Endabgabe - Konzept

Donnerstag, 10. Juni 2021 11:10

zweidimensionales Array mit Objekten

i ↓ / h →	0	1	2
0	0_0	0_1	0_2
1	1_0	1_1	1_2
2	2_0	2_1	2_2

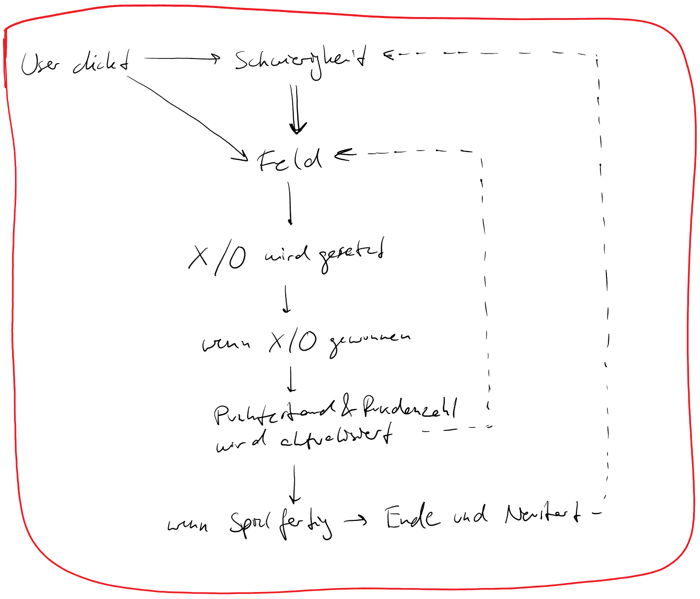
```
Objekt interface: free: boolean
                  X: boolean
                  O: boolean
```

durchgehen des arrays: verschachtelte Schleife

```
for (let i = 0; i < array.length; i++) {
    for (let h = 0; h < array.length; h++) {
        array[i][h]
    }
}
```

HTML:

```
div id: Field
    div id: [i.toString() + h.toString()] /div
    div ...
    div
    div
    ...
/div
```


User clickt
Schwierigkeit
Feld
X / O wird gesetzt
wenn X/O gewonnen
Punktestand & Rundenzahl wird aktualisiert
wenn Spiel fertig → Ende und Neustart

Com turn:

```
→ get random number of id's on field
→ squareClickHandle( ↙ )
```

Variablen deklarieren:

```
Interface Objekt: {}            player1turn = false
ObjektArray[] = [ ]             player1 score = 0
Schwierigkeit[] = [3, 4, 5]     player2 score = 0
                                Runde = 0
```

Seite laden:

```
window.eventListener('load')
  ↳ rufe Funktion drawStartScreen auf
```

drawStartScreen:

```
→ Zeichne Buttons,
→ gib Buttons IDs: 0 = einfach, 1 = mittel, 2 = schwer
→ gib Buttons Eventlisteners (→ DifficultyHandle(ID))
```

difficultyHandle (ID):

```
→ erkennen, welcher Knopf gedrückt wurde (ID)
→ ObjektArray leeren
→ for-Schleife, array füllen mit i/h <= difficulty[ID]

→ drawField
```

drawField:

```
- div leeren
- Schleife (i)/schleife (h)
    → Objekte als HTML Elements in div erstellen
    → ids vergeben (i/h)
    → free, X oder O
    ↳ wenn X oder O, in div <i> mit icon entsprechend
    ↳ EventListener mit (i/h) für free divs → SquareClickHandle (i/h)
```

squareClickHandle (i/h):

```
→ checken welches Objekt geklickt wurde
  ↳ if (player1turn=false) {
        ObjektArray.X = true    ; let Symbol = X
    }
    else {
        ObjektArray.O = true }    let Symbol = O
  ↳ ObjektArray.free = !ObjektArray.free
→ playerturn = ! player1turn
→ drawField
→ let win: boolean = checkRoundEnd()
→ if (win == true)
    ↳ endRestartRound()
```

checkRoundEnd (Symbol): boolean

```
→ checken ob Runde gewonnen:

       { Schleife i < ObjektArray.length                        { Schleife
       |  let win = true                                        |
       |  Schleife h < ObjektArray.length                       |
rows   {    if ObjektArray[i][h].Symbol = false      columns    {
       |      ↳ win = false                                     |
       |  ↳ if (win = true)                                     |
       {      ↳ return                                          {
```

endRestartRound:

```
→ wenn player1turn = true
      ↳ player2score += 1
↳ else
    ↳ player1score += 1
→ Runde += 1
→ let difficultyID = 0
→ wenn 4 Arrays
    ↳ difficultyID = 1
  else wenn 5 Arrays
    ↳ difficultyID = 2
→ wenn Runde < ObjektArray.length
  ↳ difficultyHandle(difficultyID)
else
```

↳ gameOver(difficulty ID)

gameOver(ID)
→ div leeren
→ div füllen mit
↳ RestartButton + EventListener → difficultyHandle (ID)
↳ choosedifficulty + EventListener → drawStartScreen