Funktionale Analyse

Plattform ist der PC, da man somit eine Steuerung des Spielfisches durch wasd ermöglicht.

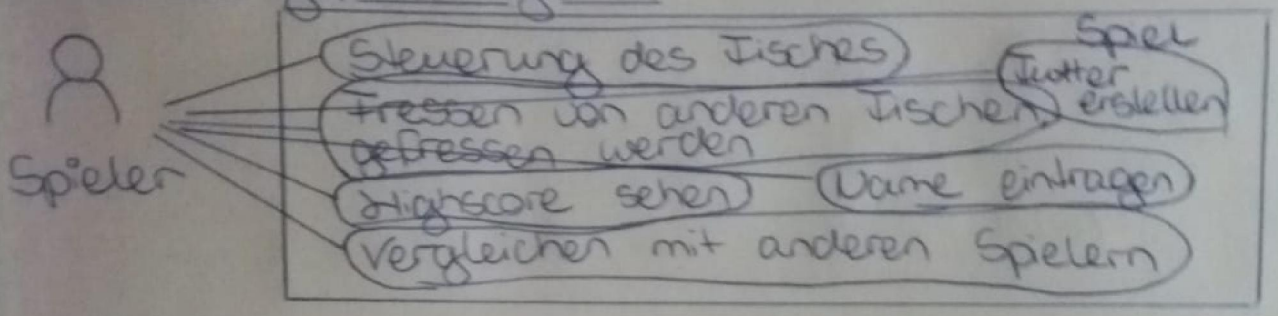
HTML Skizze

canvas
1500px • 850px
statisch

highscore
 mit id

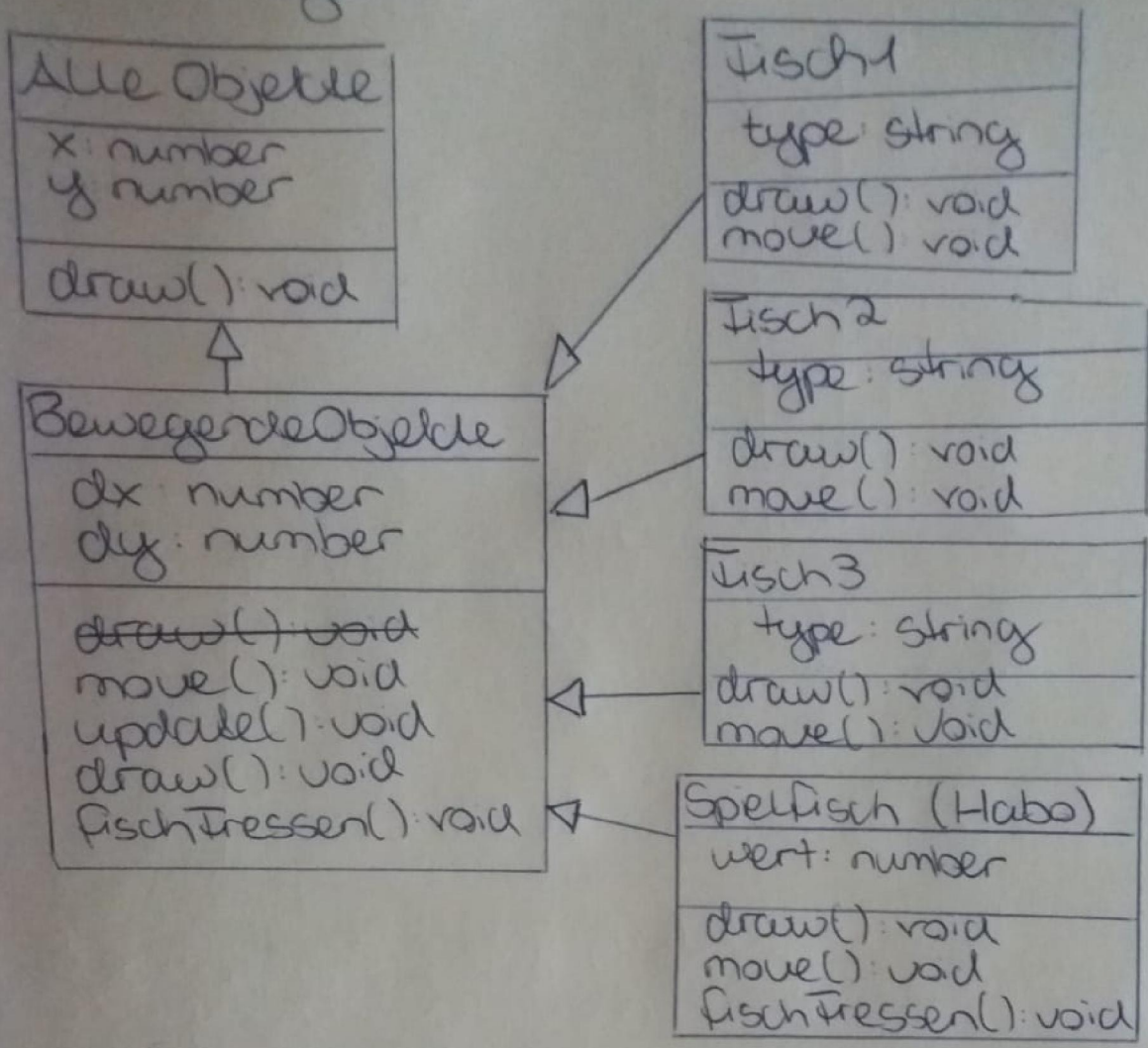
Andere Spieler
 mit id

Anwendungsfalldiagramm



* Bewegung erfolgt mittels wasd
* Hauptfisch/Spielfisch kann kleinere Fische fressen
* Bei Spielende kann der Spieler seinen Namen eintragen
* Spieler wird kleiner bei bestimmten Fischen (Kugelfisch) und verliert Punkte
* Spieler stirbt bei bestimmten Fisch berührungen mit Qualle
* Futter durch klicken erstellen

Klassendiagramm



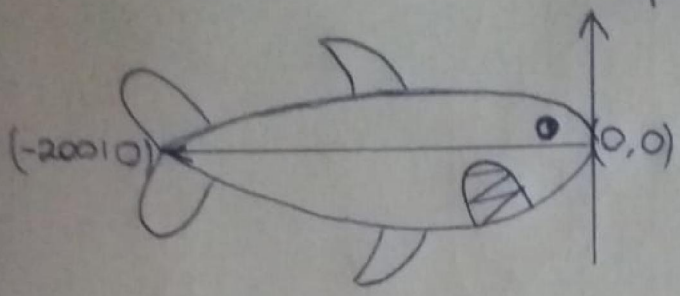
Interaktionsmöglichkeiten

- ×Spielfisch frisst kleinen Tisch
- ×Punktzahl erhöht sich
- ×kleiner Tisch wird entfernt und neuer hinzugefügt
- ×Spielerfisch wächst

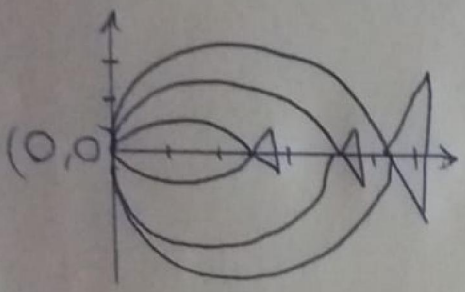
- ×Spiel beendet sobald ich versuche einen großen Tisch zu fressen
- ×Spiel hält an, ich gebe meinen Namen ein und sehe meinen Highscore

- ×Spiel endet bei Berührung mit Qualle
- ×beenden

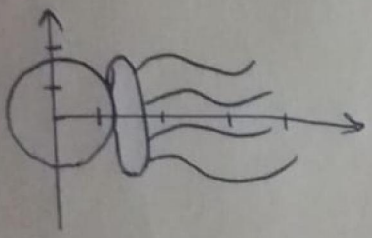
Zeichnungen Genauen Maße werden später entschieden



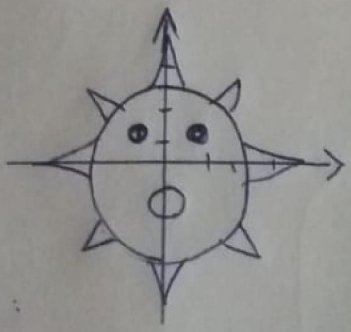
Spielerfisch



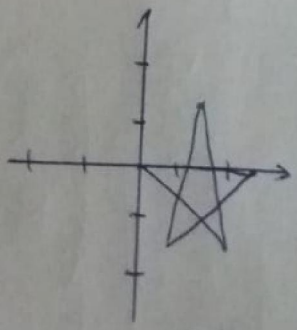
3 zu fressende Fische



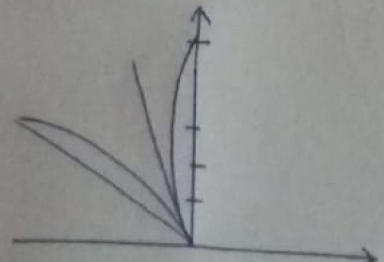
Qualle



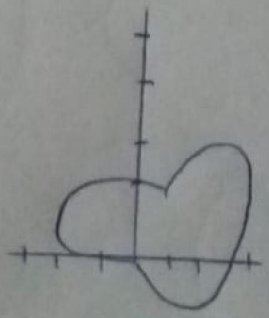
Kugelfisch



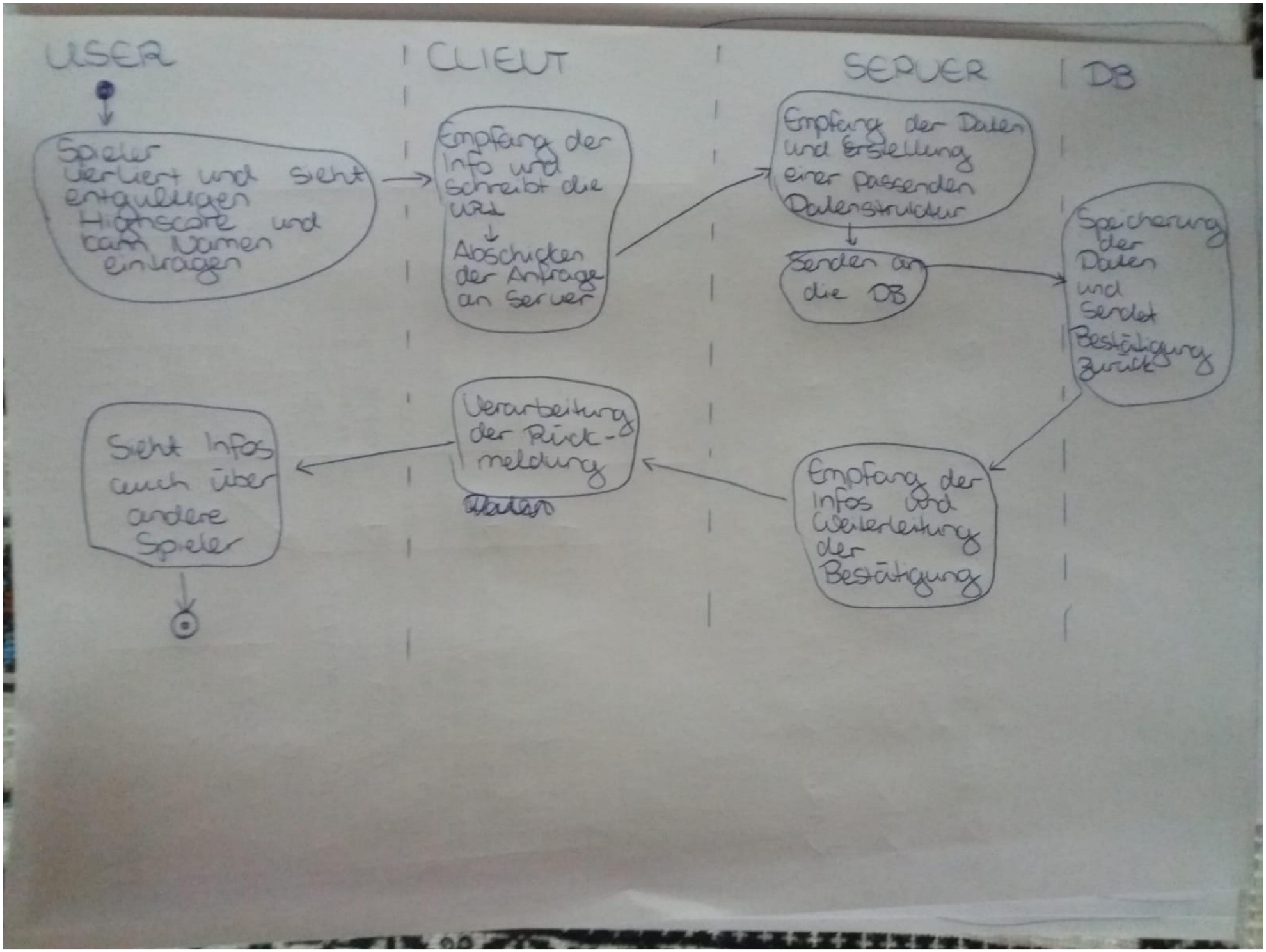
Stern



Pflanze

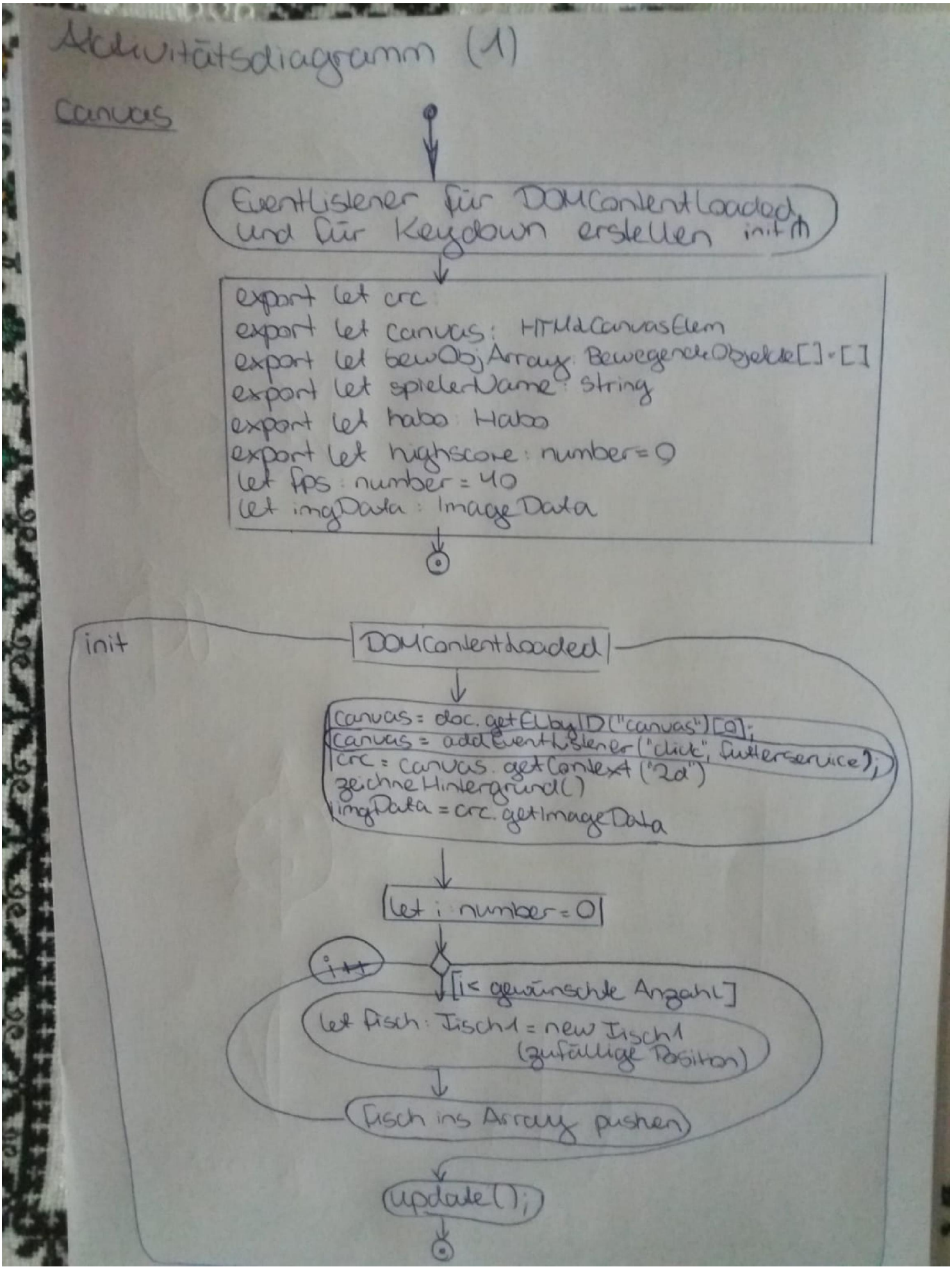


Stein

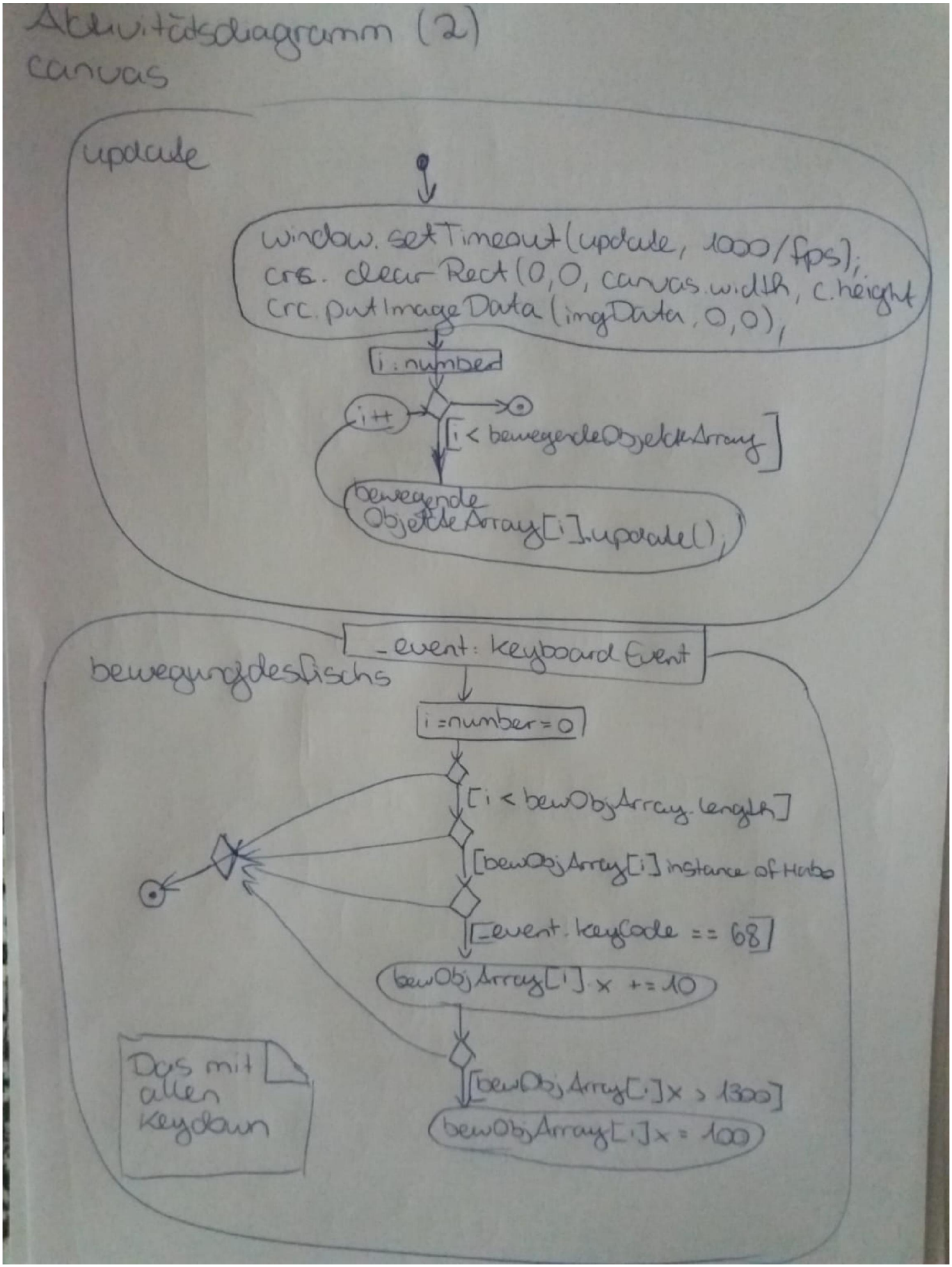

USER
CLIENT
SERVER
DB
Spieler verliert und sieht entgültigen Highscore und kann Namen eintragen
Empfang der Info und schreibt die URL
↓
Abschicken der Antrage an Server
Empfang der Daten und Erstellung einer passenden Datenstruktur
Senden an die DB
Speicherung der Daten und sendet Bestätigung zurück
Sieht Infos auch über andere Spieler
Verarbeitung der Rück-meldung
Empfang der Infos und Weiterleitung der Bestätigung

Aktivitätsdiagramm (1)

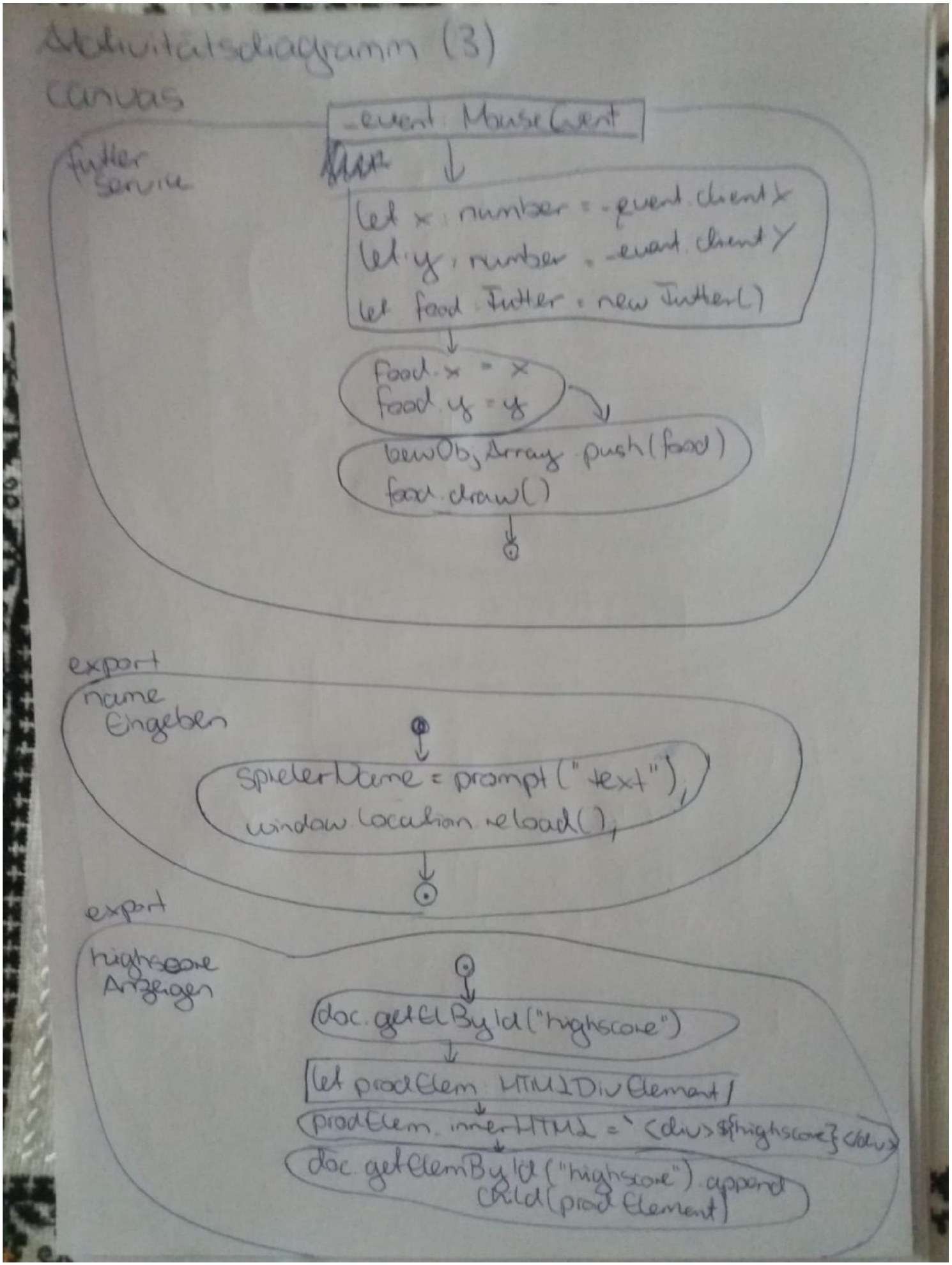
Canvas



Aktivitätsdiagramm (2)
canvas



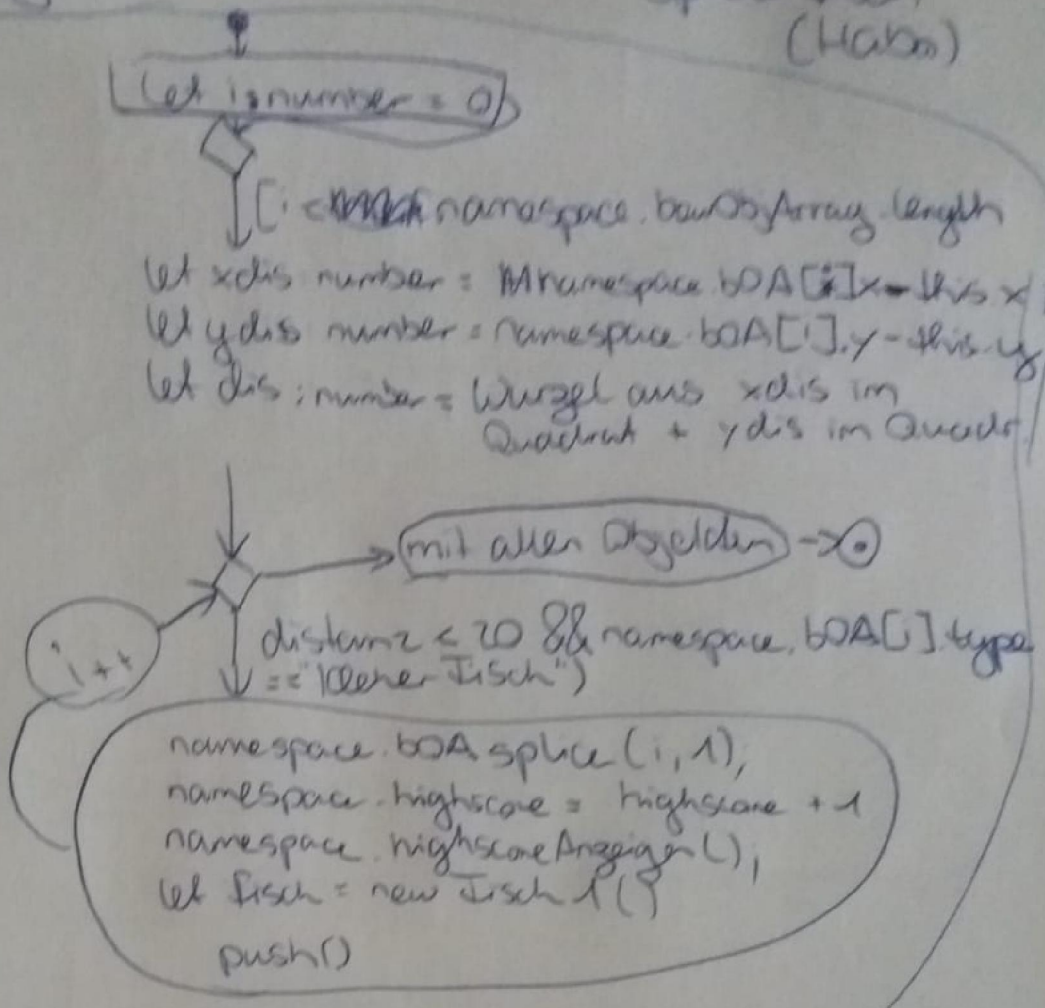
Aktivitätsdiagramm (3)



Aktivitätsdiagramm (4)

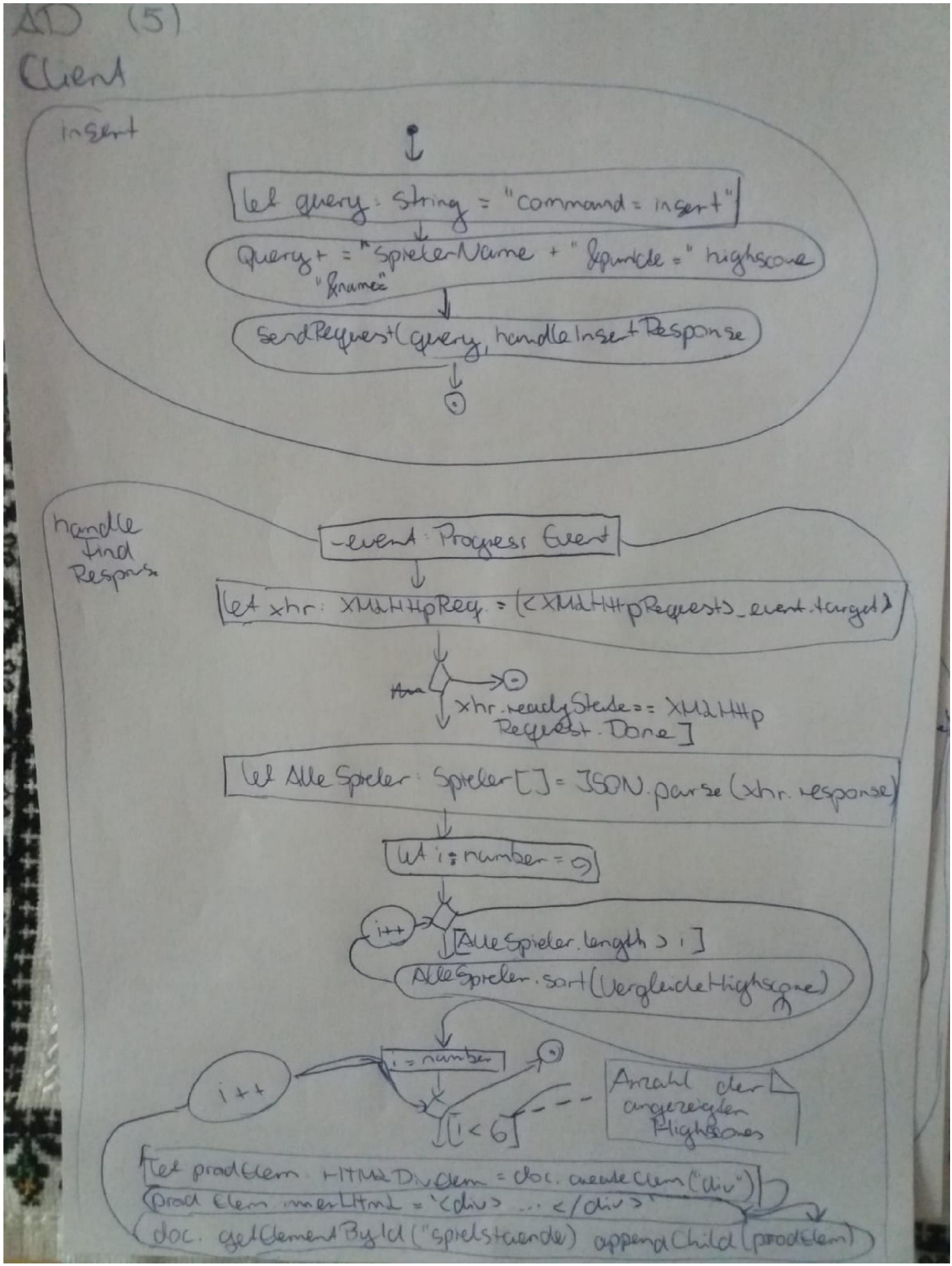
Spielerfisch (Habs)

Fischfressen



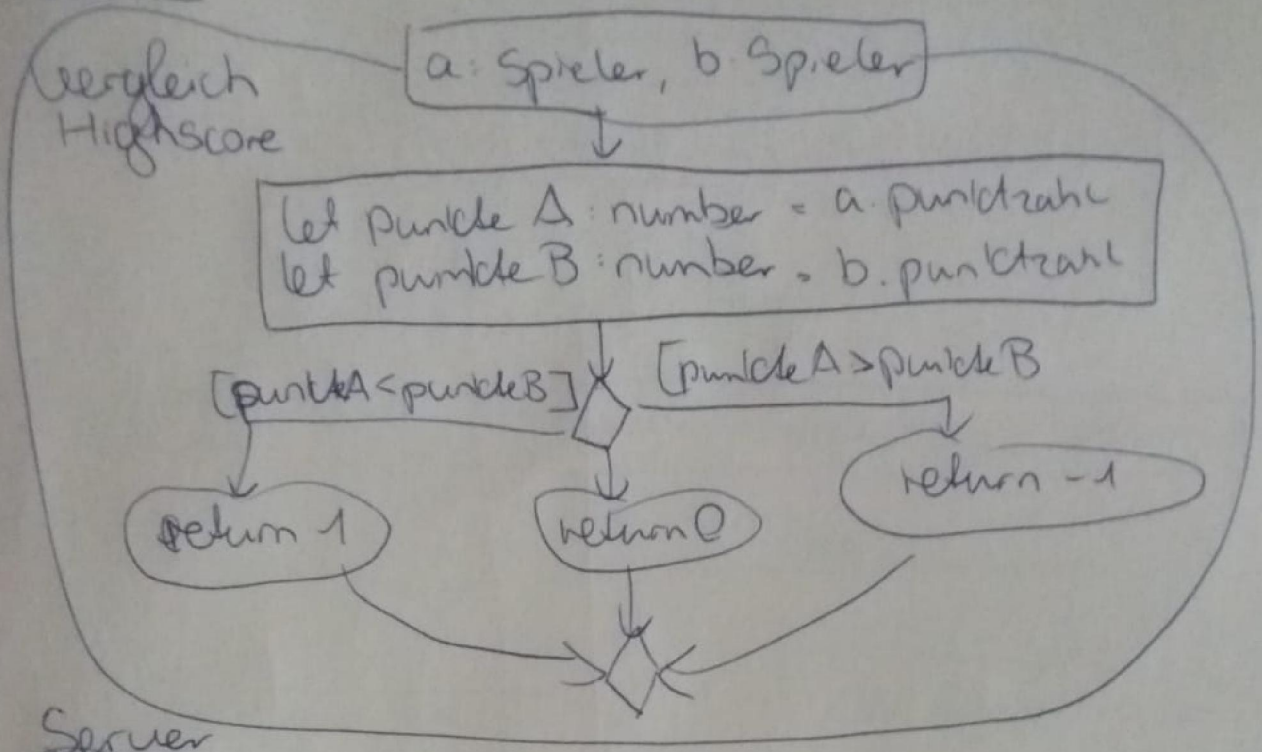
AD (5)

Client

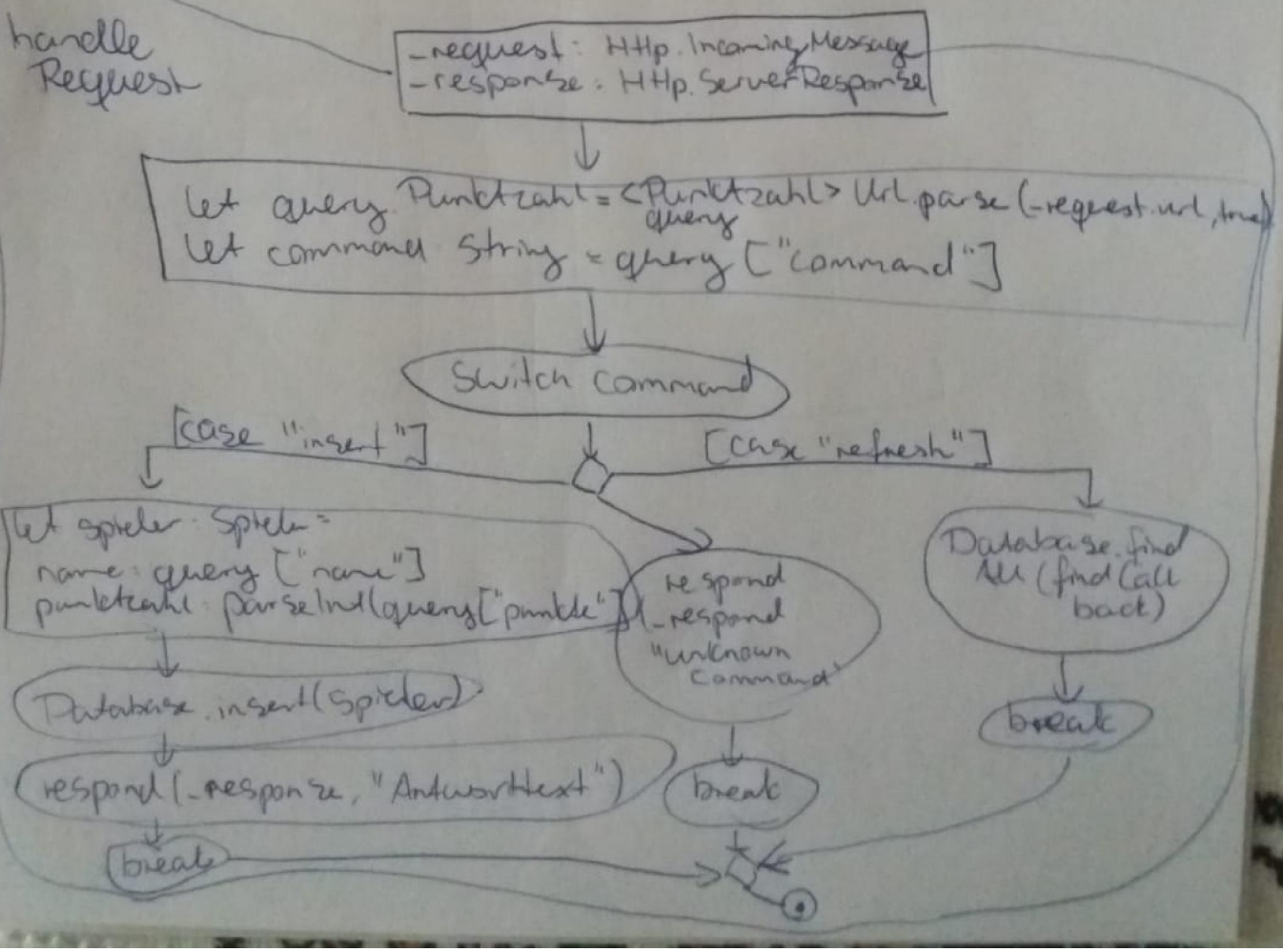

insert
let query: String = "command = insert"
query += "spielerName + "&punkte =" highscore "&name="
sendRequest(query, handleInsertResponse
handle find Respons
event: ProgressEvent
let xhr: XMLHttpReq. = (<XMLHttpRequest>_event.target)
xhr.readyState == XMLHttp Request.Done]
let AlleSpieler: Spieler[] = JSON.parse(xhr.response)
let i: number = 0
i++
AlleSpieler.length > i]
AlleSpieler.sort(VergleicheHighscore)
i = number
i++
i < 6]
Anzahl der angezeigten Highscores
let prodElem: HTMLDivElem = doc.createElem("div")
prodElem.innerHtml = ' ... '
doc.getElementById("spielstaende) appendChild(prodElem)

AD (6)

Client



Server



AD (7) Database