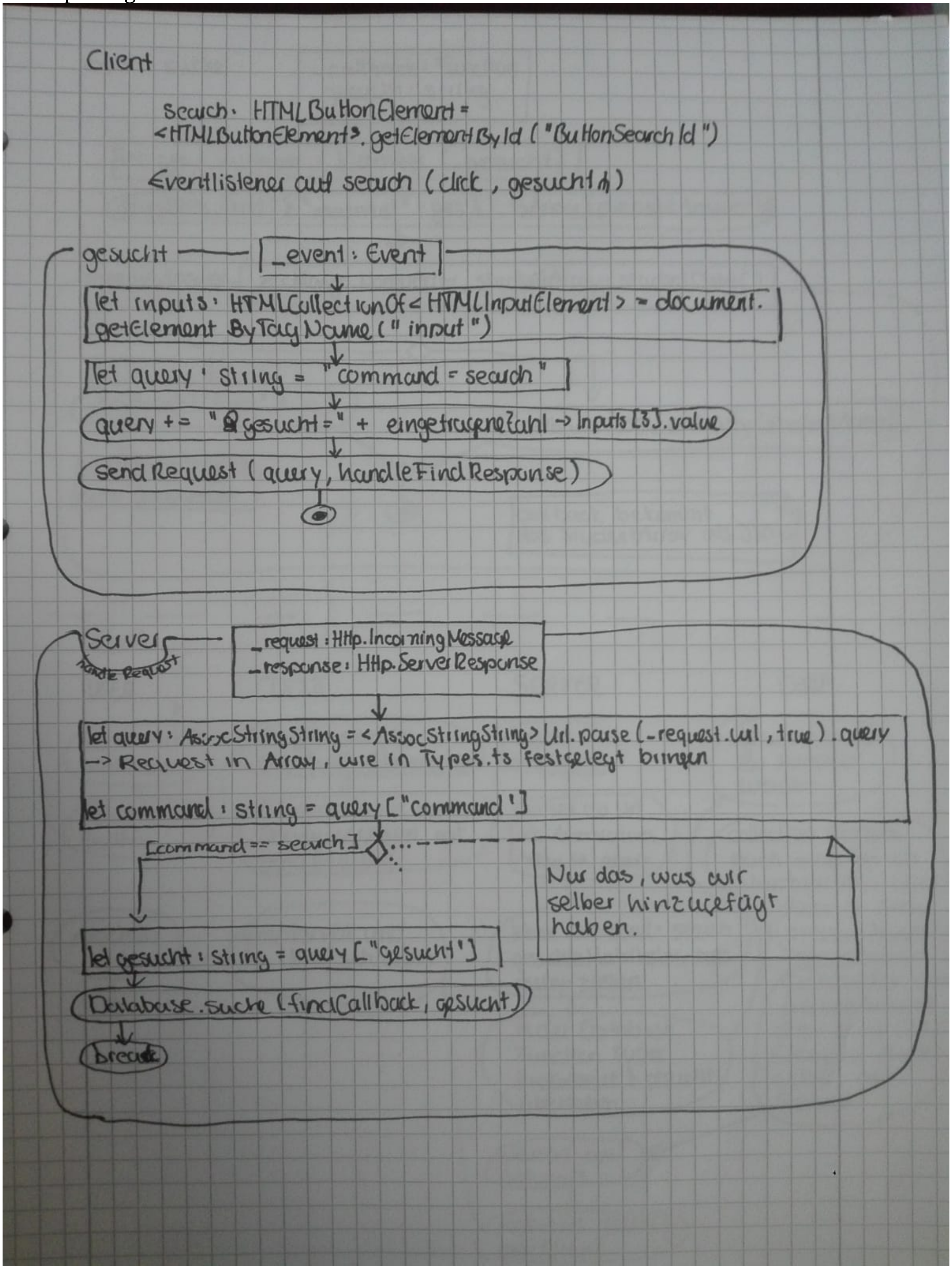
Konzept Aufgabe 9 Überarbeitet

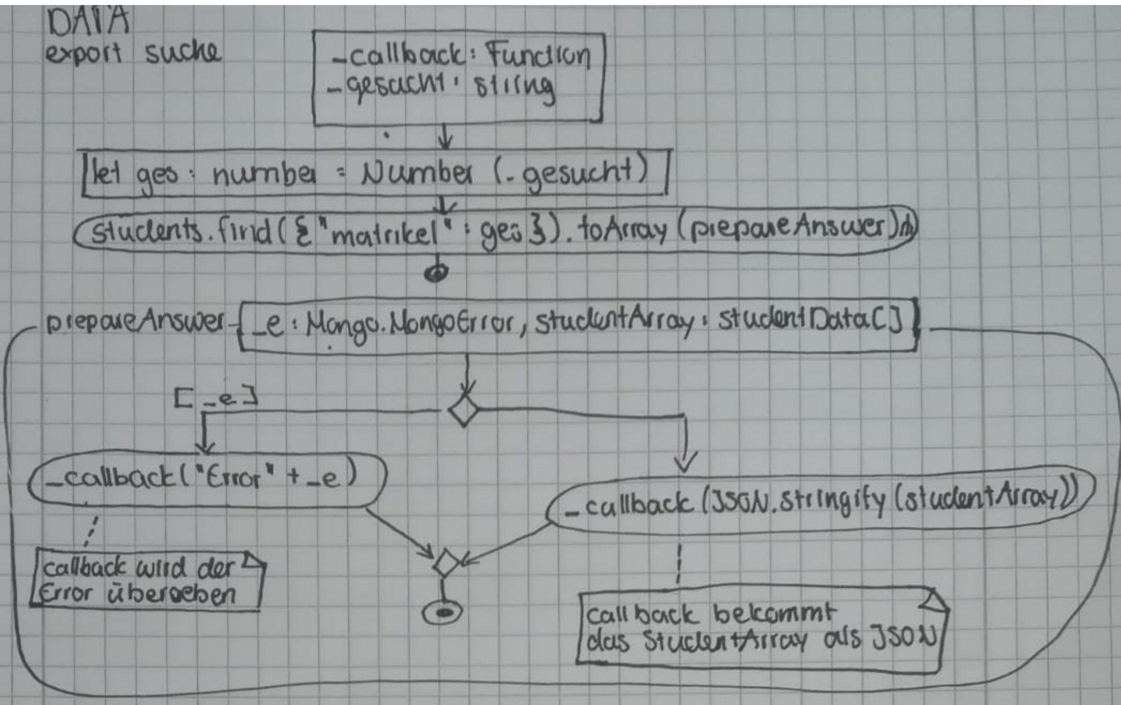


```
search: HTMLButtonElement =
<HTMLButtonElement>.getElementById ("ButtonSearch Id")

Eventlistener auf search (click, gesucht)
```

```
_event: Event
let inputs: HTMLCollectionOf<HTMLInputElement> = document.
getElement ByTagName (" input ")
let query: string = "command = search"
query += "&gesucht=" + eingetrageneZahl -> Inputs [3].value
send Request (query, handleFindResponse)
```

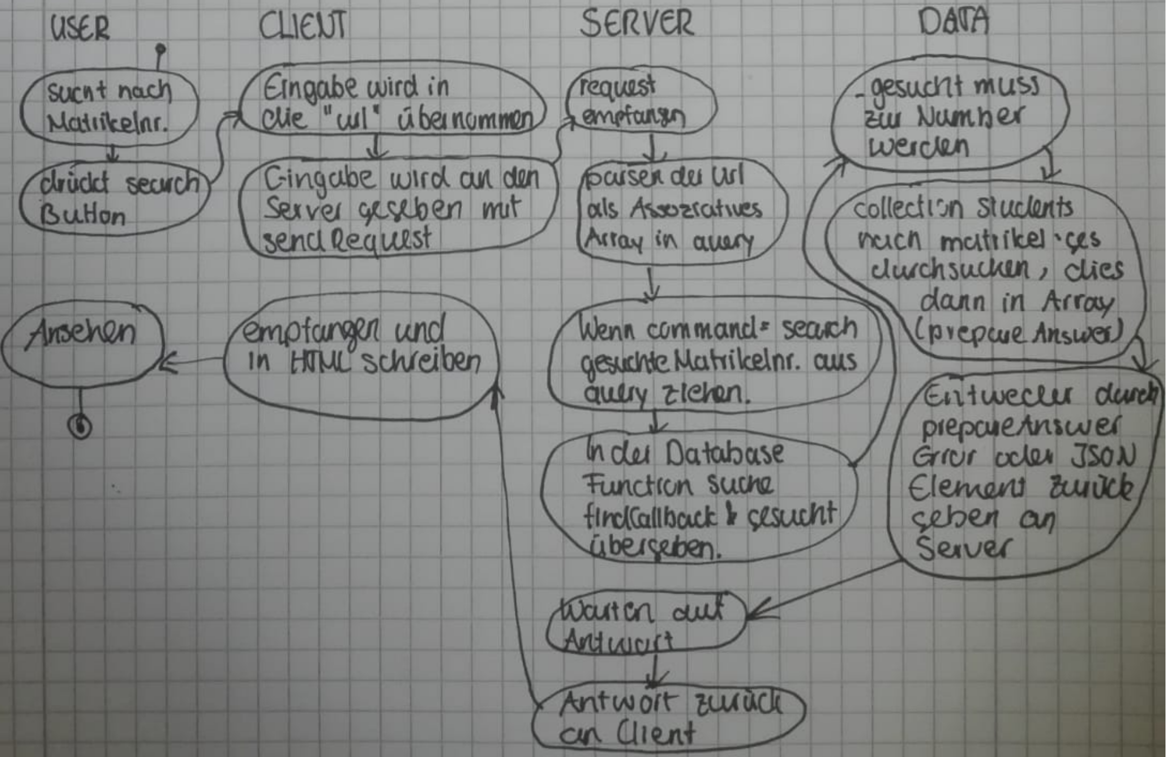
```
_request: Http.IncomingMessage
_response: Http.ServerResponse
let query: AssocStringString = <AssocStringString> Url.parse (_request.url, true).query
-> Request in Array, wie in Types.ts festgelegt bringen

let command: string = query ["command"]
let gesucht: string = query ["gesucht"]
Database.suche (findCallback, gesucht)
break
```

DATA
export suche



_callback: Function
_gesucht: string
let ges: number = Number (_gesucht)
students.find({"matrikel": ges}).toArray(prepareAnswer);
prepareAnswer
_e: Mongo.MongoError, studentArray: studentData[]
[_e]
_callback("Error" + _e)
callback wird der Error übergeben
_callback (JSON.stringify(studentArray))
callback bekommt das StudentArray als JSON

(Zusatz)



USER
CLIENT
SERVER
DATA
sucht nach Matrikelnr.
drückt search Button
Eingabe wird in die "url" übernommen
Eingabe wird an den Server gegeben mit sendRequest
request empfangen
parsen der Url als Assoziatives Array in query
Wenn command = search gesuchte Matrikelnr. aus query ziehen.
In der Database Function suche findCallback & gesucht übergeben.
_gesucht muss zur Number werden
collection students nach matrikel: ges durchsuchen, dies dann in Array (prepareAnswer)
Entweder durch prepareAnswer Error oder JSON Element zurück geben an Server
Warten auf Antwort
Antwort zurück an Client
empfangen und in HTML schreiben
Ansehen