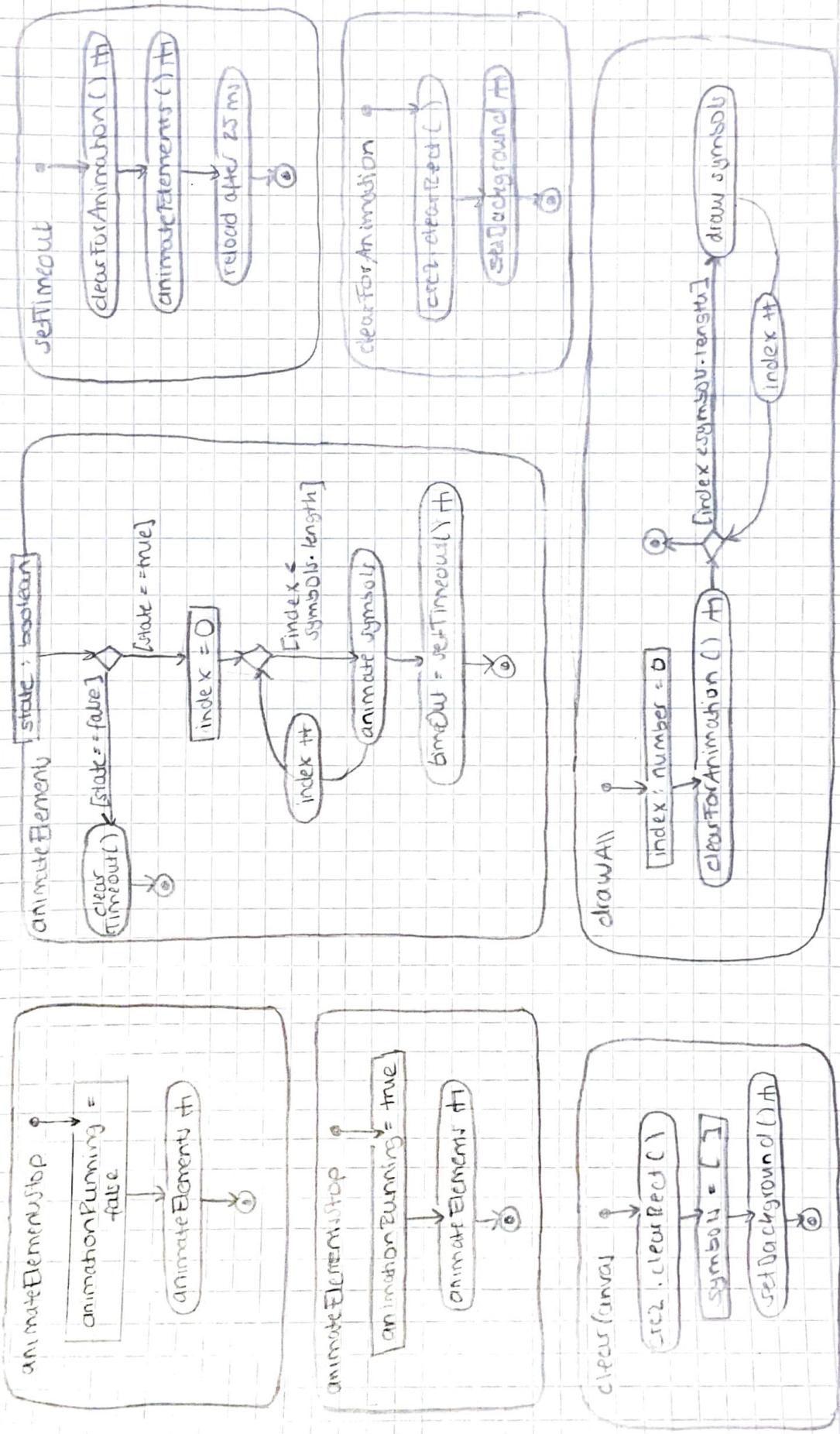
Magic Canvas

- ACHIVITY BLOGAM - MAMU (3)



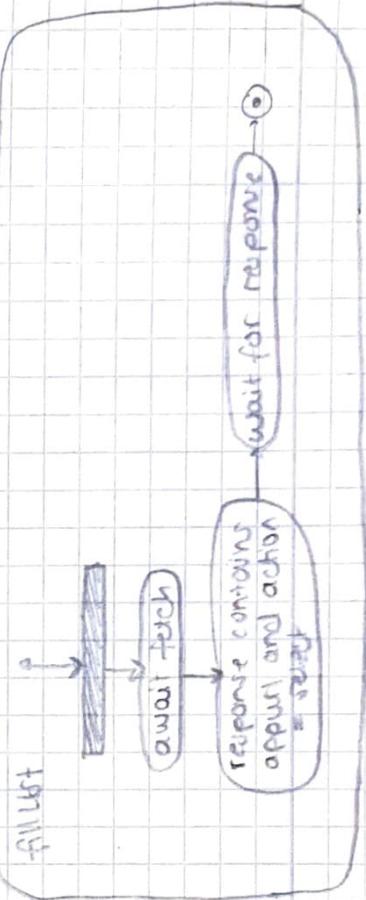
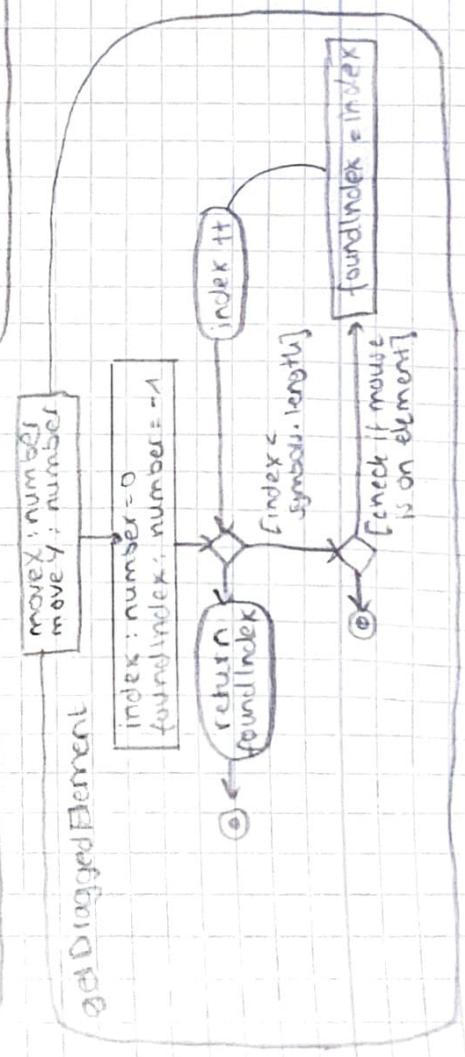
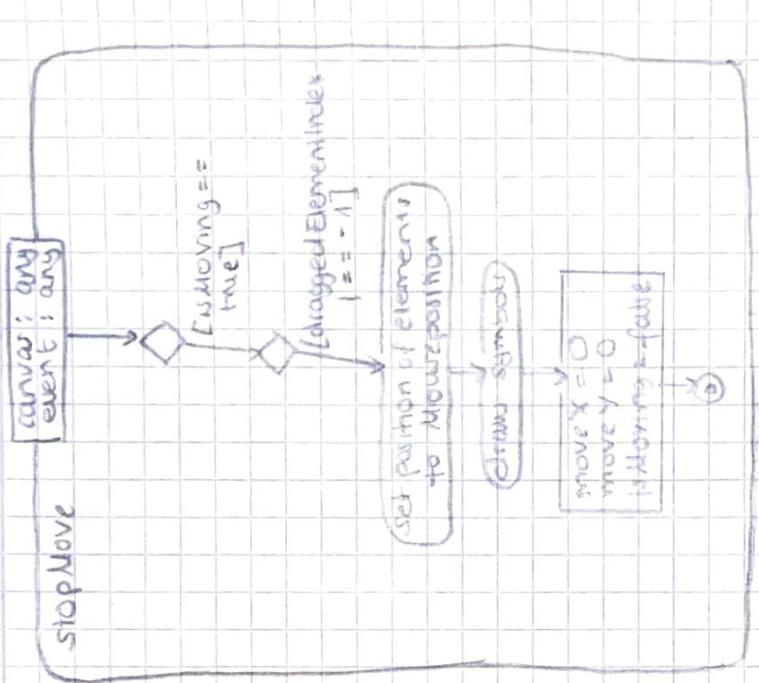
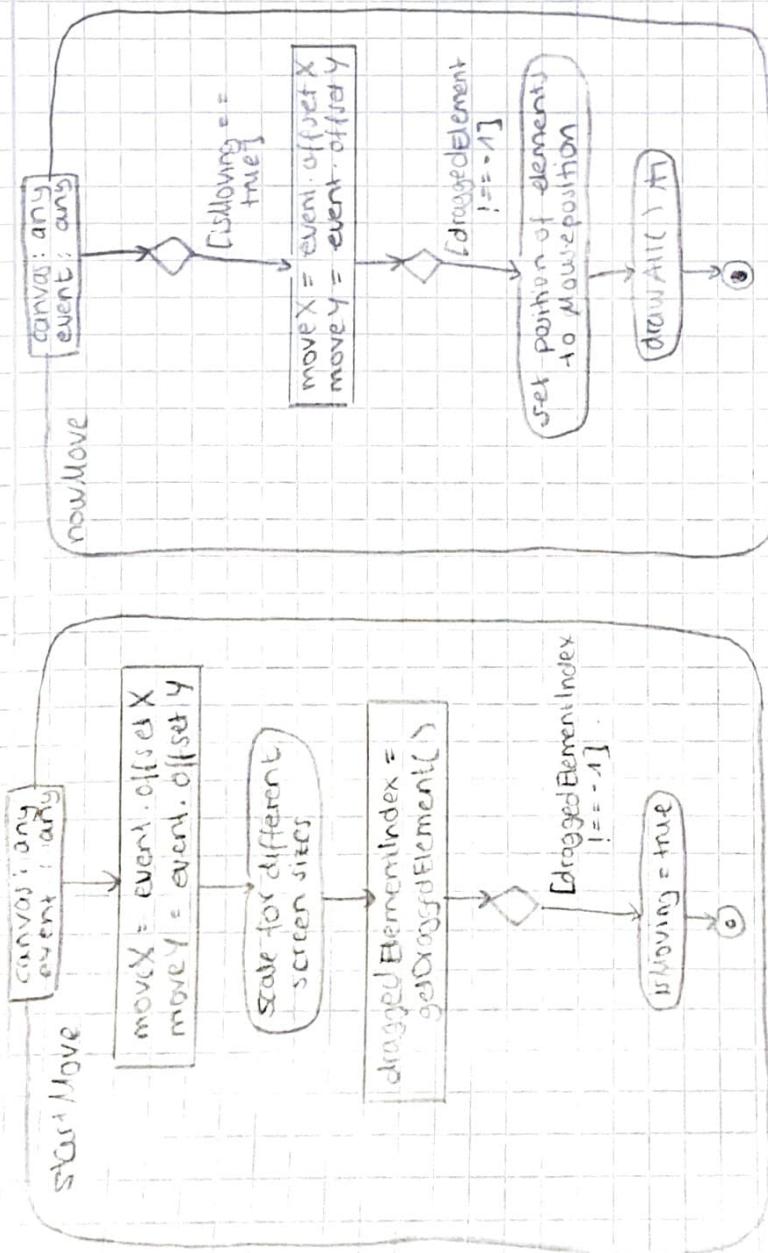
MagicCanvas

- ActivityDiagram
- Main(1)



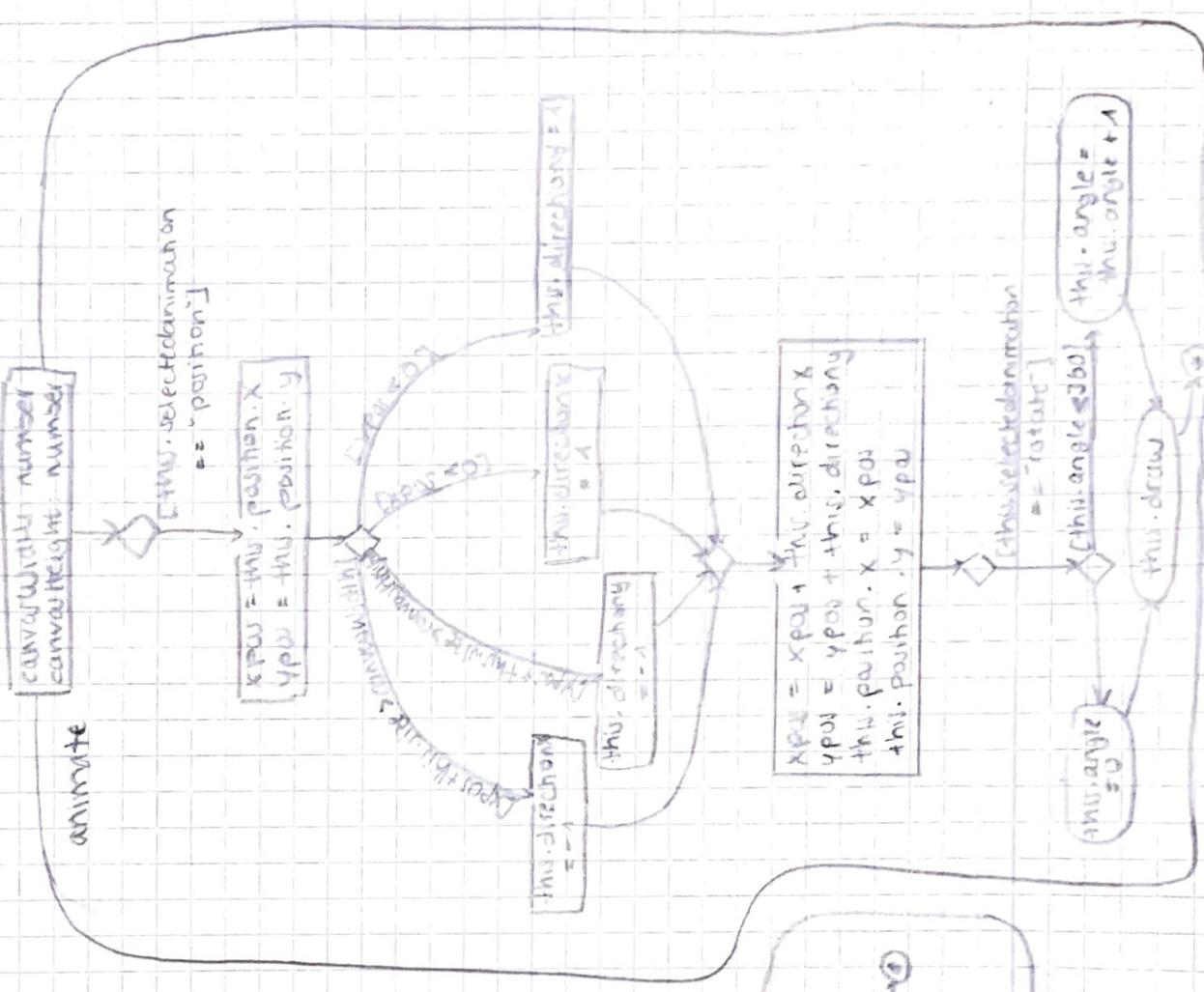
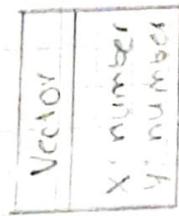
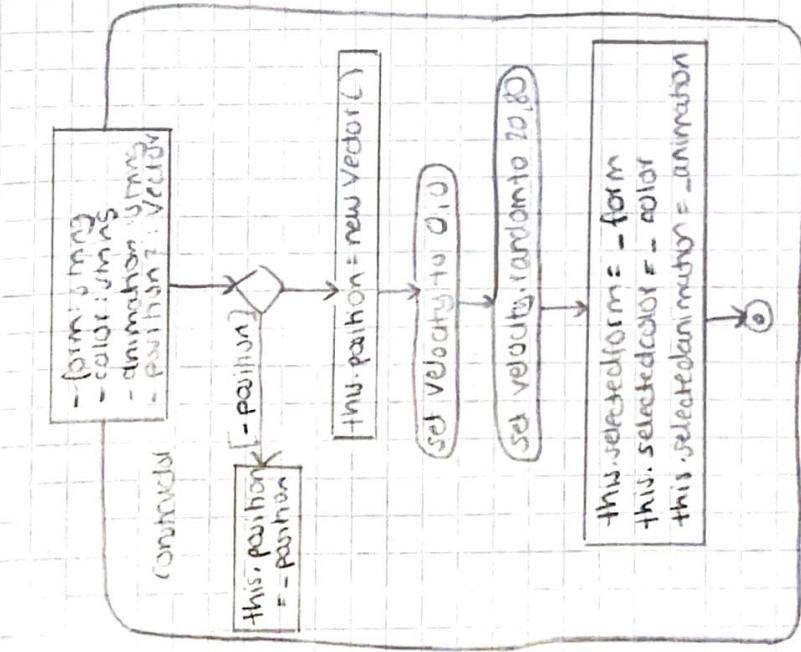
MagicCanvas

- Activity Diagram
- Main(5)



MagicCanvas

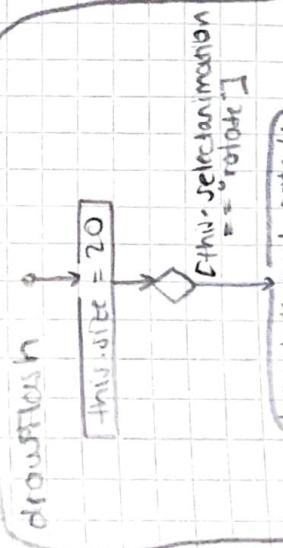
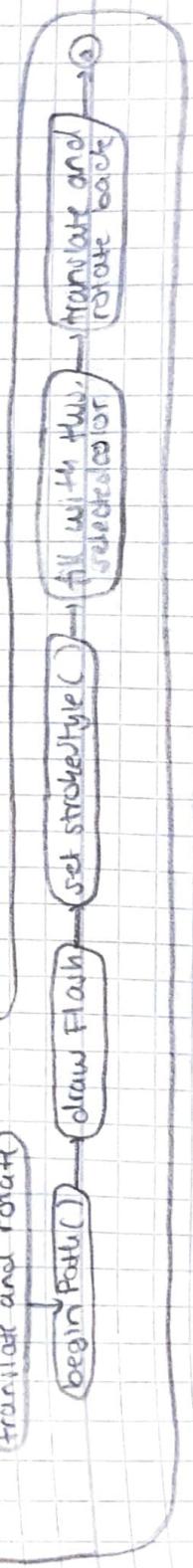
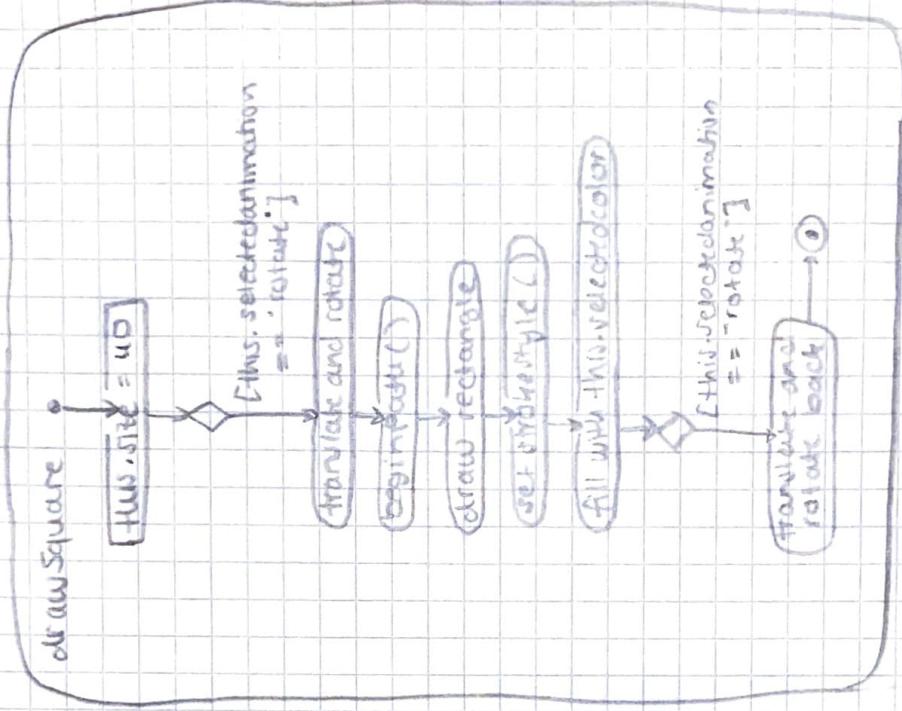
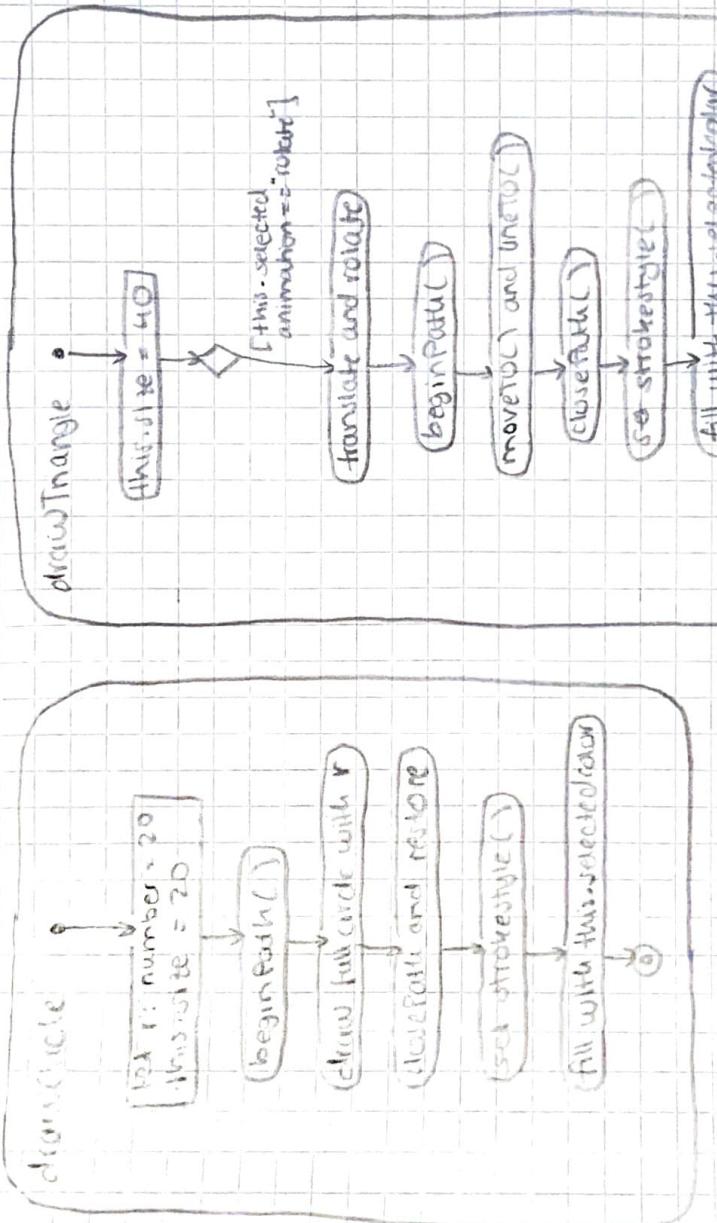
- ActivityDiagram
- canvasElements(A)



MainCanvas

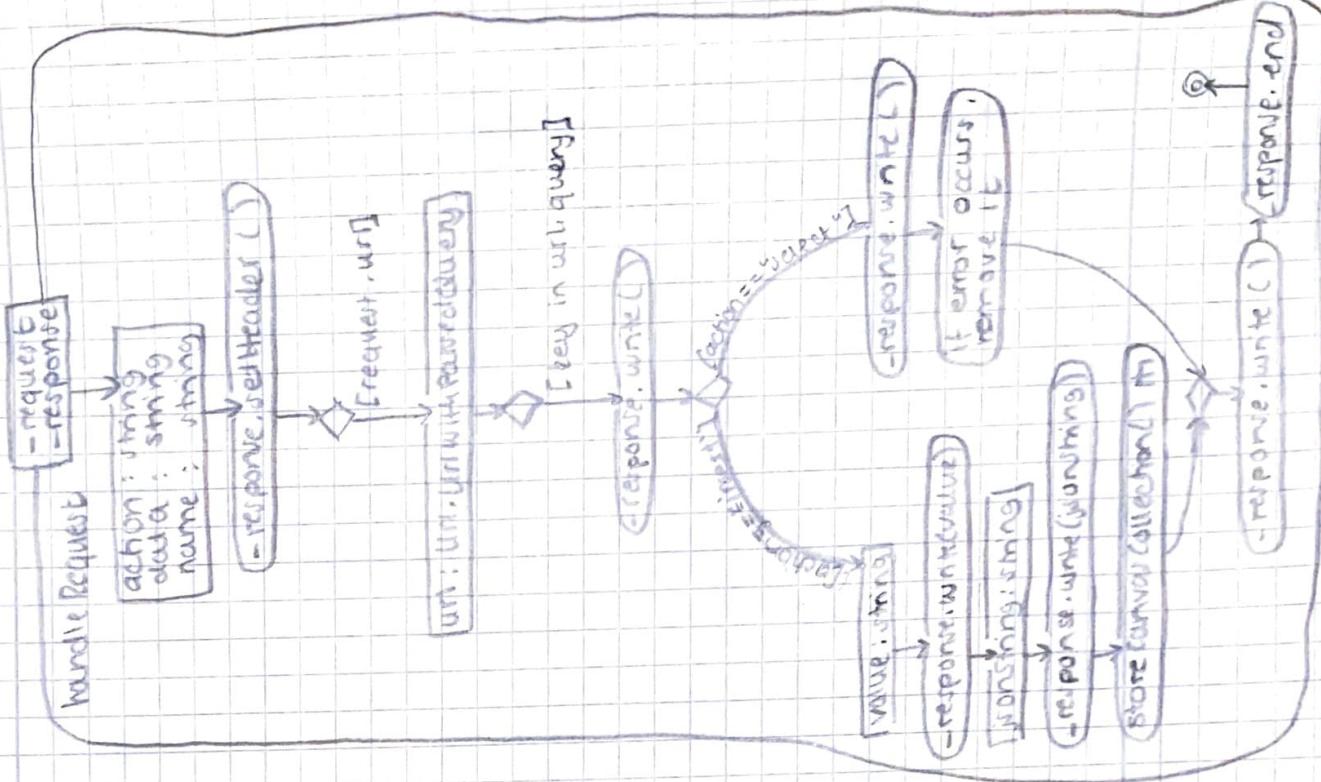
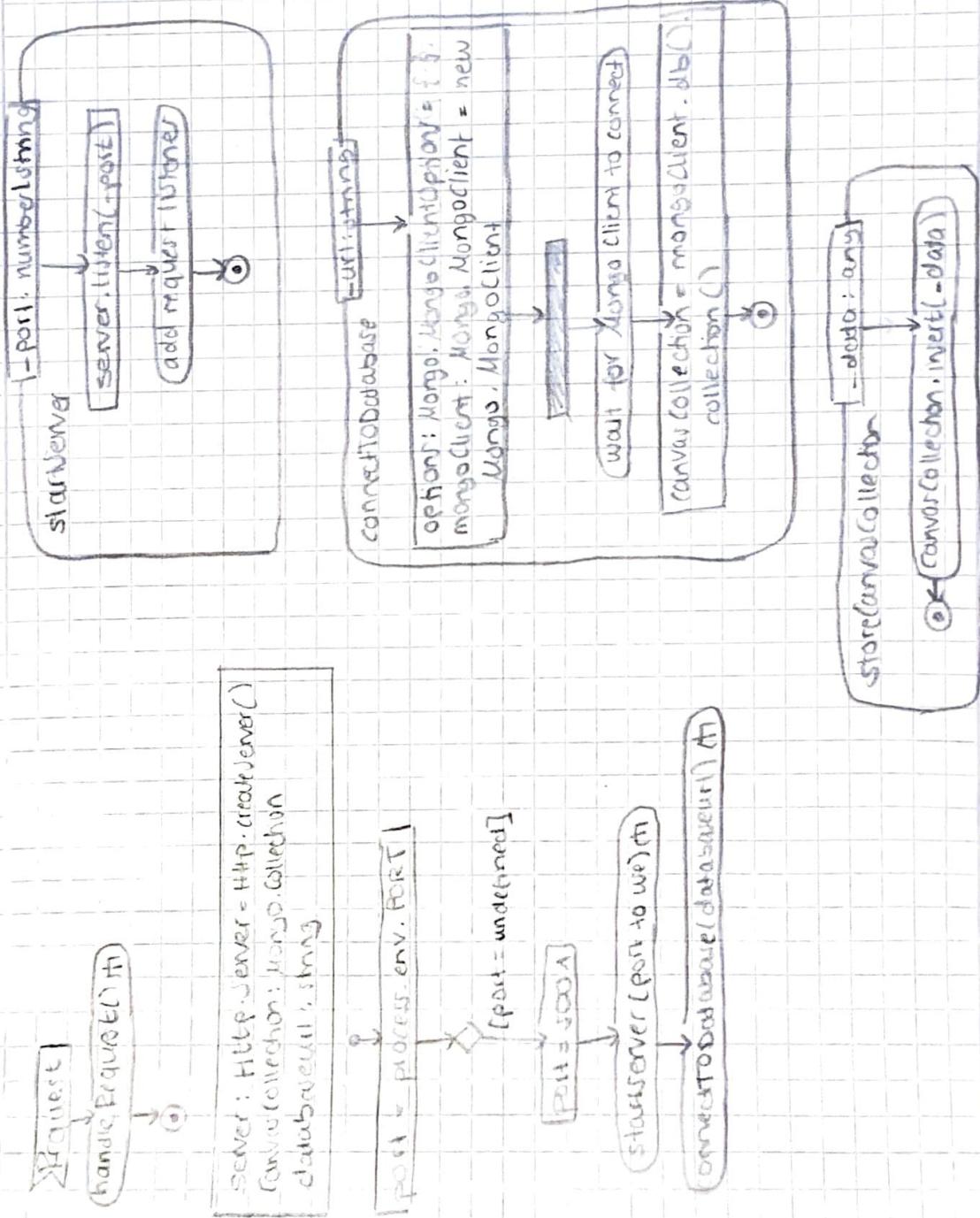
- Activity Diagram

- CanvasElement (2)



MagicCanvas

- Activity Diagram
- Server



Magic Canvas

-Swimmers Lane

