Sequenzmemory: Use Case Diagramm

enterTerm --- show Cards

Start

choose Card · - · Flip Card

set Timer

check Card

TimeOut

turn cards back around

End Game

User
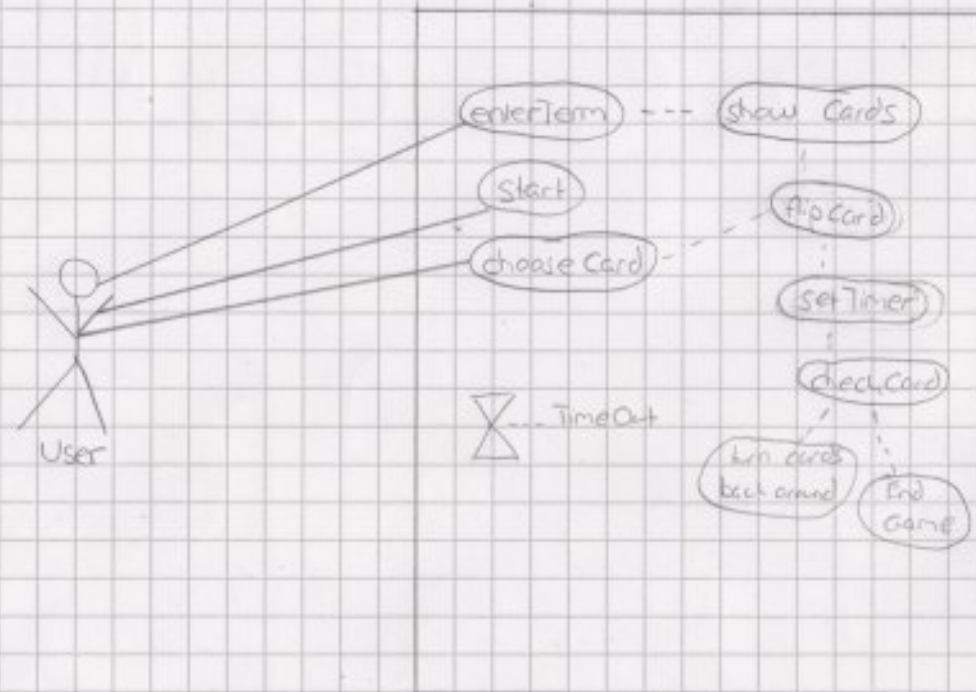
Sequenzmemory: UI Scribble

<h1>

<p>

Sequenzmemory
press start and enter term to play

<span>
id = lime

<div>
id = gamefield

Start

<button>
> click
prompt()

<span>
id = feedback

Sequenzmemory
press start and enter term to play

<span>
id = letter 7
> click

<span>
id = letter 2
> click

<span>
id = letter 3
> click

...
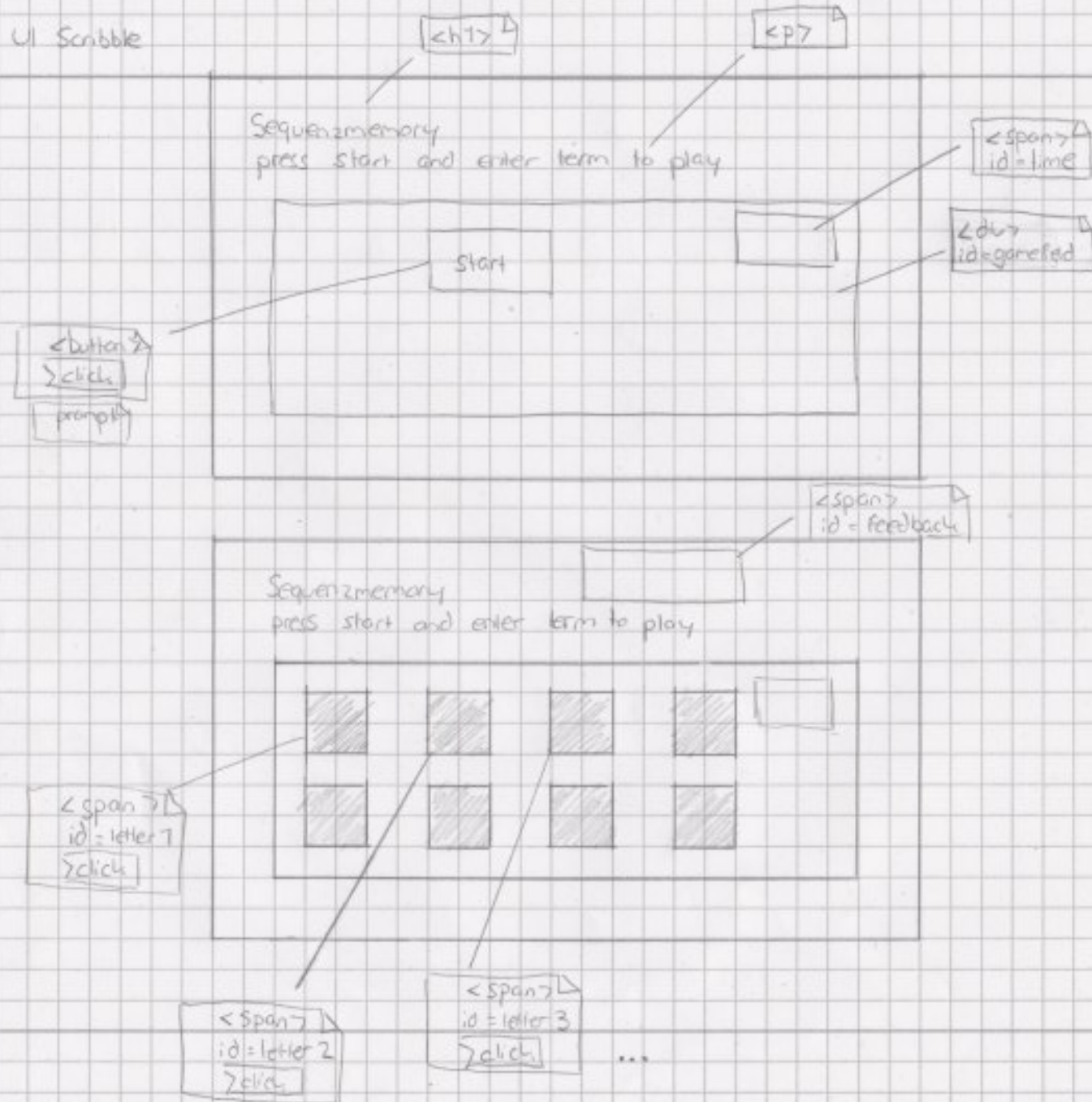
Sequenzmemory: Activity Diagramm
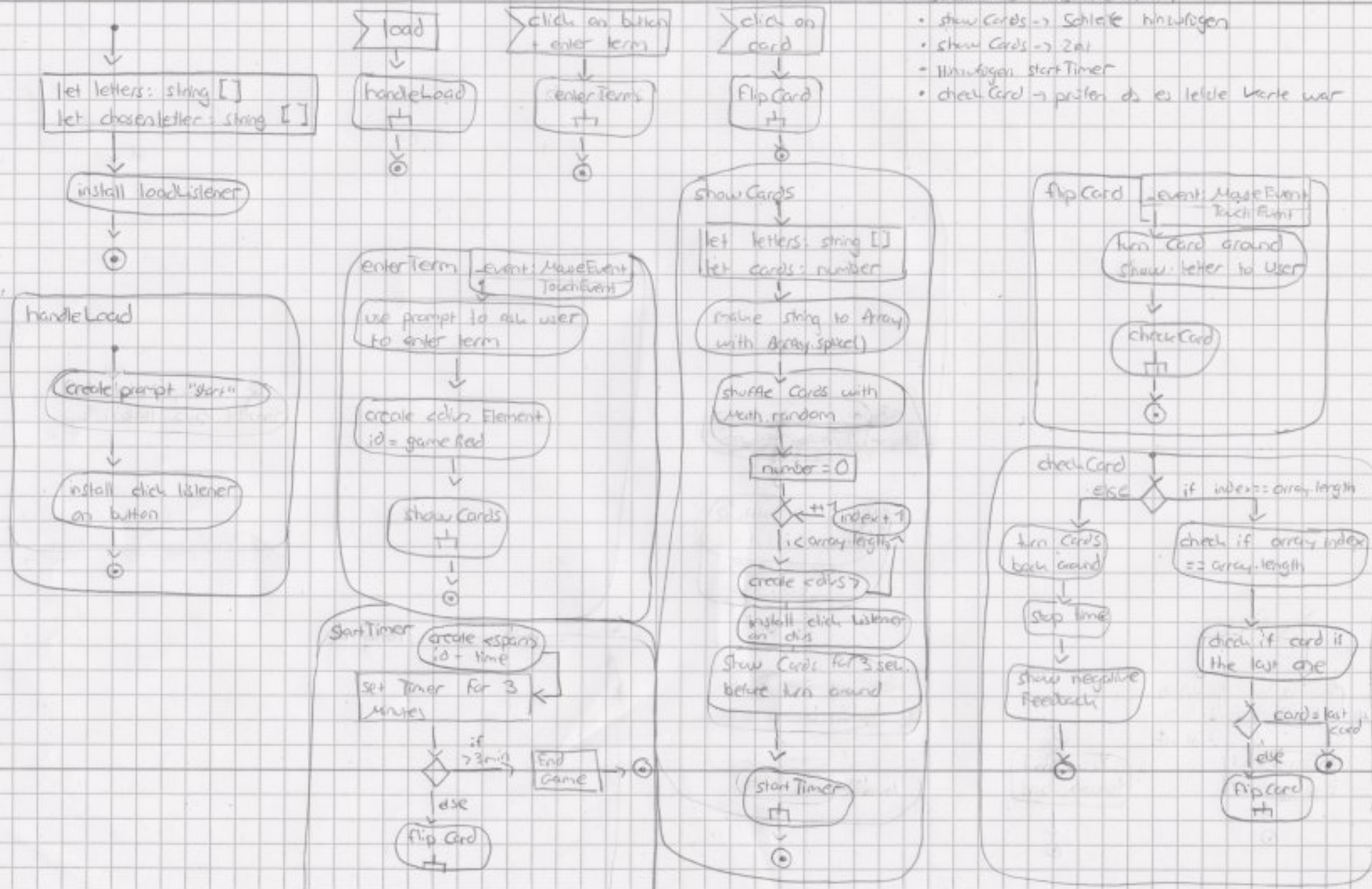
Verbesserungen:
- handleLoad → create prompt
- handle Load → click Listener dus entfernen
- showCards → shuffle Array
- show Cards → Schleife hinzufügen
- show Cards → 2er
- Hinzufügen startTimer
- check Card → prüfen ob es letzte Werte war

Column 1:

let letters: string [ ]
let chosenletter: string [ ]

↓

install loadListener

⊙

handleLoad

create prompt "start"

install click Listener on button

⊙

Column 2:

load → handleLoad ⊙

enterTerm | event: MauseEvent / TouchEvent

use prompt to ask user to enter term

create <div> Element id = game Bad

show Cards ⊙

Start Timer

create <span> id = time

set Timer for 3 minutes

if >3min → End Game → ⊙

else

flip Card

Column 3:

click on button + enter term → enterTerm ⊙

Column 4:

click on card → Flip Card ⊙

show Cards

let letters: string [ ]
let cards: number

make string to Array with Array.splice()

shuffle Cards with Math.random

number = 0

◇ i < array.length → index +1 / ++

create <div>

install click Listener on div

Show Cards for 3 sec. before turn around

start Timer ⊙

Column 5 (flip Card):

flip Card | event: MauseEvent / TouchEvent

turn Card around show letter to user

checkCard ⊙

check Card

◇ else ... if index == array.length

turn Cards back around → stop time → show negative feedback ⊙

check if array index == array.length

check if card if the last one

◇ card = last card / else ⊙

flip Card