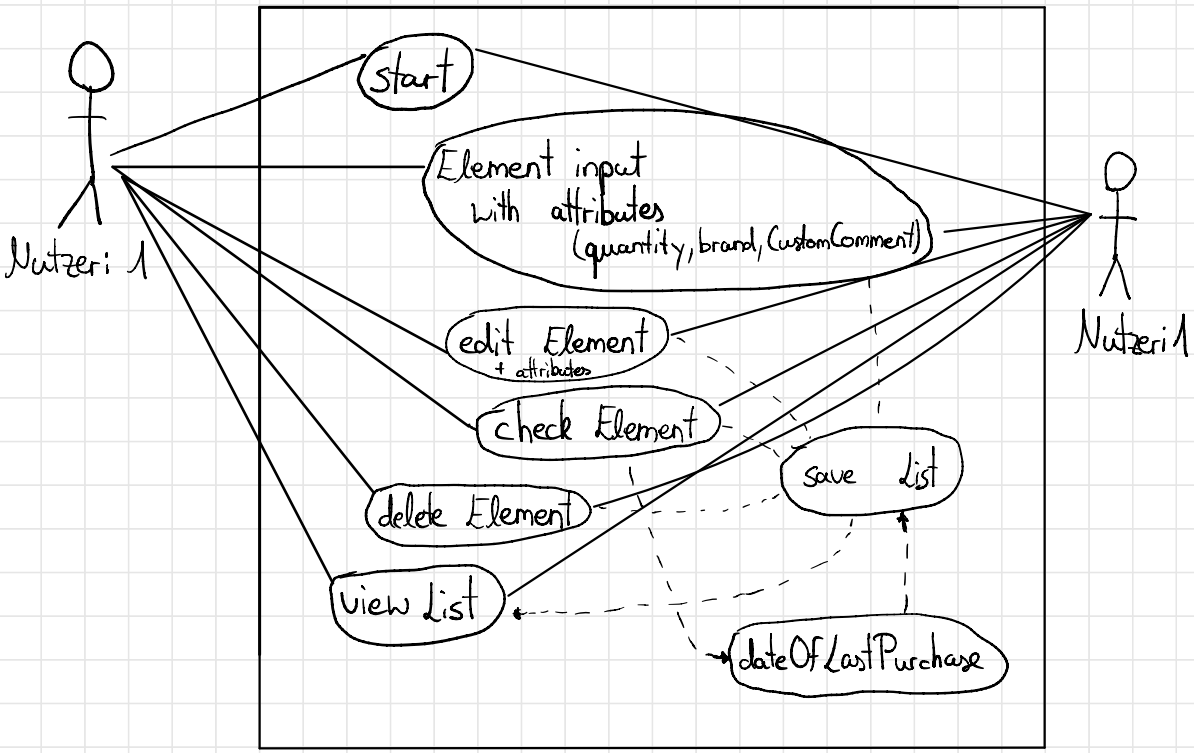
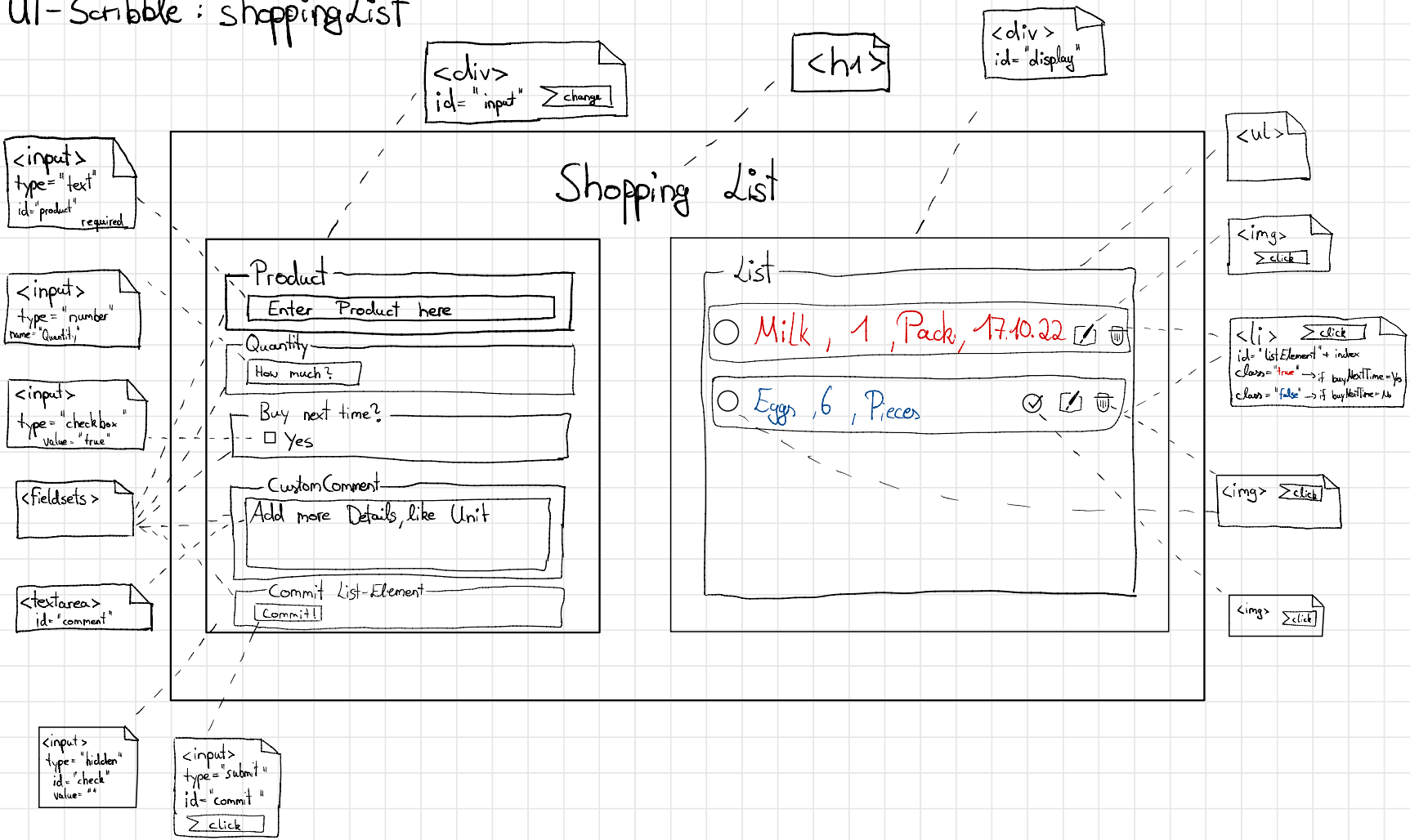
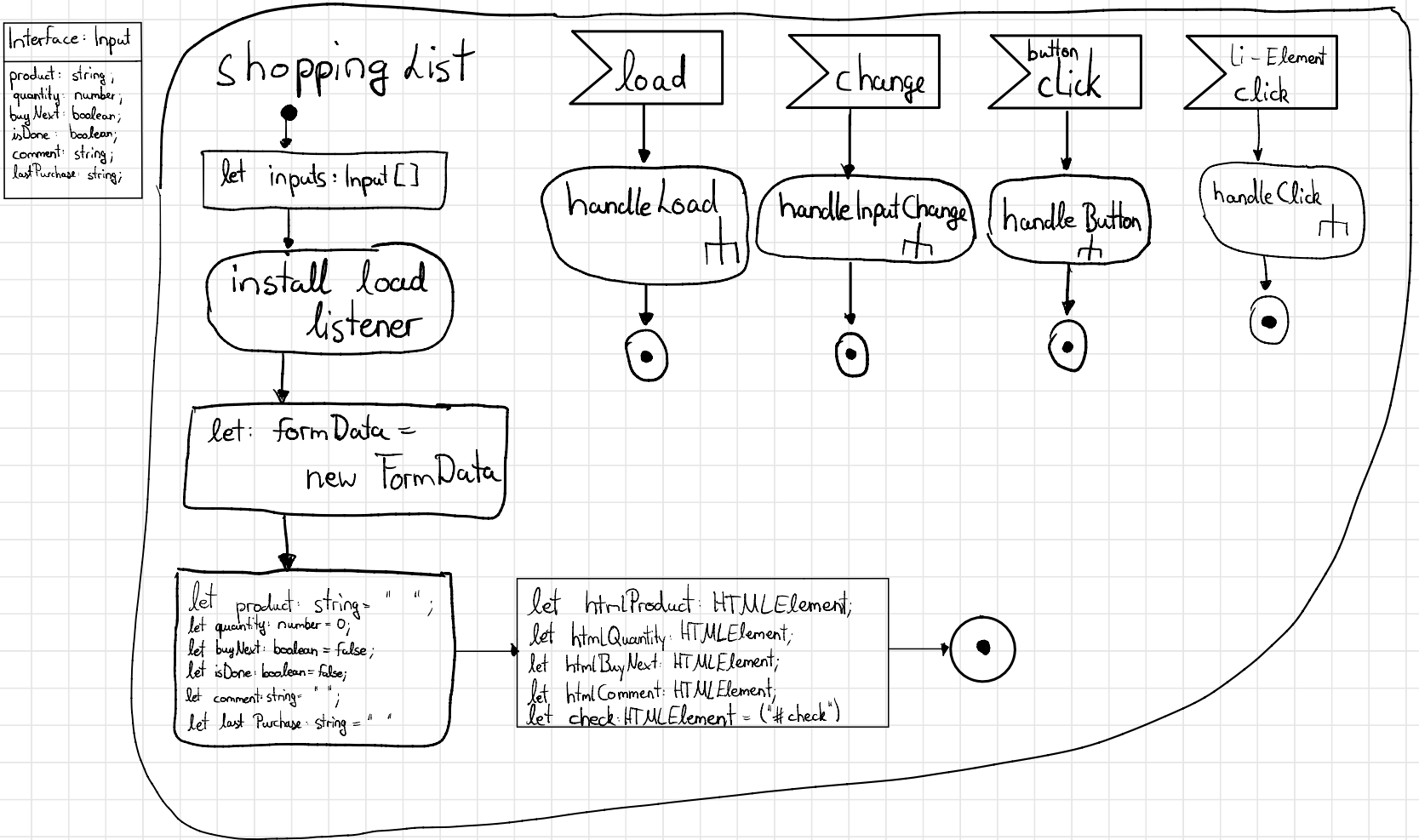
Use-Case-Diagramm: shoppingList

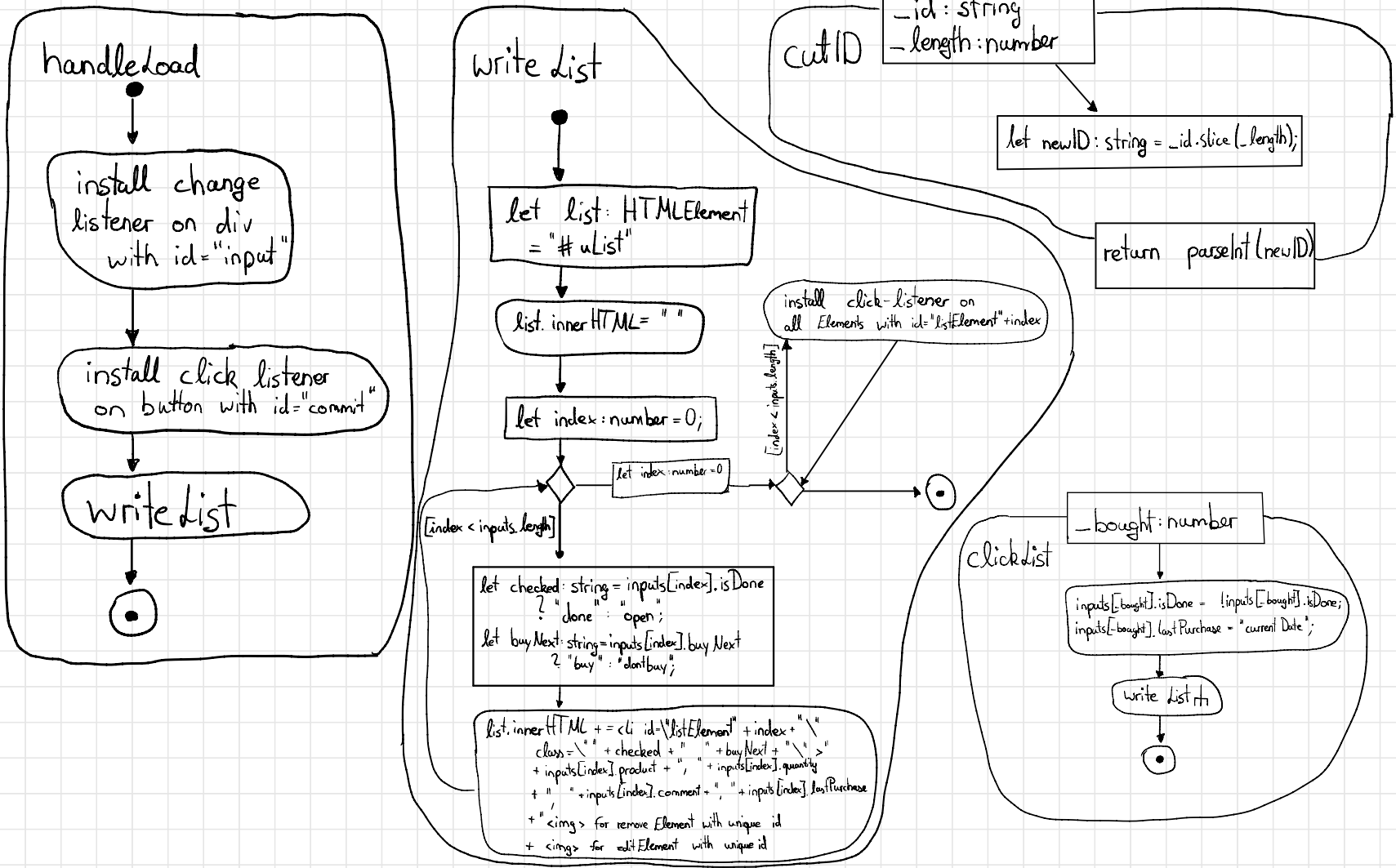


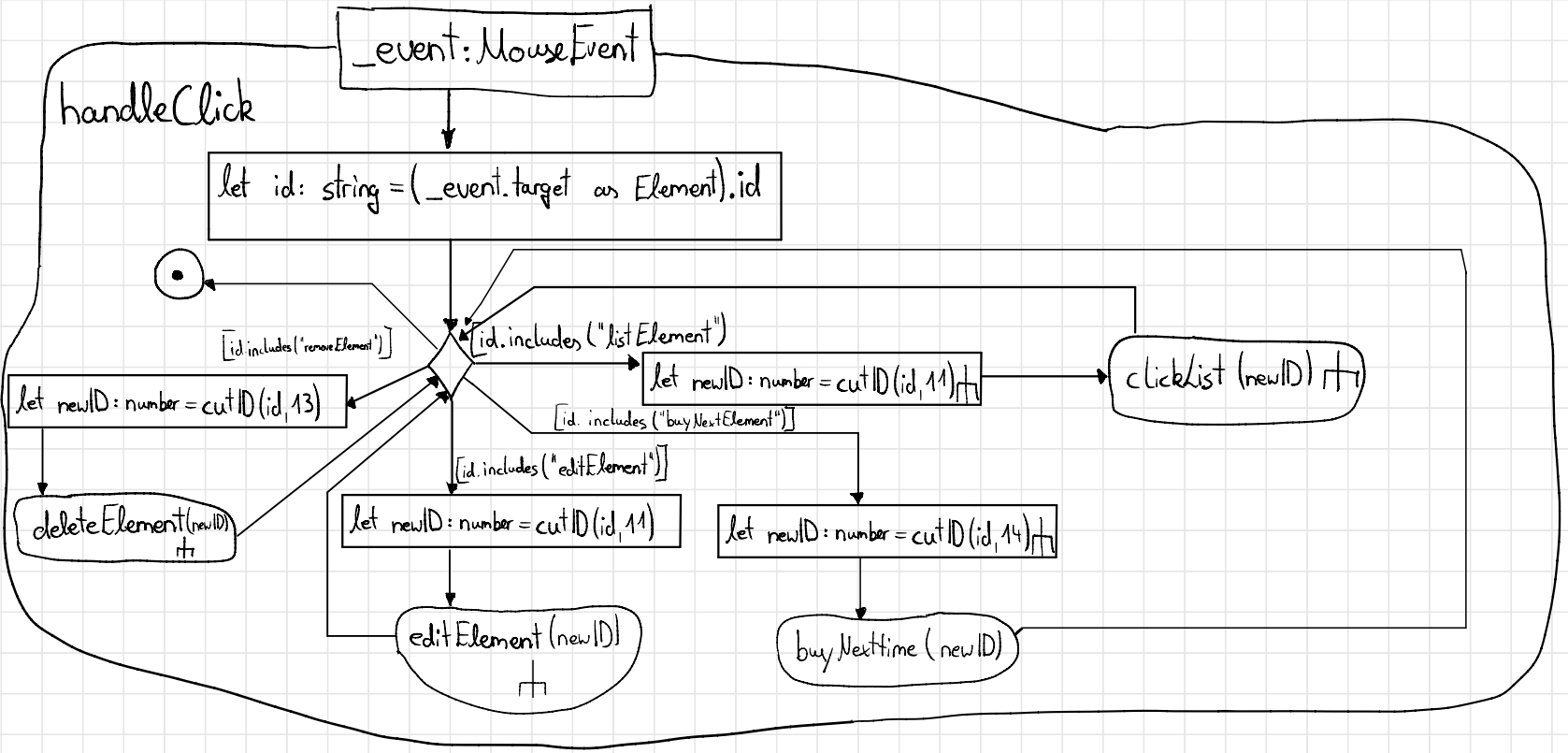
UI-Scribble : shoppingList

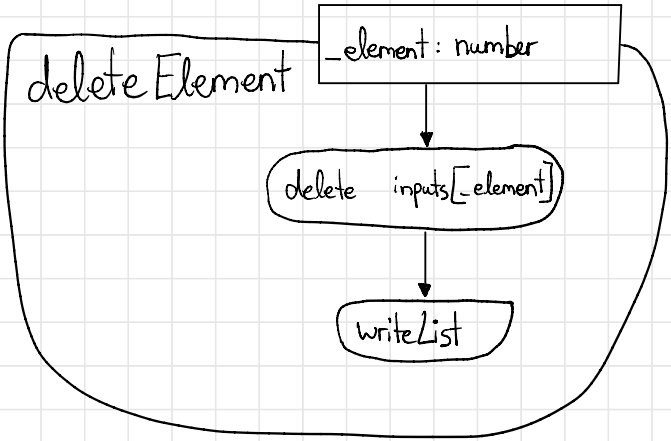


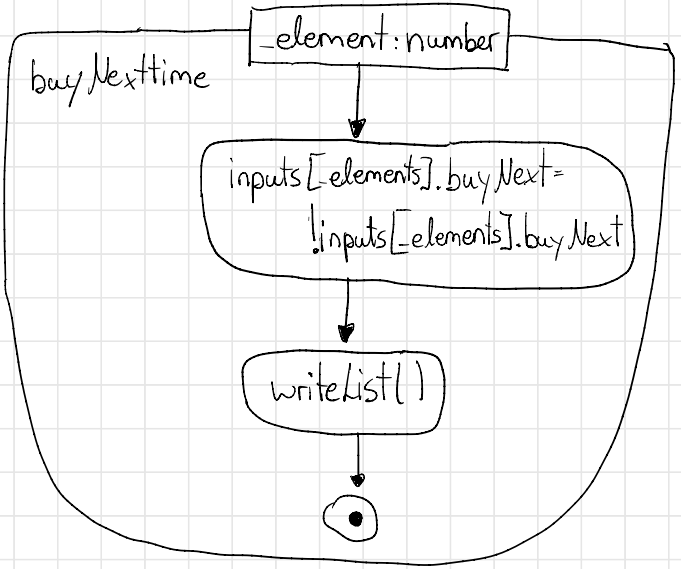
Activity Diagramm

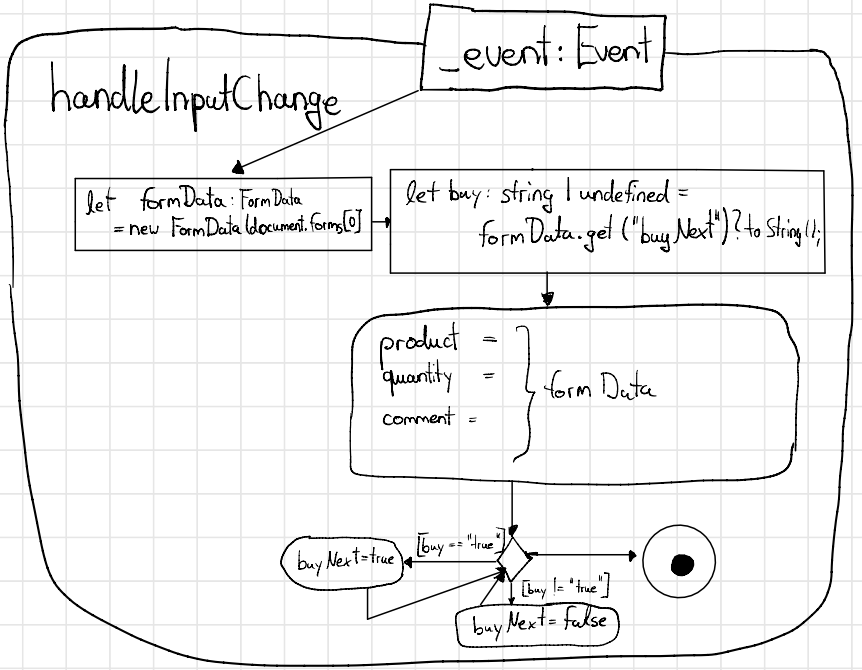


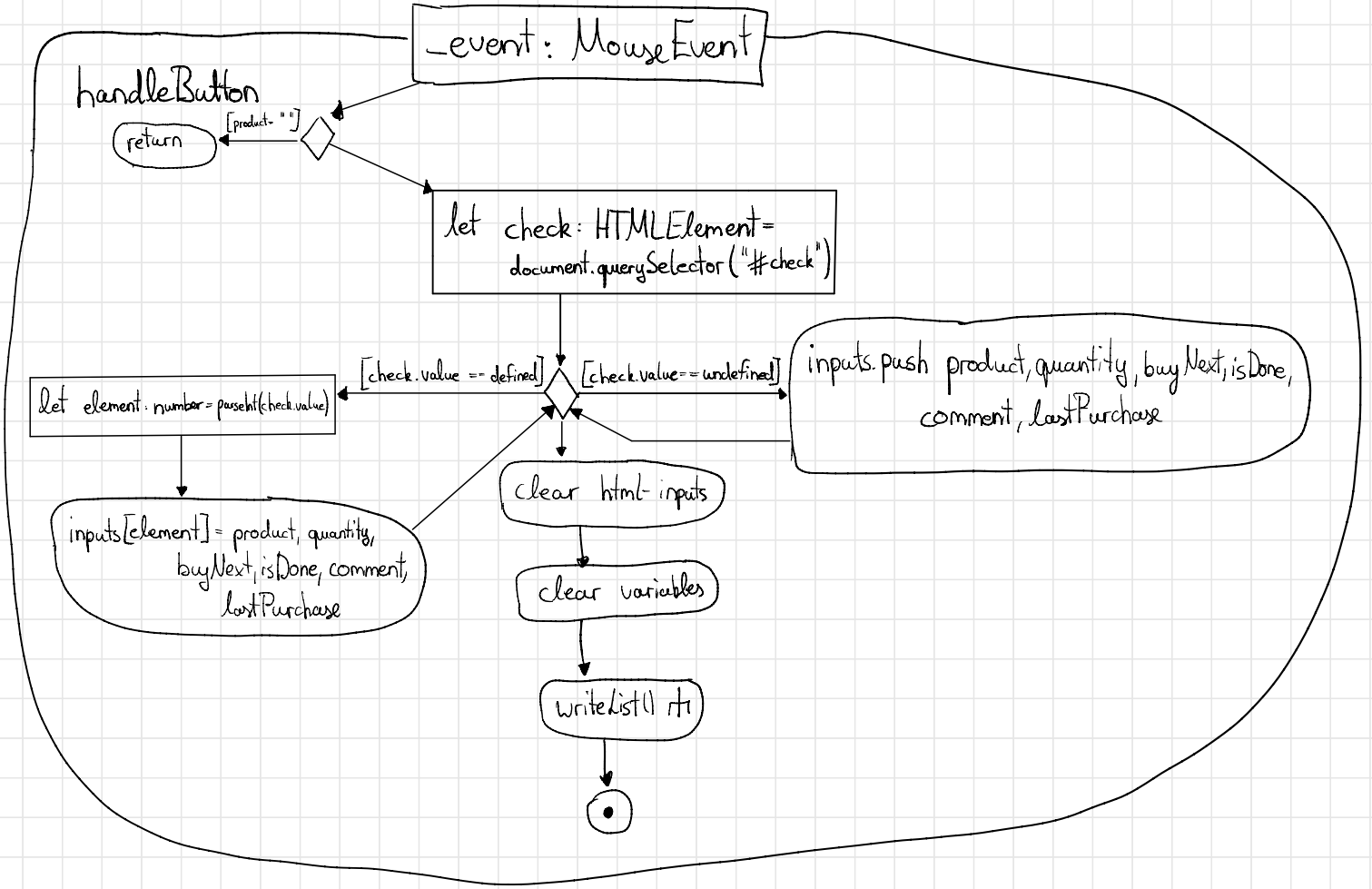

handleLoad
install change listener on div with id="input"
install click listener on button with id="commit"
writeList
writeList
let list: HTMLElement = "#uList"
list.innerHTML = ""
let index: number = 0;
[index < inputs.length]
let index: number = 0
let checked: string = inputs[index].isDone ? "done" : "open";
let buyNext: string = inputs[index].buyNext ? "buy" : "dontbuy";
list.innerHTML += <li id="listElement" + index + "\" class=\" + checked + " " + buyNext + "\">" + inputs[index].product + ", " + inputs[index].quantity + " " + inputs[index].comment + ", " + inputs[index].lastPurchase + ", " + <img> for remove Element with unique id + <img> for edit Element with unique id
[index < inputs.length]
install click-listener on all Elements with id="listElement"+index
cutID
_id: string
_length: number
let newID: string = _id.slice(_length);
return parseInt(newID);
clickList
_bought: number
inputs[_bought].isDone = !inputs[_bought].isDone;
inputs[_bought].lastPurchase = "currentDate";
writeList()

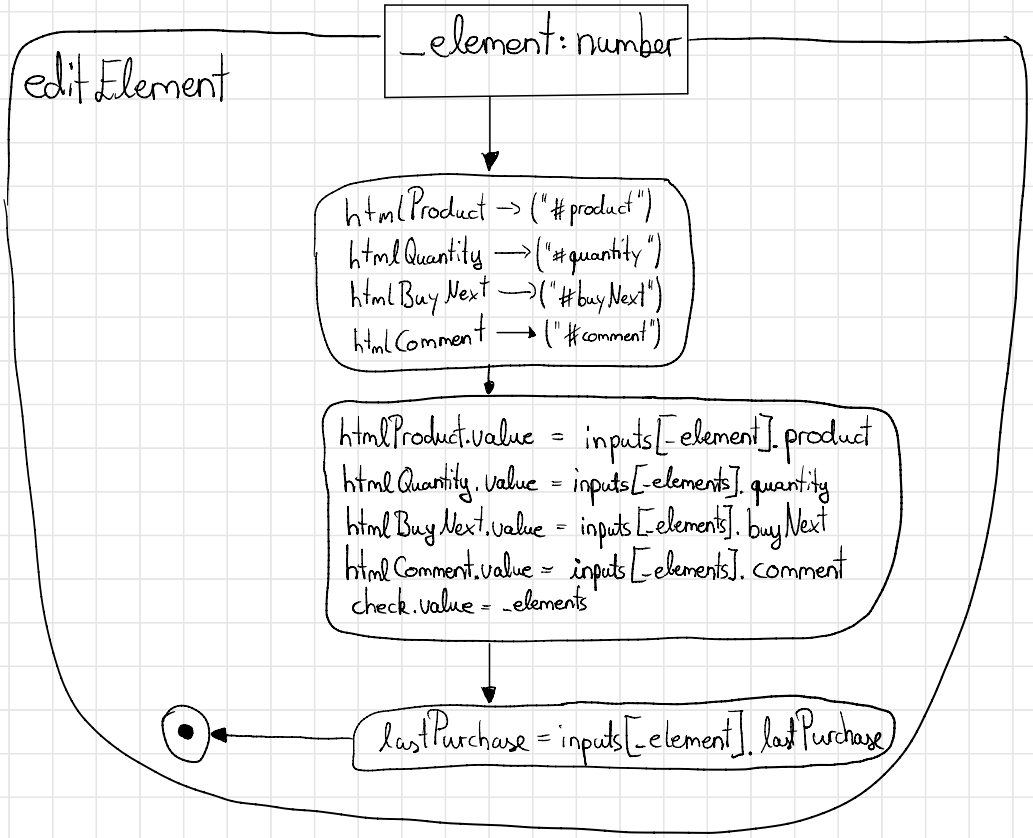

_event: MouseEvent
handleClick
let id: string = (_event.target as Element).id
[id.includes("removeElement")]
let newID: number = cutID(id, 13)
deleteElement(newID)
[id.includes("listElement")]
let newID: number = cutID(id, 11)
clickList(newID)
[id.includes("buyNextElement")]
let newID: number = cutID(id, 14)
buyNextTime(newID)
[id.includes("editElement")]
let newID: number = cutID(id, 11)
editElement(newID)


deleteElement
_element: number
delete inputs[_element]
writeList


buyNexttime
_element: number
inputs[_elements].buyNext=
!inputs[_elements].buyNext
writeList()


handleInputChange
_event: Event
let formData: FormData
= new FormData(document.forms[0]
let buy: string | undefined =
formData.get("buyNext")?.toString();
product =
quantity =
comment =
formData
buyNext=true
[buy == "true"]
[buy != "true"]
buyNext= false


_event: MouseEvent
handleButton
return
[product-"]
let check: HTMLElement=
document.querySelector("#check")
[check.value == defined]
[check.value == undefined]
let element: number = parseInt(check.value)
inputs.push product, quantity, buyNext, isDone, comment, lastPurchase
inputs[element] = product, quantity, buyNext, isDone, comment, lastPurchase
clear html-inputs
clear variables
writeList() ;


editElement
_element: number
htmlProduct -> ("#product")
htmlQuantity -> ("#quantity")
htmlBuyNext -> ("#buyNext")
htmlComment -> ("#comment")
htmlProduct.value = inputs[_element].product
htmlQuantity.value = inputs[_elements].quantity
htmlBuyNext.value = inputs[_elements].buyNext
htmlComment.value = inputs[_elements].comment
check.value = _elements
lastPurchase = inputs[_element].lastPurchase