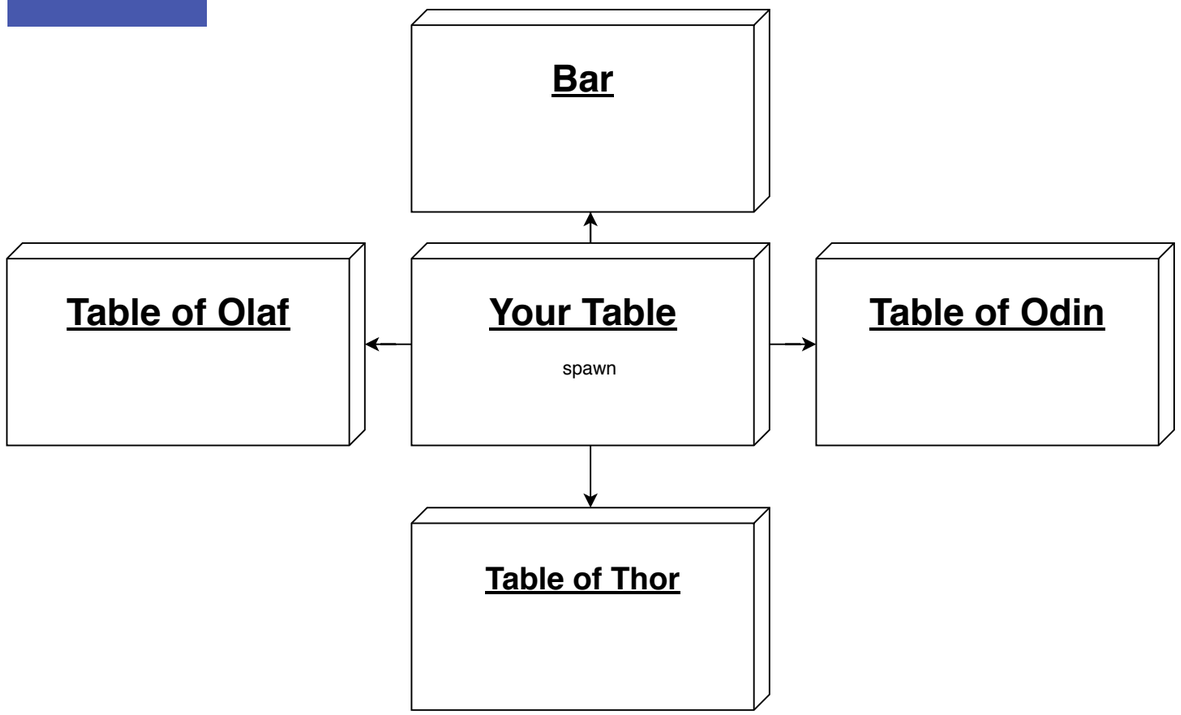
Room Sketch



Bar

Barkeeper: NPC
Id: 1

Table of Olaf

Olaf: NPC
Id: 4

Your Table

Olaf: NPC
Fork: Item
id : 0

Table of Odin

Odin : NPC
Odin's Eye: Item
Id: 2

Table of Thor

Thor : NPC
Hammer : item
Id: 3

Room IDs from 0 -1000
NPC IDs from 1000-2000
Item IDs from 2000 to 3000

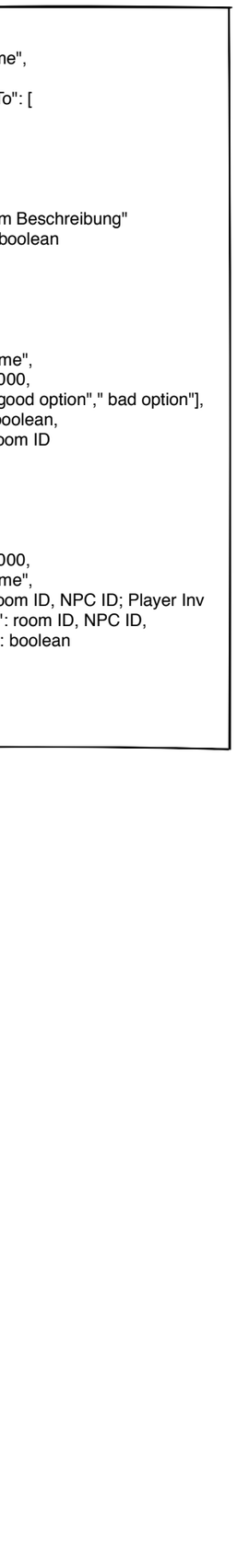
```
{
"room": [{
"name: "nam
"id": 0-1000,
"connectedT
north,
east,
south,
west
],
"desc": "raur
"hasSeen": f
},
{},
...

"npcs":[{
"name": "nar
"id": 1000-20
"dialog:": [ "G
"likesyou": b
"position": ro
},
{},
.....
]
"items": [{
"id": 2000-30
"name": "nar
"position": ro
"destination"
"canPickup"
},
{}
...
```

You are in a tavern, You can see no one, There are no items
walk(w) direction,inventory(i), look(l), take(t) item, drop(p) item,speak(s) person,commands(c), quit(q)

look

/

Class Diagram

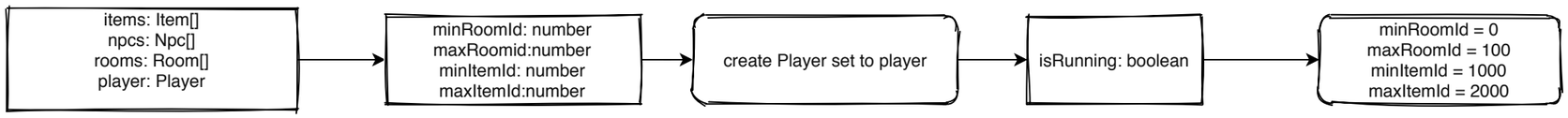

me",
To": [
m Beschreibung"
boolean
me",
000,
good option"," bad option"],
oolean,
oom ID
000,
me",
oom ID, NPC ID; Player Inv
": room ID, NPC ID,
: boolean

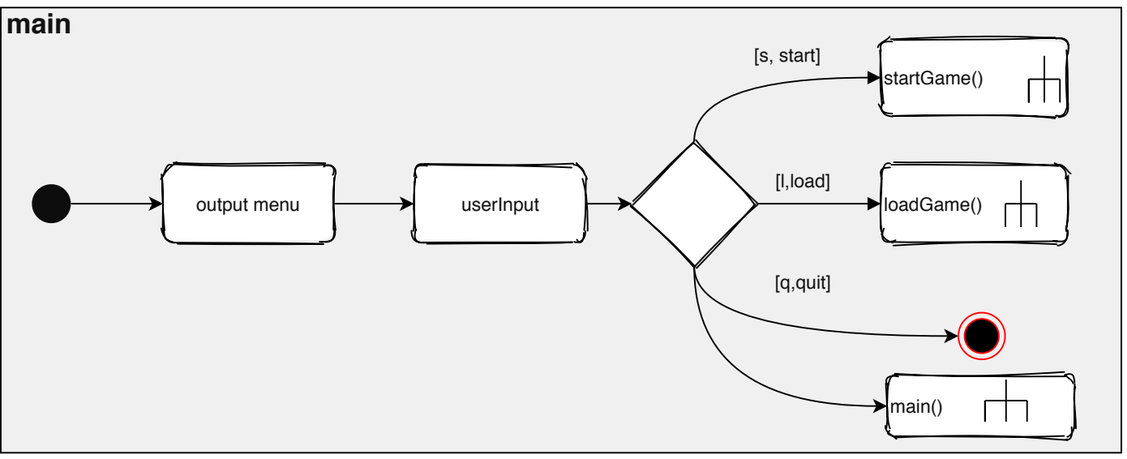
Npc
+dialog:string +id: number +likesyou: boolean
getDialog: string

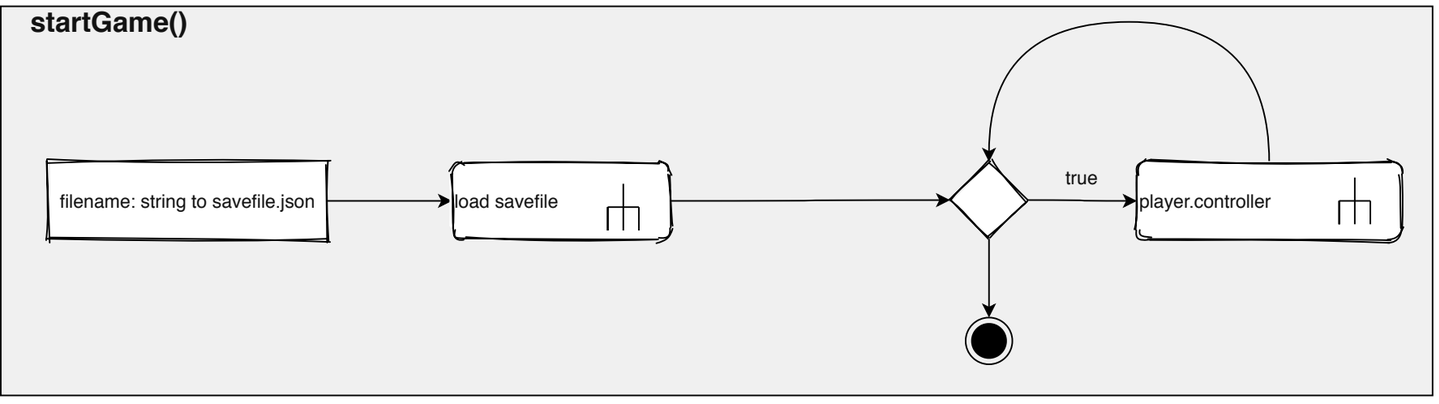
Player
+ canDrop: boolean +canSpeak:boolean +canTake:boolean +canGive:boolean
+commands():string

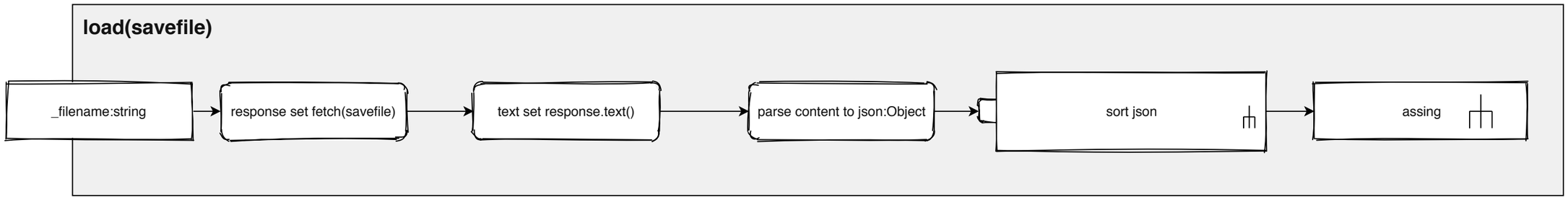
Activity Diagram

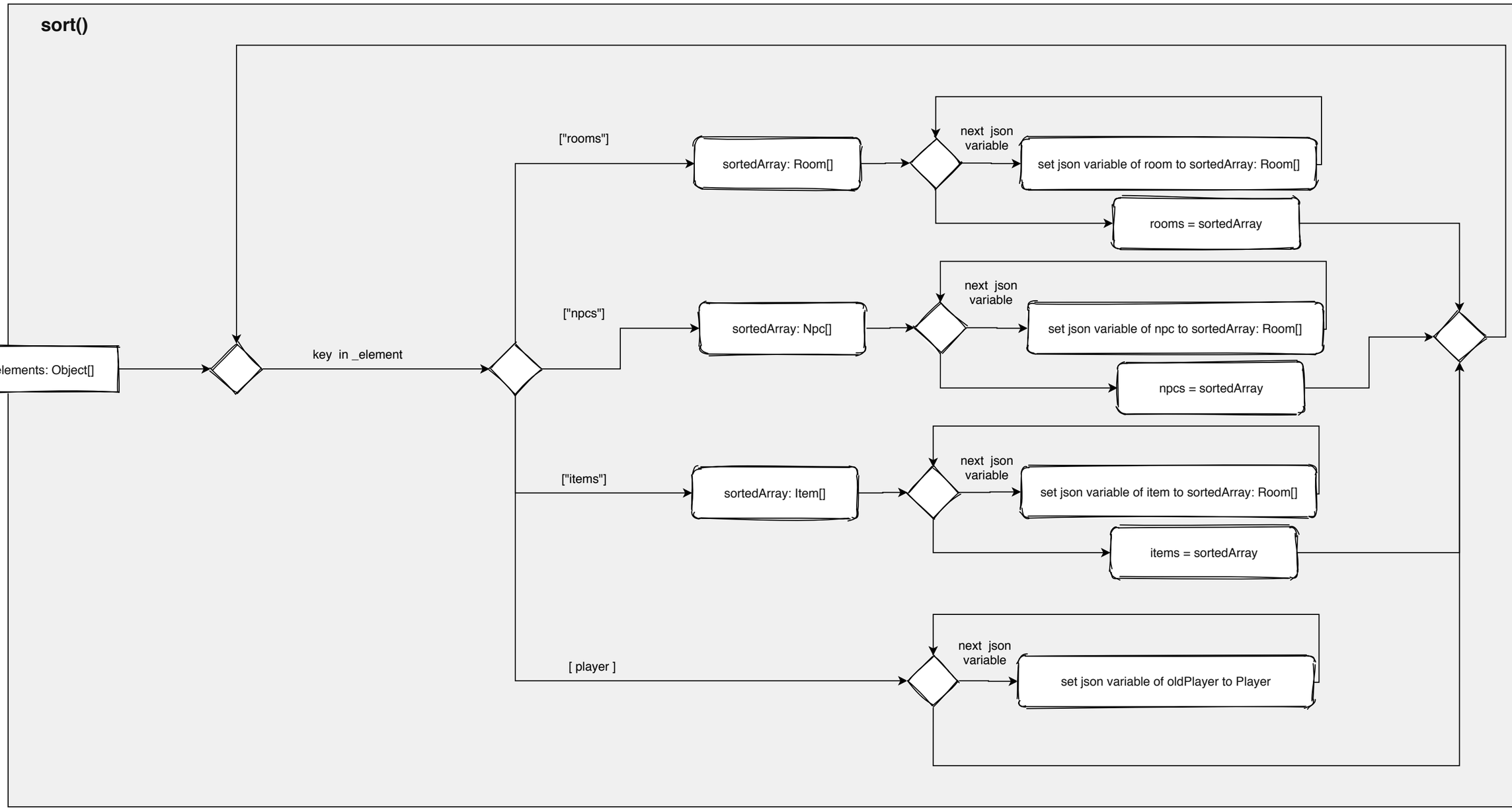
text

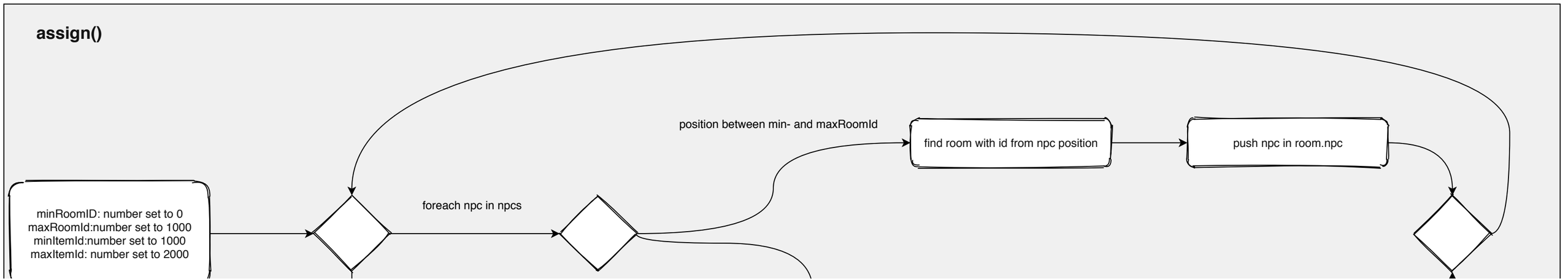









sort()
elements: Object[]
key in _element
["rooms"]
sortedArray: Room[]
next json variable
set json variable of room to sortedArray: Room[]
rooms = sortedArray
["npcs"]
sortedArray: Npc[]
next json variable
set json variable of npc to sortedArray: Room[]
npcs = sortedArray
["items"]
sortedArray: Item[]
next json variable
set json variable of item to sortedArray: Room[]
items = sortedArray
[player]
next json variable
set json variable of oldPlayer to Player


assign()
minRoomID: number set to 0
maxRoomId:number set to 1000
minItemId:number set to 1000
maxItemId: number set to 2000
foreach npc in npcs
position between min- and maxRoomId
find room with id from npc position
push npc in room.npc

position between min- and maxItemId

find room with id from item position

push item in room.item

forEach item in
items

playerController()

currentRoom
hastn seen

output
look()

answerMenu : string[]

currentRoom: Room

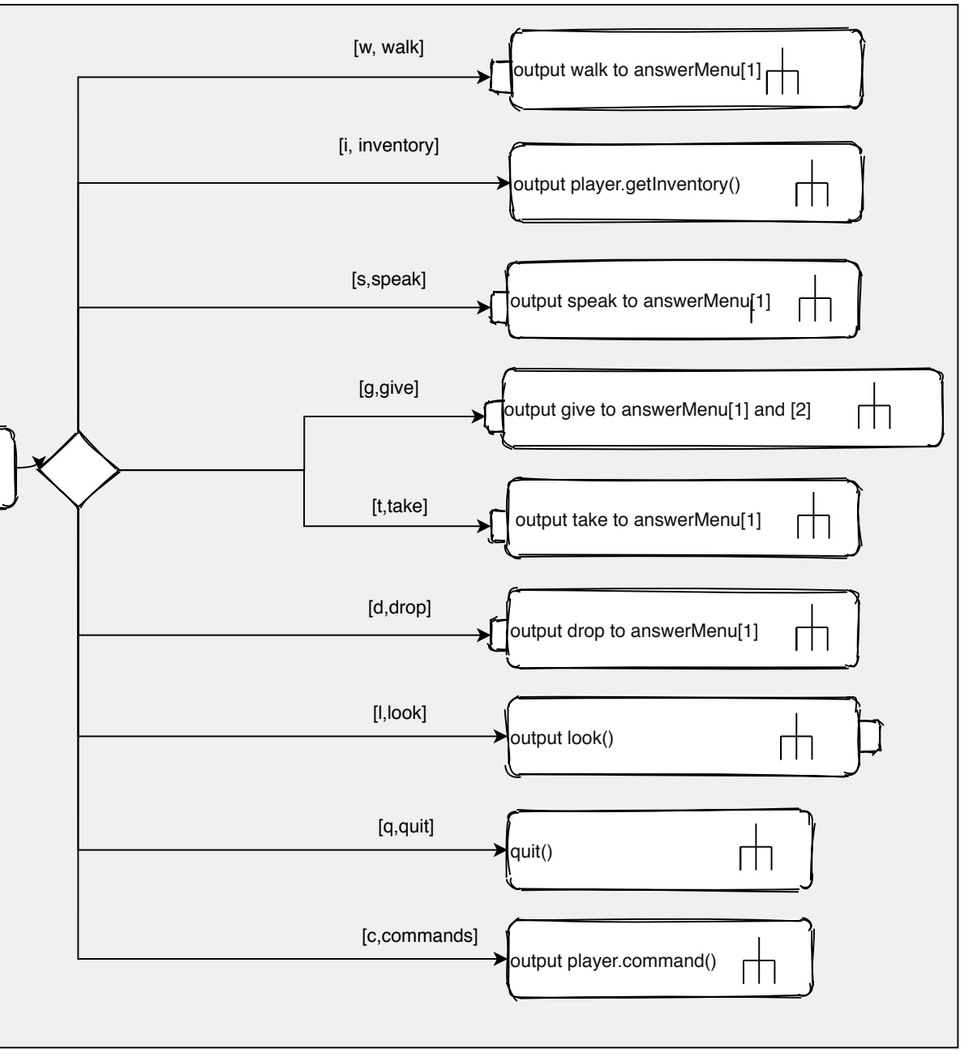
setBool

updateBeerBoard

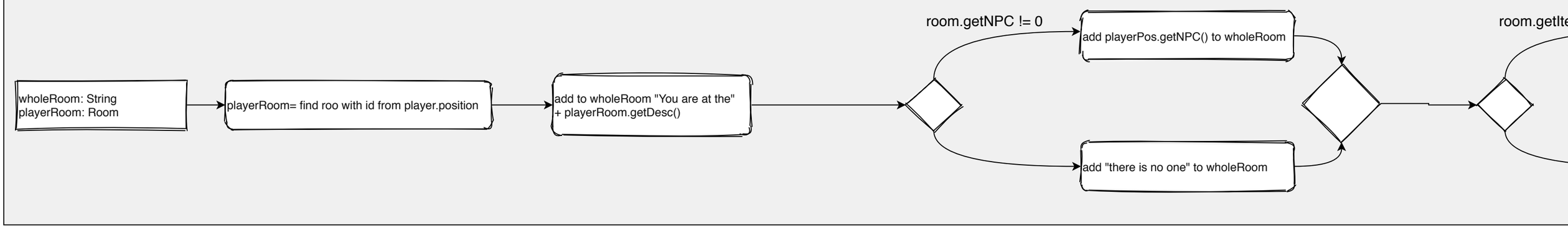
output player.commands()

answerMenu.split(" ") set to prompt

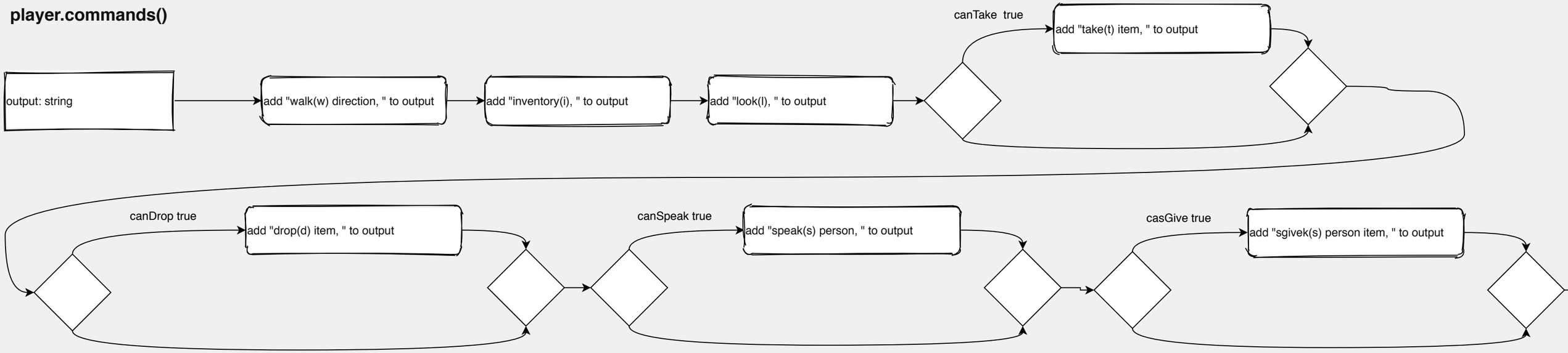
changePicture


[w, walk]
output walk to answerMenu[1]
[i, inventory]
output player.getInventory()
[s,speak]
output speak to answerMenu[1]
[g,give]
output give to answerMenu[1] and [2]
[t,take]
output take to answerMenu[1]
[d,drop]
output drop to answerMenu[1]
[l,look]
output look()
[q,quit]
quit()
[c,commands]
output player.command()

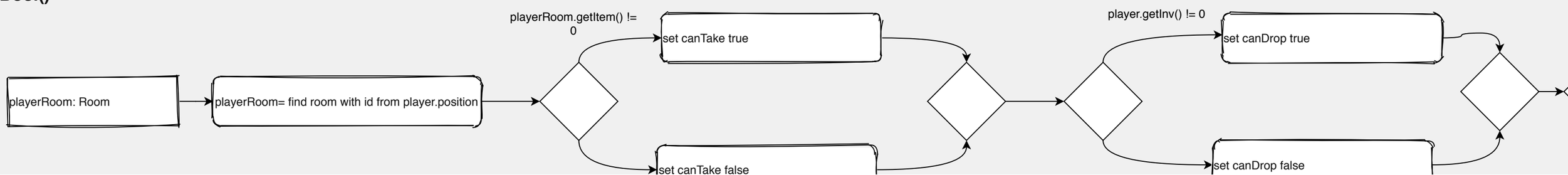
look()

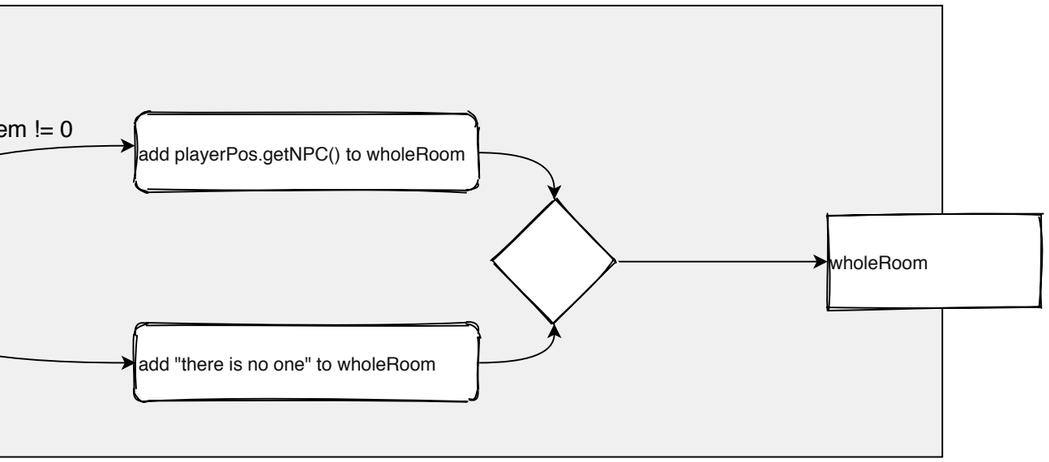


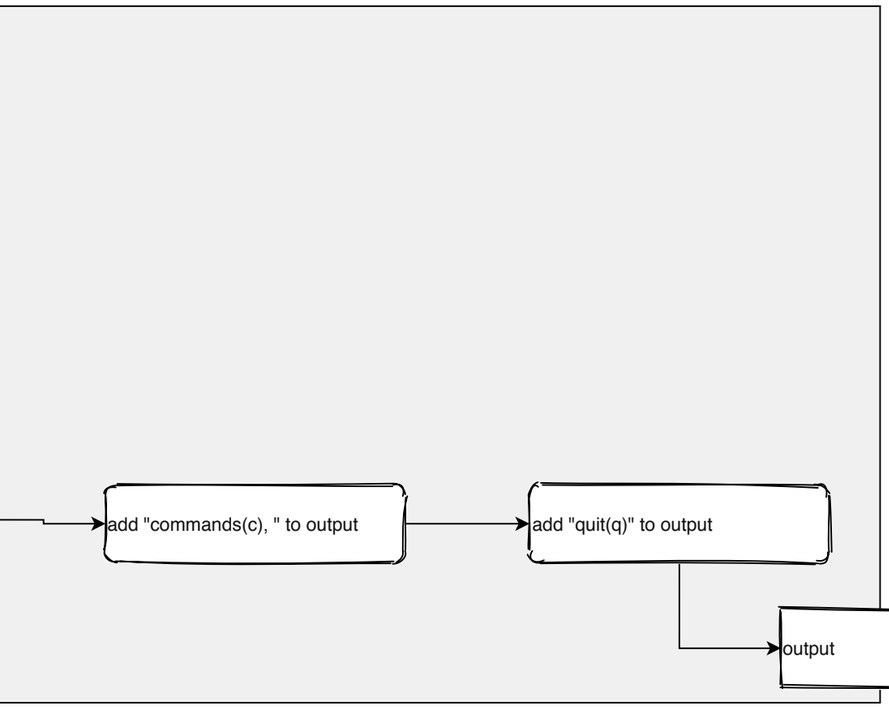
player.commands()



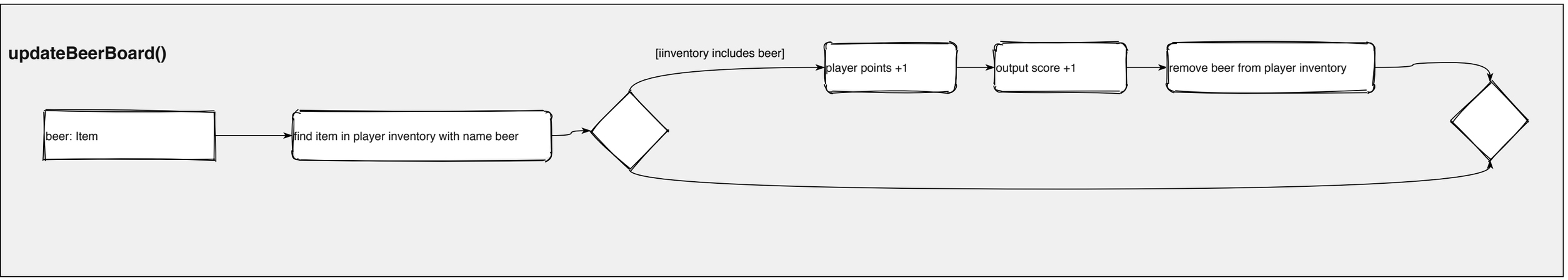
setBool()

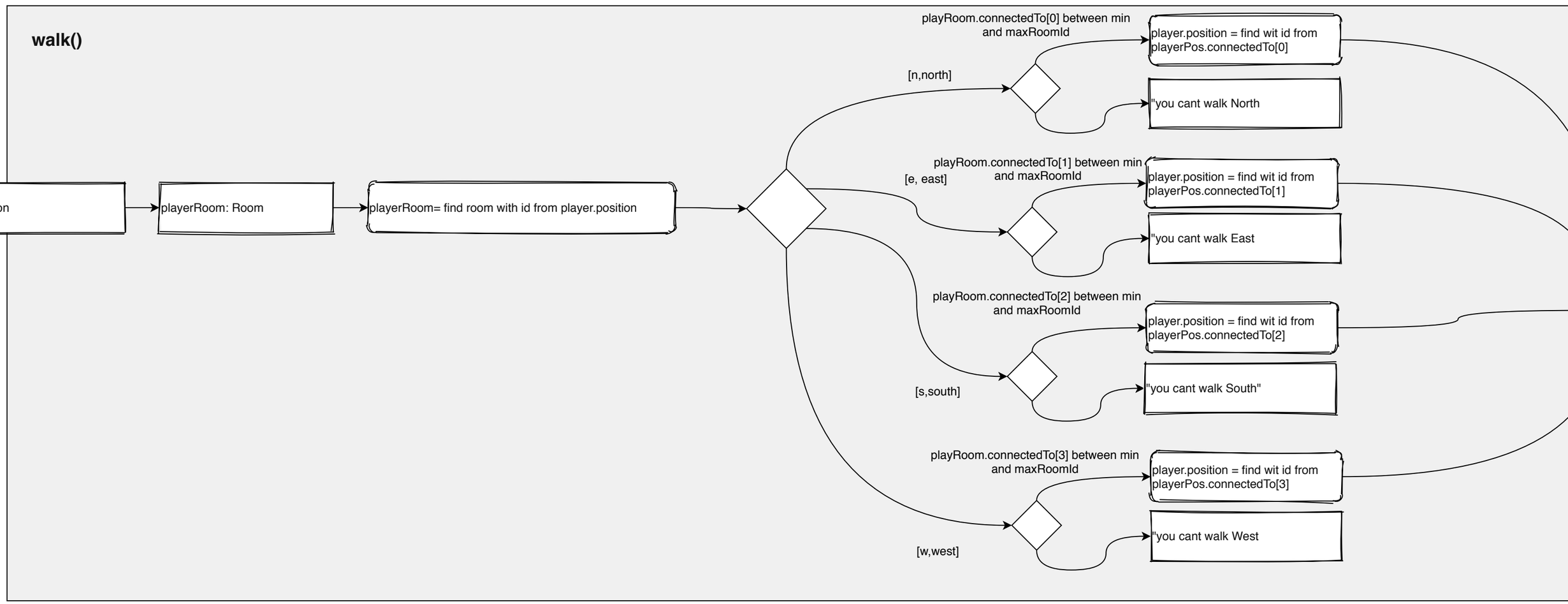



em != 0
add playerPos.getNPC() to wholeRoom
wholeRoom
add "there is no one" to wholeRoom


add "commands(c), " to output
add "quit(q)" to output
output

playerRoom.getNPC != 0
set canSpeak true
set canGuve true
set canSpeak false


updateBeerBoard()
beer: Item
find item in player inventory with name beer
[iinventory includes beer]
player points +1
output score +1
remove beer from player inventory

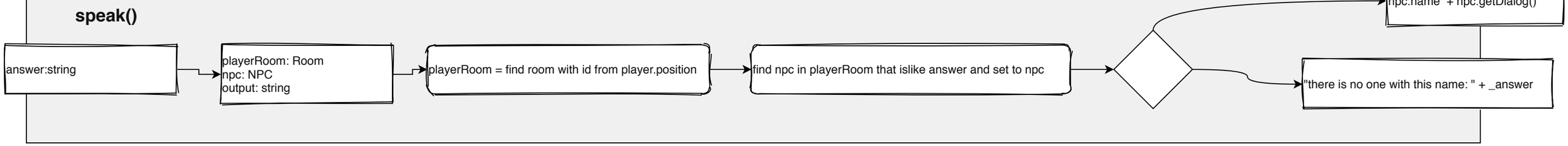

walk()
_direction
playerRoom: Room
playerRoom= find room with id from player.position
playRoom.connectedTo[0] between min and maxRoomId
player.position = find wit id from playerPos.connectedTo[0]
[n,north]
"you cant walk North
playRoom.connectedTo[1] between min and maxRoomId
[e, east]
player.position = find wit id from playerPos.connectedTo[1]
"you cant walk East
playRoom.connectedTo[2] between min and maxRoomId
player.position = find wit id from playerPos.connectedTo[2]
[s,south]
"you cant walk South"
playRoom.connectedTo[3] between min and maxRoomId
player.position = find wit id from playerPos.connectedTo[3]
[w,west]
"you cant walk West

npc is not undefined

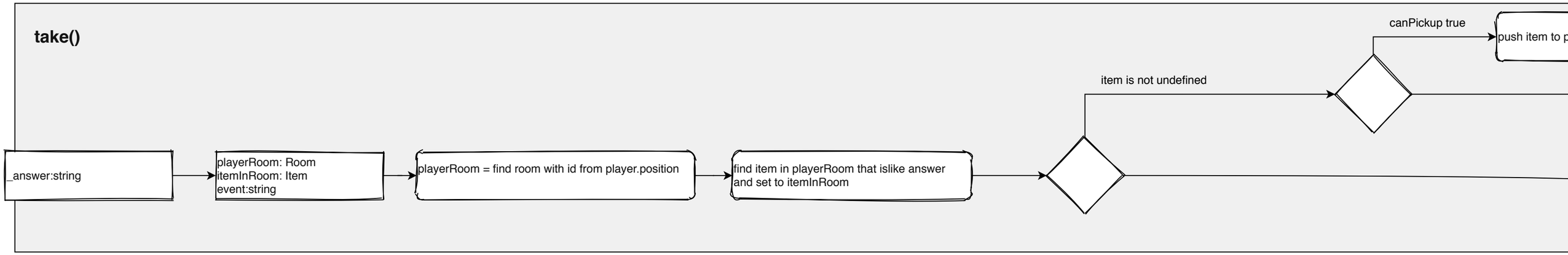
set canGive false

"you walked" + direction

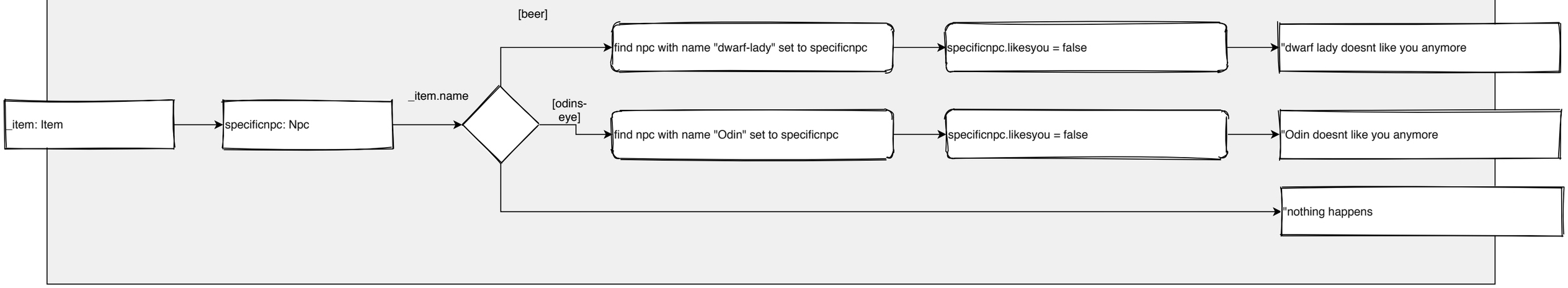
speak()



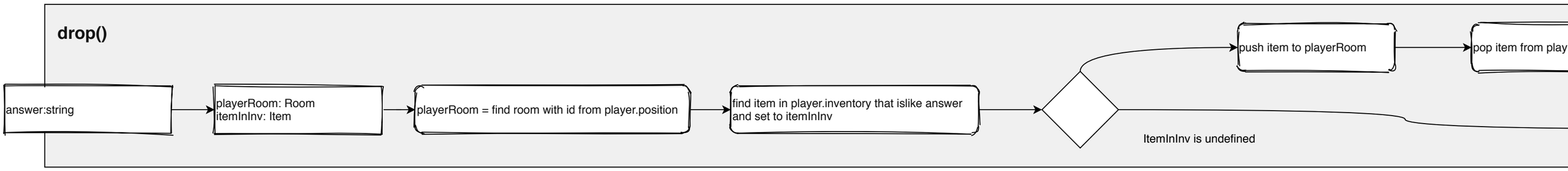
take()

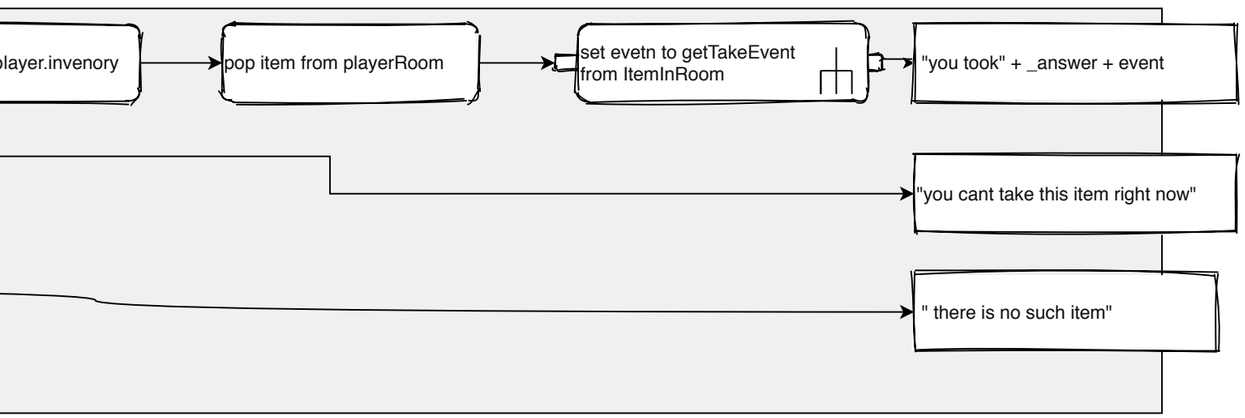


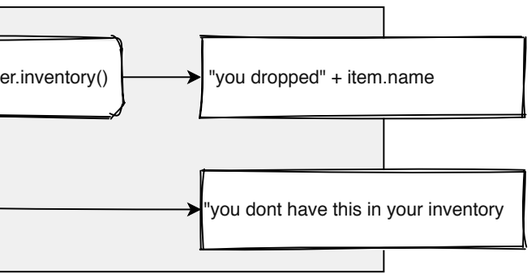
getTakeEvent()

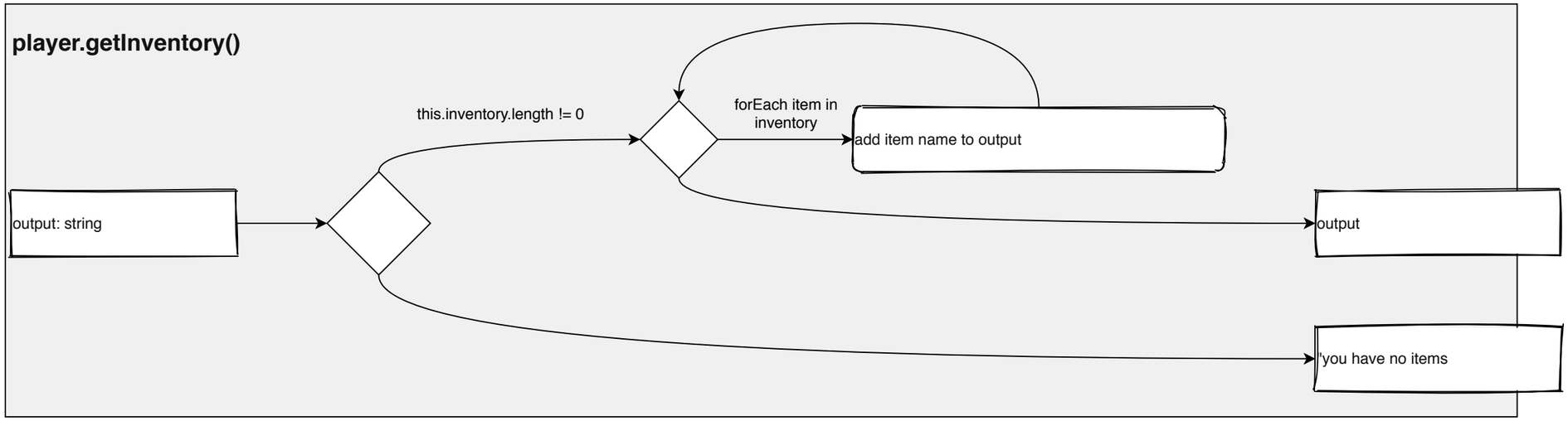


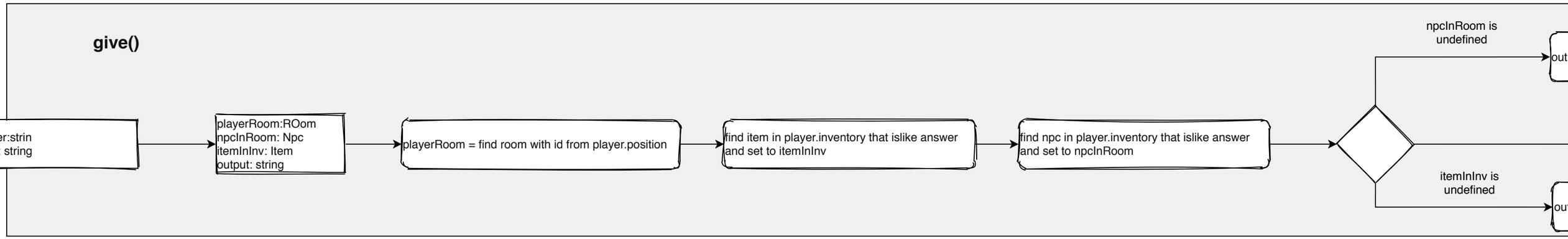
drop()

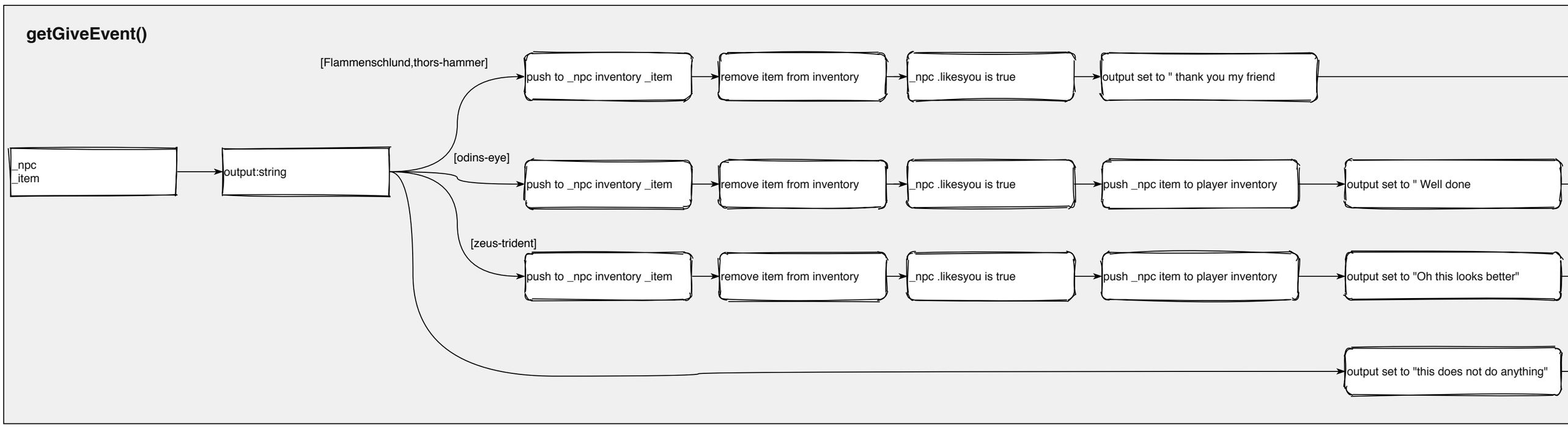


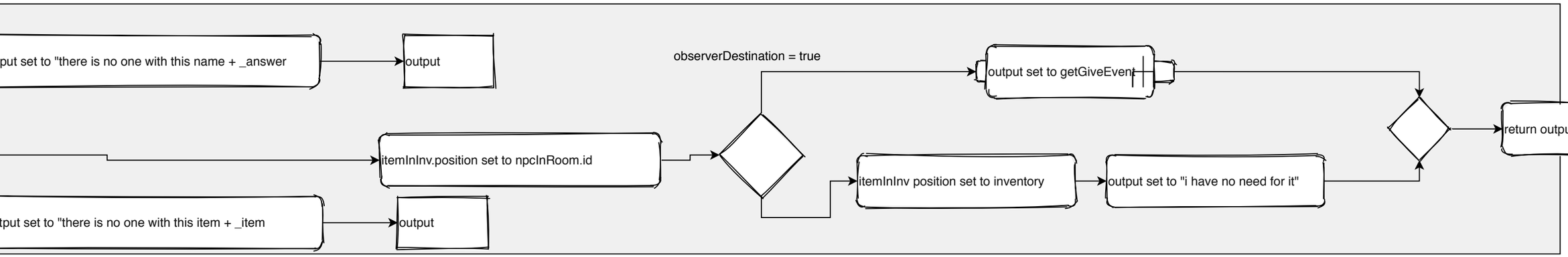

layer.invenory
pop item from playerRoom
set evetn to getTakeEvent
from ItemInRoom
"you took" + _answer + event
"you cant take this item right now"
" there is no such item"

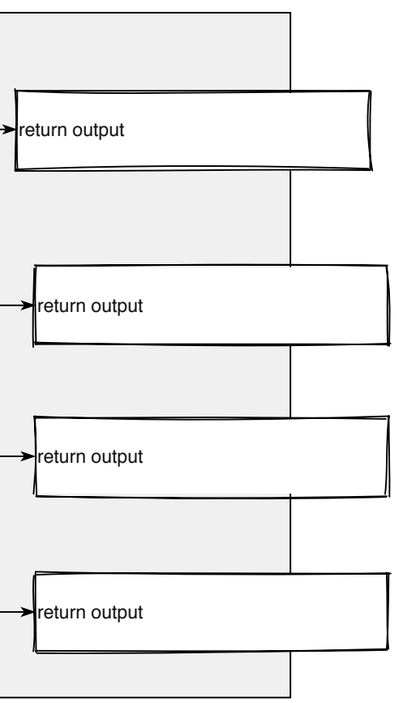

er.inventory()
"you dropped" + item.name
"you dont have this in your inventory

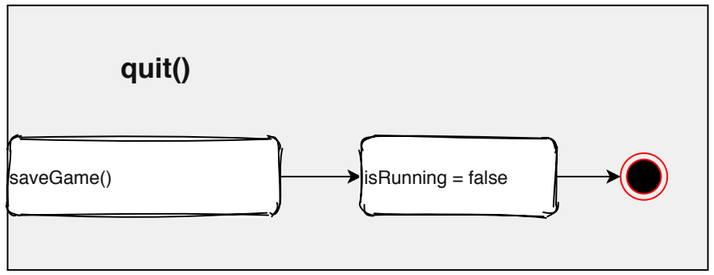

player.getInventory()
output: string
this.inventory.length != 0
forEach item in inventory
add item name to output
output
'you have no items

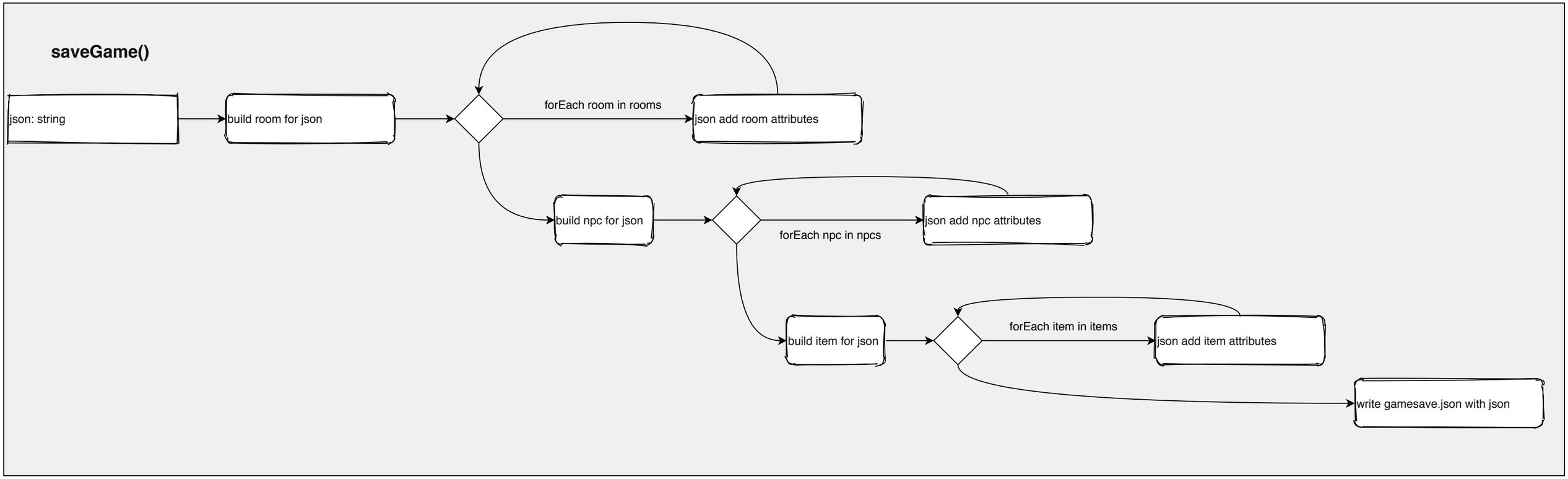

give()
answer:strin
_item: string
playerRoom:ROom
npcInRoom: Npc
itemInInv: Item
output: string
playerRoom = find room with id from player.position
find item in player.inventory that islike answer and set to itemInInv
find npc in player.inventory that islike answer and set to npcInRoom
npcInRoom is undefined
itemInInv is undefined

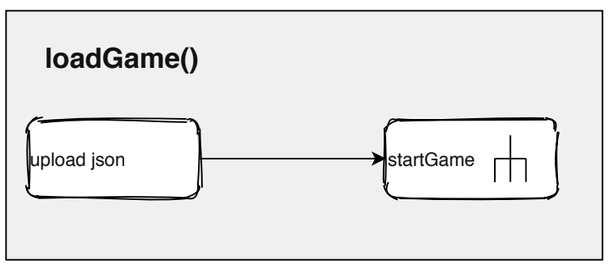

getGiveEvent()
_npc
_item
output:string
[Flammenschlund,thors-hammer]
push to _npc inventory _item
remove item from inventory
_npc .likesyou is true
output set to " thank you my friend
[odins-eye]
push to _npc inventory _item
remove item from inventory
_npc .likesyou is true
push _npc item to player inventory
output set to " Well done
[zeus-trident]
push to _npc inventory _item
remove item from inventory
_npc .likesyou is true
push _npc item to player inventory
output set to "Oh this looks better"
output set to "this does not do anything"


put set to "there is no one with this name + _answer
output
observerDestination = true
output set to getGiveEvent
itemInInv.position set to npcInRoom.id
itemInInv position set to inventory
output set to "i have no need for it"
return output
tput set to "there is no one with this item + _item
output


return output
return output
return output
return output


quit()
saveGame()
isRunning = false


saveGame()
json: string
build room for json
forEach room in rooms
json add room attributes
build npc for json
forEach npc in npcs
json add npc attributes
build item for json
forEach item in items
json add item attributes
write gamesave.json with json


loadGame()
upload json
startGame