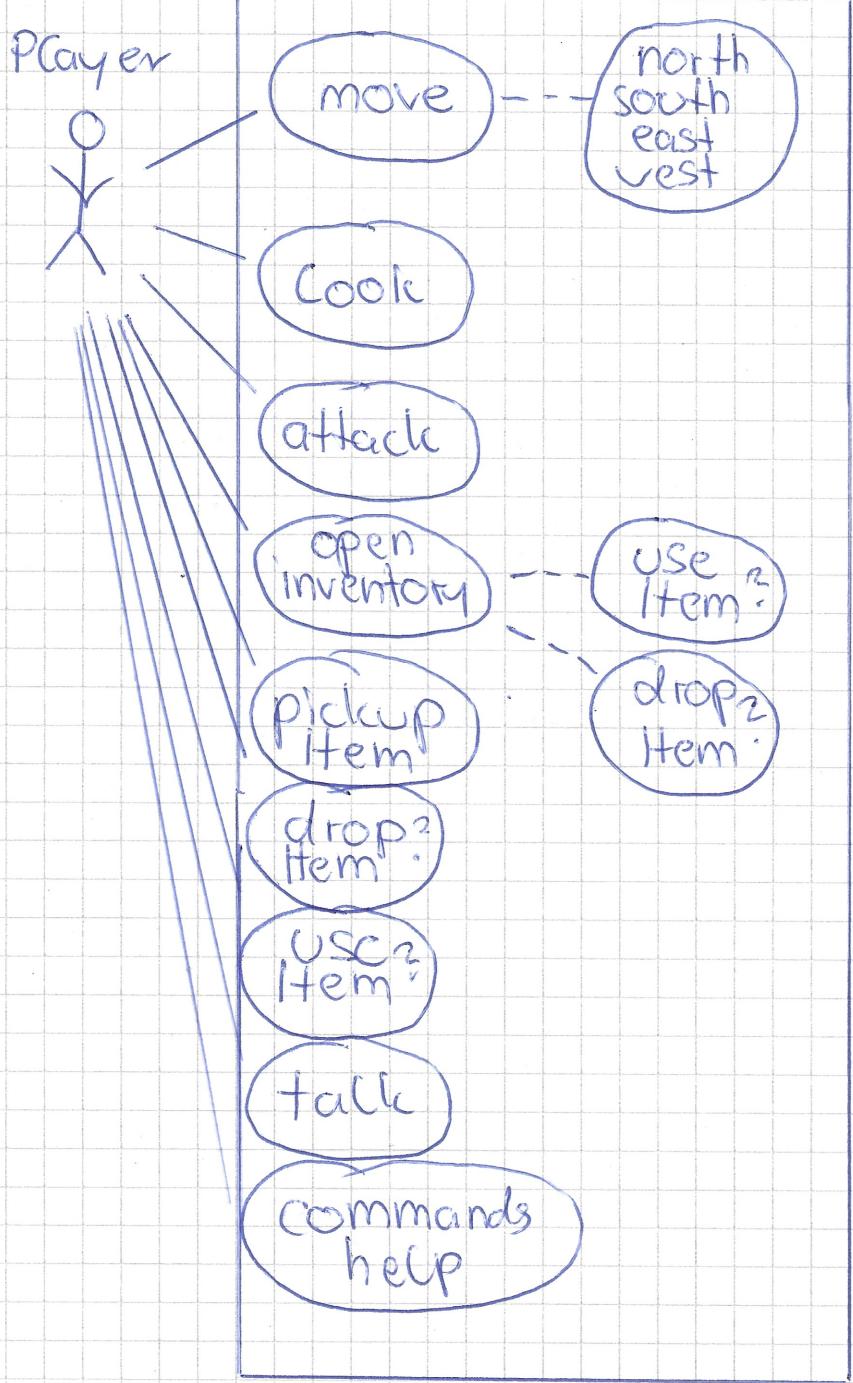
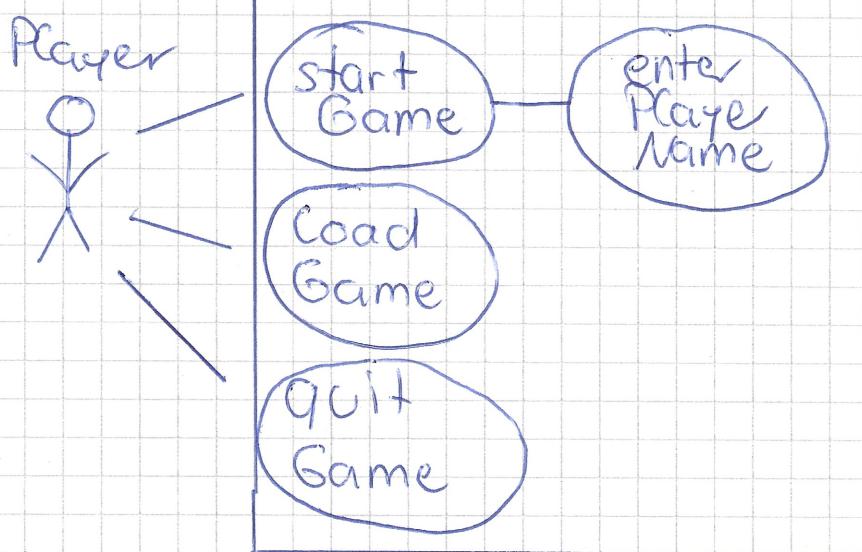


Softwaredesign Text Adventure Game

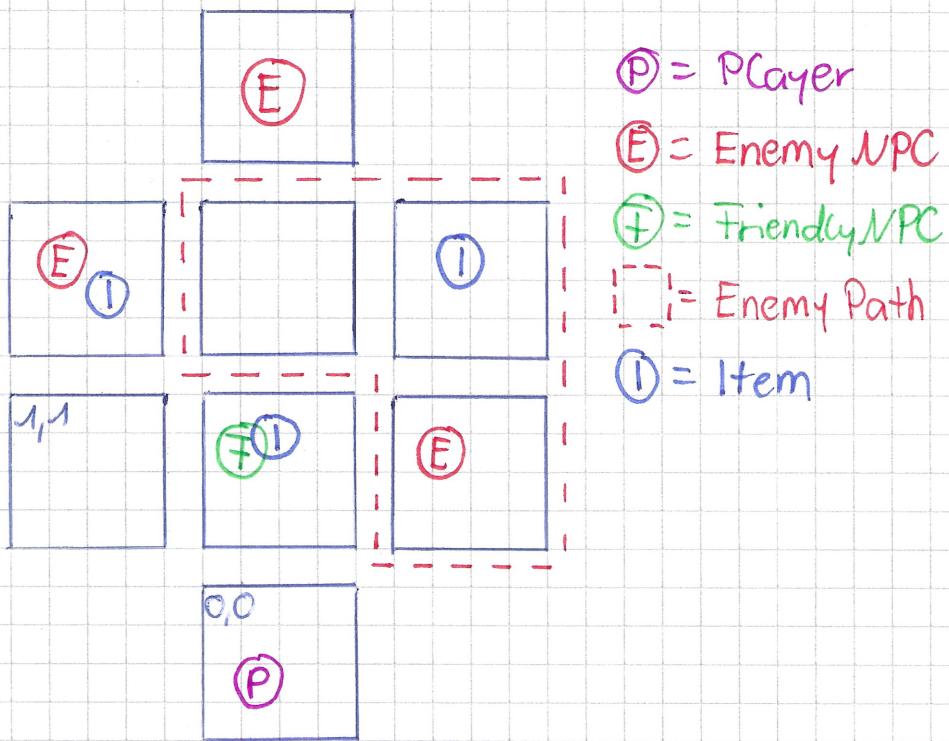


! drop and use item out of inventory or always possible?

Game Menu



Softwaredesign Text Adventure



☐ Enemies should get a Path to limit their Movement if possible.

~~Implementation + Design~~

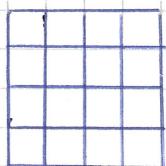
- player needs a possibility to get Items from NPC's \rightarrow maybe through combat?
- player needs to be able to interact with friendly NPC's
- combat system \rightarrow turn-based
 - ☐ player should be able to heal \leftarrow change health based on values.
 - ☐ finish combat / death
- Room information has to be provided
 - current Room
 - NPC's
 - Items
 - Exits? \rightarrow
 - ☐ How does the Navigation work?
 - ☐ Grid?

Software design Text Adventure

Navigation / Map generation

Grid-system

- coordinate based
↳ 2d Array



- + easy to get exits information based on x & y
- Limited creativity
→ Can't make bigger rooms with more than 4 entries

adjacent Rooms

- every Room gets 4 exits variables

North Exit : Room
South Exit : Room
East Exit : Room
West Exit : Room

- + allows more possibilities to design the world
- + more control
- takes up more code space

⚠ Try the adjacent Room system to keep open more possibilities for world creation.
If its too messy go for Grid System. //

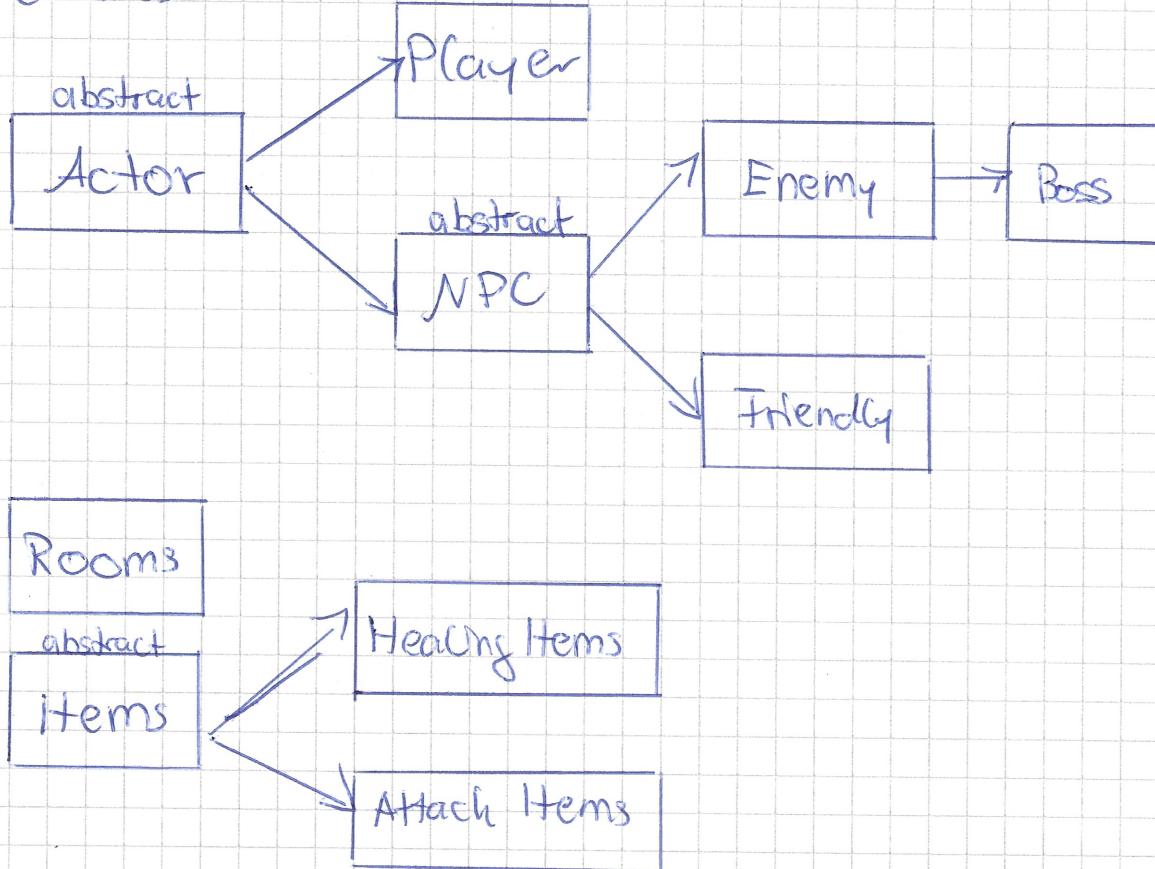
Player Movement

if (player input == "north")

set player.position to player.position, northExit

Software design Text Adventure

Classes



Actor

- name
- position
- inventory
- health
- max. health

Items

- name
- description

Healing

- healValue

Weapons

- attackValue

Rooms

- 4 Exits
- description
- name
- inventory
- characters

Player

- attackValue
- playerName

NPC

- description
- name
- fulfilledValue
- ~~- diet~~
- Movement Path?

Enemy

- attackValue
-

Friendly

- dialogText

Software design Text Adventure

classes

Actor (abstract)

pub. name: string
pub. health: number
pub. maxHealth: number
pub. inventory: Item[]
pub. position: room

takeDmg ()
showInventory ()
pickupItem ()
dropItem ()

Player

pub. attackValue: number
pub. name: string

playMovement ()
lookAt ()
attackTarget ()
talkTo ()
useItem ()
Death ()

NPC (abstract)

pub desc: string
pub name: string
pub fallbackTo: boolean
pub MovementPath: Room[]

npcAI ()
npcMovement
Death ()

Friendly

pub dialog

npcResponse

Enemy

pub attackValue

attackTarget

NPC needs basic AI for making decisions 

Room

pub name: string
pub desc: string
pub Exits: Room[]
pub inventory: Item[]
pub characters: NPC[]

createRoomText ()

Softwaredesign Text Adventure

Items (abstract)

pub name : string
pub desc : string

^{abstract}
use Item()

Healing

pub healValue : number
use Item()

Weapon

pub attackValue : number
use Item()

Software design Text Adventure

Game Menu

print Menu

→ set Player Input

switch

case "start" → set Player Name

→ create Player

case "load" → get save file

→ Play Game

case "quit" → close window

Play Game

print Room Desc

→ set Player Input

Save Input
to action
and target

switch
(action)

case "n" "s" "e" "w" → player. Movement()

case "bold" → player. collect()

case "attack" → player. attack Target()

case "i" → player. show inventory

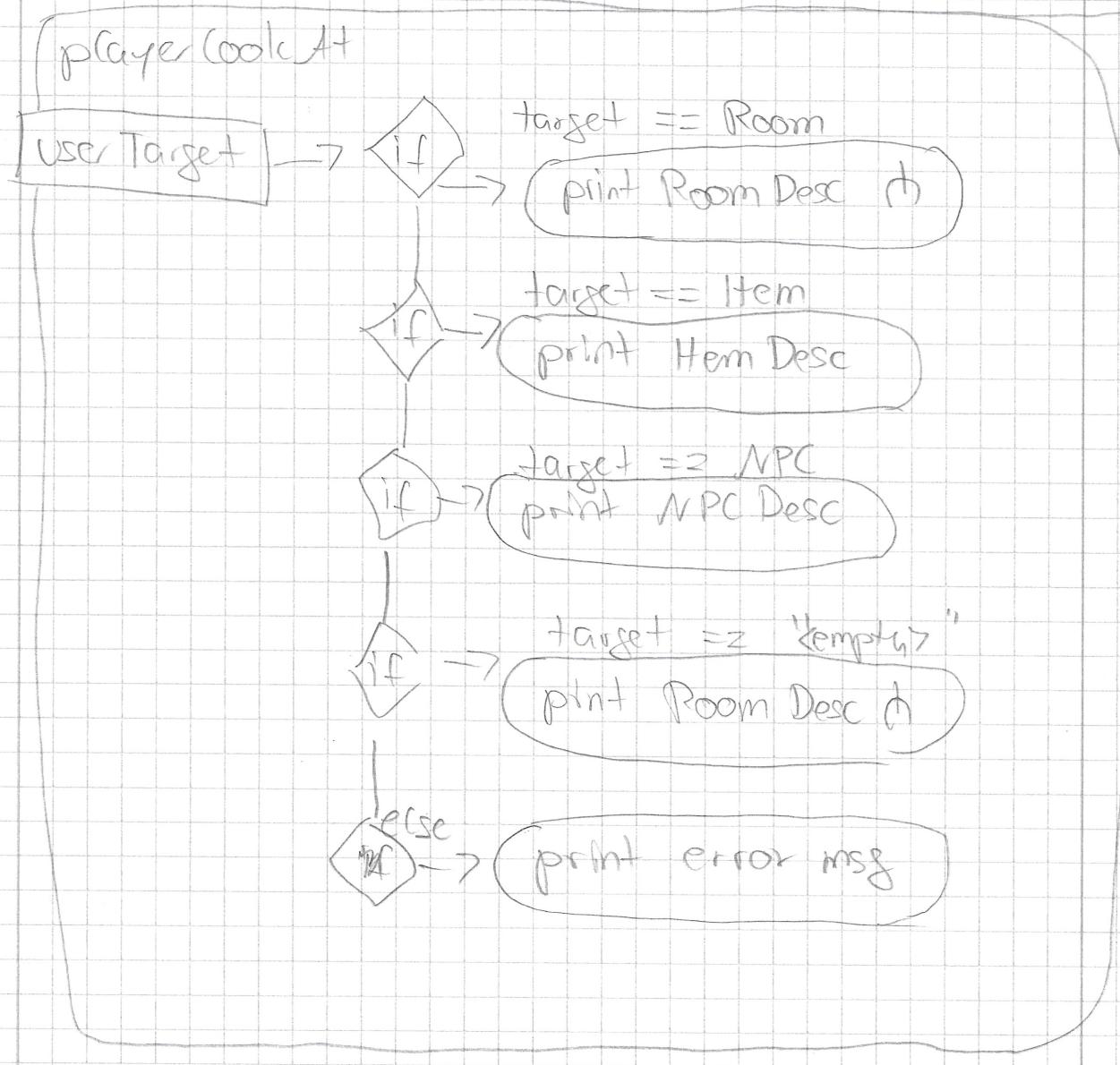
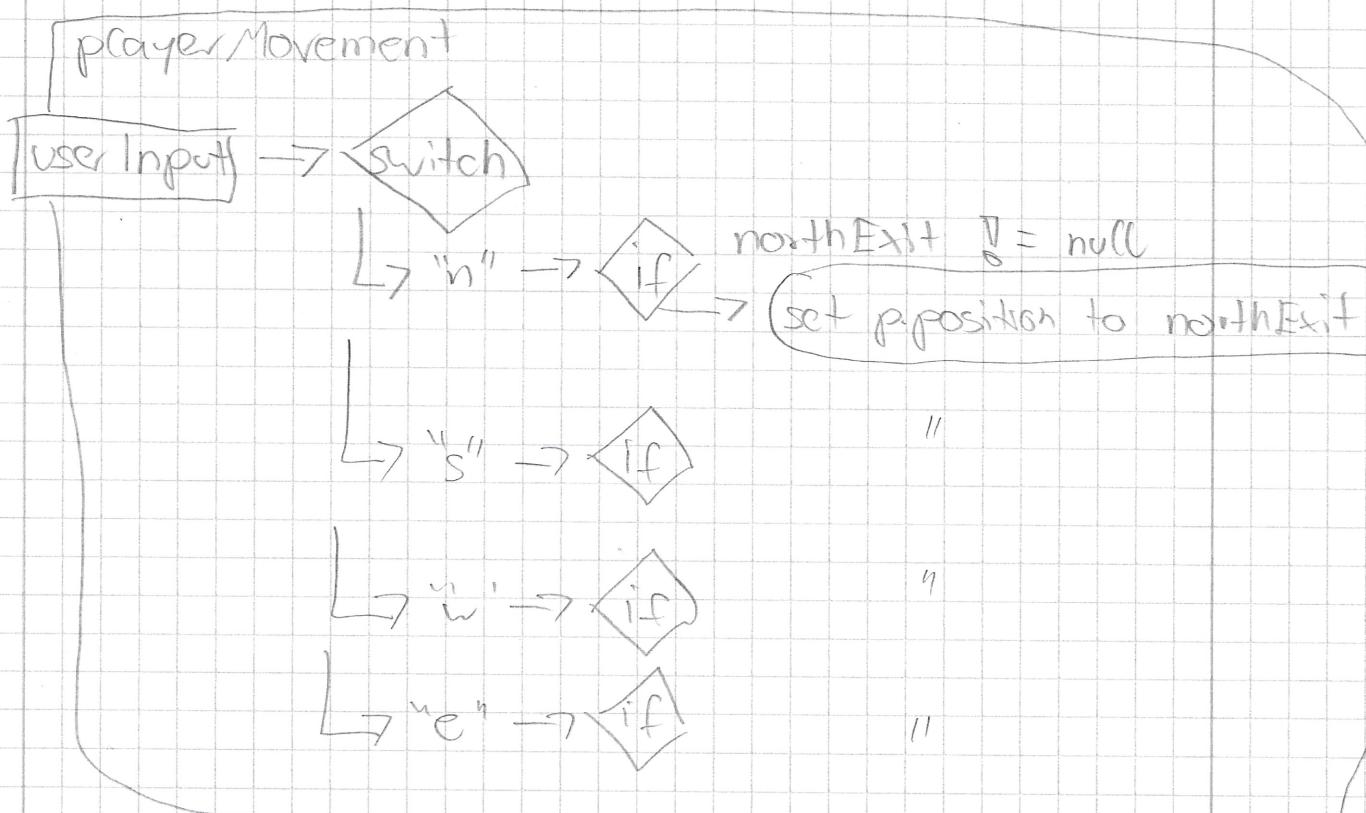
case "p" → player. pickUp Item

case "d" → player. drop Item

case "u" → player. useItem

case "+" → player. talkTo

case "c" → player. print commands



player. attack Target

target

select target
out of Room

player

Attack Val.

print combat
text

target. take DMG

NPC. take DMG

DMG Value

target. health - dmG Value

spice from
Room

drop Inventory

health < 0

if

Show Inventory

i++

inventory

for

inventory[i]. name add to text

Print Inventory
text

Pickup Item

i++

add to
this. inventory

target

for

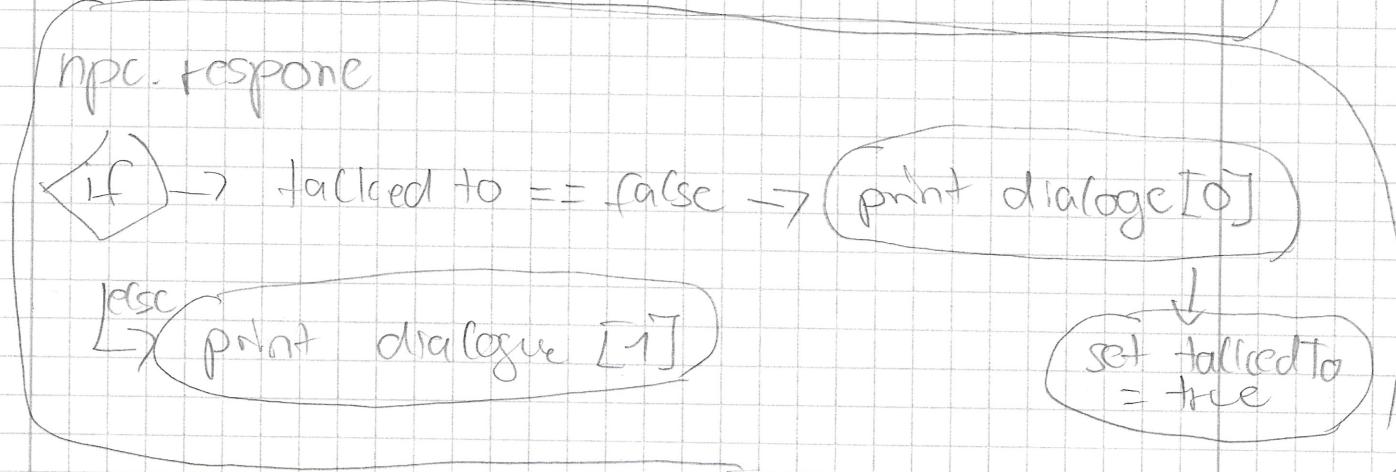
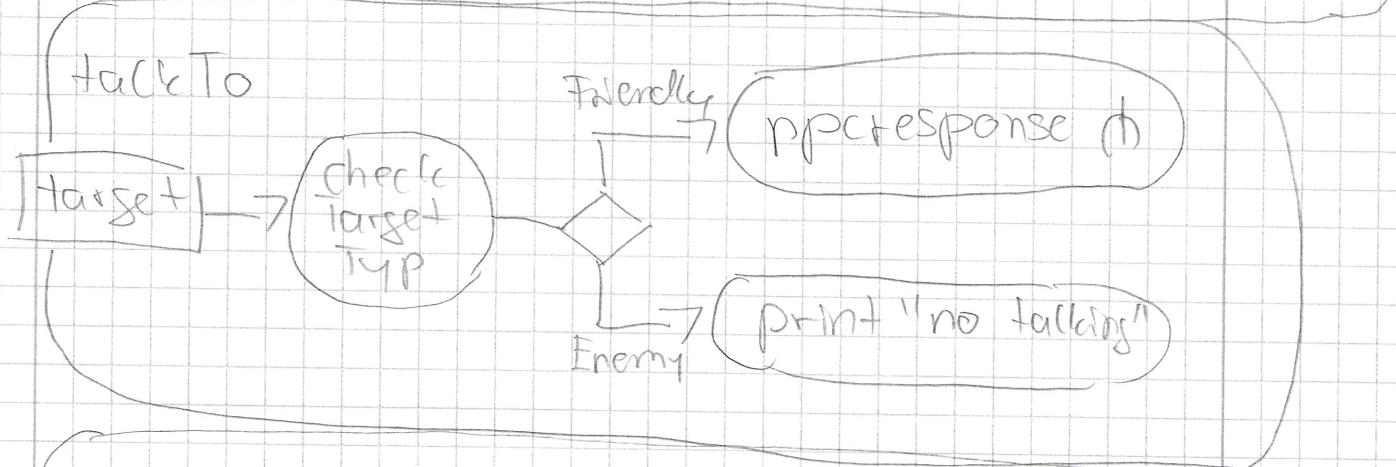
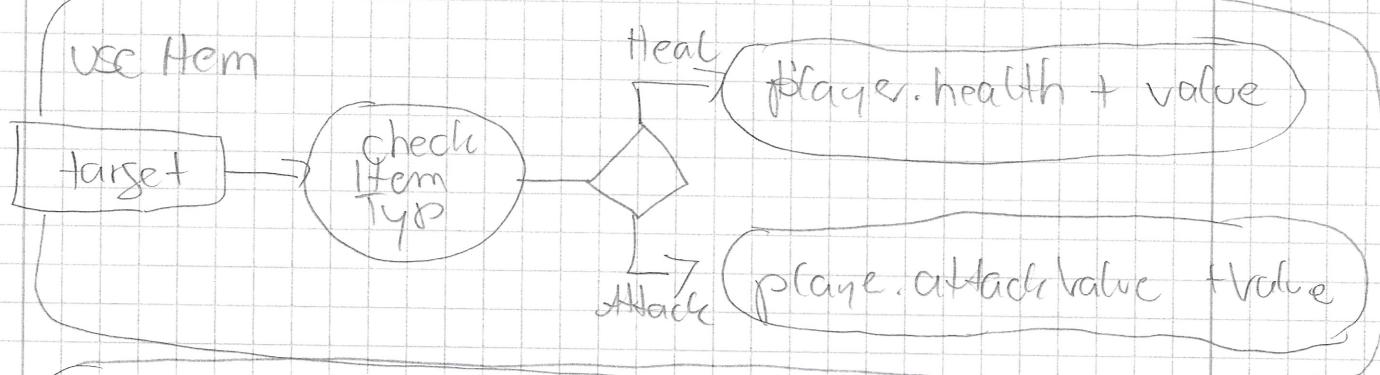
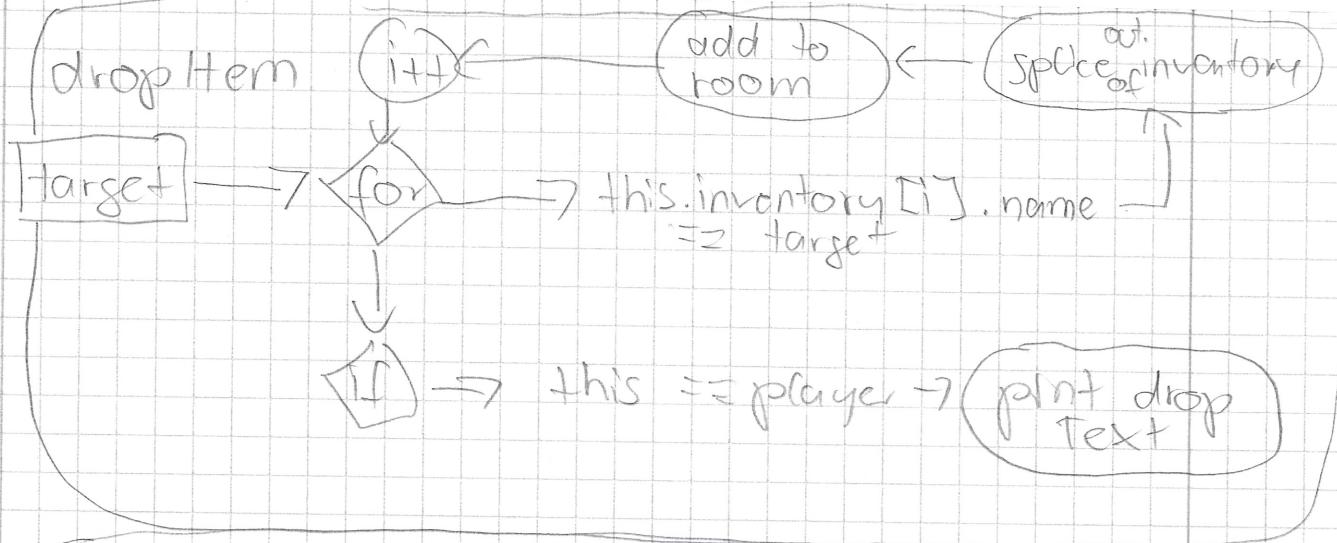
this.room.inventory[i]. name == target

if

if

this == player

print pickup
text



handle NPC AI

! Do after Player move & attack

(go through all
active NPC)

start NPC AI h

Enemy AI

if → this.position == player.position → (attack Player)

↳ random number

switch

case "0" → (this.npcMovement h)

↳ case "1" → this.pickUpItem (first)
(in Room)

NPC Movement

random number

between
0-3

switch

↳ case 0 → change npc position to north
ex: ↑ south

↳ case 1 → "

south

↳ case 2-7 "

east

↳ case 3-7 "

west

Software design Text Adventure

UI Scribble

You are in a dark
Room. With a small
Actor in the centre.

You can see
a Skeleton
a Mummy

There is
a Torch
a Potion

The Skeleton is att-
acking you for 20.
You have 80 health
left.

a Skeleton

} user Input Element

Two HTML Text Elements: all descriptions
are inside the descText as well as
all player action. Gets renewed after
an action.

All combat information is inside combatText
and gets cleared after each turn.

→ Roomdesc stays during combat

Softwaredesign Text Adventure

Saving ?

~~XML~~

LocalStorage ?

↳ LocalStorage

+ Files are saved in Browser

+ easy to access

✓

based on JSON

- requires "key" and desired object to save

→ create Object SaveFile and save
with player.name as key

needs to be converted to a string ↗
with JSON.stringify(obj)

⚠ can't stringify because of object
referencing ! ⚡

possible solution :

save keys for objects and get
back to the desired object with them

⚠⚠⚠ need to be unique ⚡⚠⚠

Things to save :

- player : name / health / position / inventory
- Room with Contents
- NPC , health / position / inventory