

## **Entwicklung Interaktiver Anwendungen - Julian Schüler - Abschlusssaufgabe**

### **Funktionale Analyse**

#### **Aufbau**

Aufgebaut ist die Seite wie folgt. Links befindet sich das Spielfeld in dem alles stattfindet. Rechts neben dem Spielfeld befinden sich zwei Felder. Das Obere zeigt die gesammelten Punkte und den zugehörigen Status an. Das Untere zeigt die Anleitung an in der die Steuerung und das Ziel definiert sind. Bei verkleinern des Bildschirms z.B. Tablet wechselt die Anleitung in ein DropDown Menu, erkennbar durch einen schwarzen Punkt, der sich oben links positioniert. Das Scorefeld schiebt sich einfach nur unter das Spielfeld.

#### **Ablauf**

Im Spielfeld das aus schwarzen Kästchen erstellt wird gibt es ein bewegliches Objekt und einen Punkt. Das Ziel wie es auch im Spiel definiert wird ist zu versuchen so viele Punkte wie möglich zu sammeln. Das wird mit Hilfe des beweglichen Objekts (der Schlange) umgesetzt, indem man diese über die angegebenen Tasten auf diesen Punkt steuert. Ist der Punkt erreicht verschwindet er und es entsteht ein neues Element an der Schlange, die zu Beginn aus drei Elementen besteht. Gleichzeitig wird an einer zufälligen anderen Stelle des Feldes ein neuer Punkt generiert. Wichtig dabei zu beachten ist, dass man nicht mit sich selbst kollidiert, weil ansonsten das Spiel abgebrochen wird. Die Enden des Feldes kann man durchlaufen, da man dann auf der anderen Seite wieder herauskommt. Der entscheidende Faktor hierbei ist die Schnelligkeit der Animation. Bei jedem Punkt der von der Schlange eingesammelt wird, wird die Schlange schneller. Dadurch wird das Spiel nicht langweilig und man mehr gefordert.

#### **Liste der Nutzerinteraktionen**

Steuerung über die Tasten W, A, S, D

- W → Nach oben
- A → Nach links
- S → Nach unten
- D → Nach rechts

Bei Abbruch des Spiels erscheint eine Alert Box die „Game Over“ anzeigt

Bei Klick des „ok“ Buttons wird die Seite neu geladen und das Spiel beginnt von vorne.

## **Technische Analyse**

Das Spiel wird mit Hilfe von HTML, CSS und Typescript umgesetzt.

Im HTML werden nur die Rahmenbedingungen für den inhaltlichen Aufbau gesetzt.

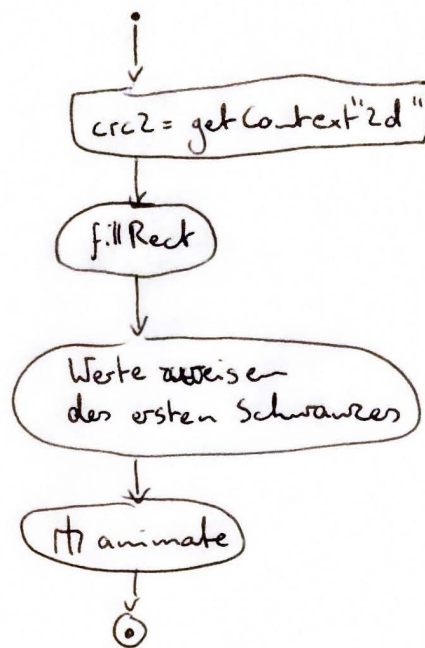
Im CSS wird die Seite gestaltet und auch für das Responsive Design verfügbar gemacht.

Im Typescript findet alles Funktionale statt. Hier werden alle Informationen und Bedingungen festgelegt.

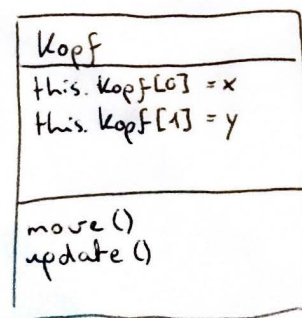
Eigentlich war der Aufbau des Spiels auch ohne Klassen möglich. Da aber Klassen verlangt waren teilte ich den Kopf in eine Klasse auf.

## Aktivitätsdiagramme

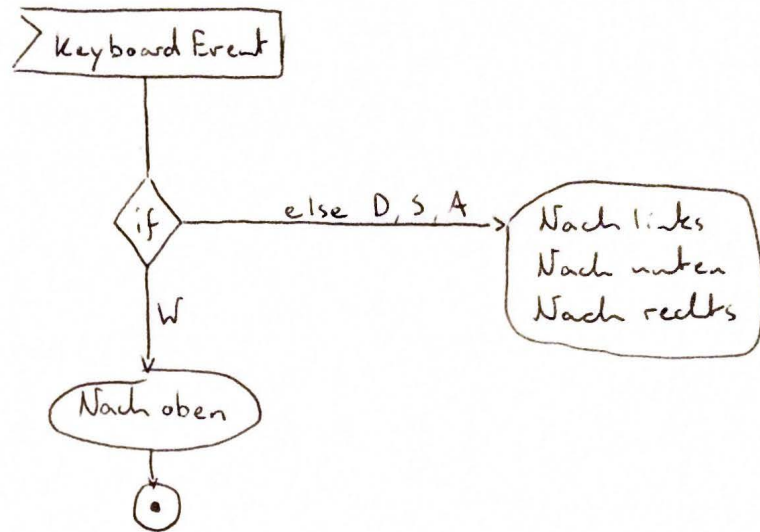
function Spiel



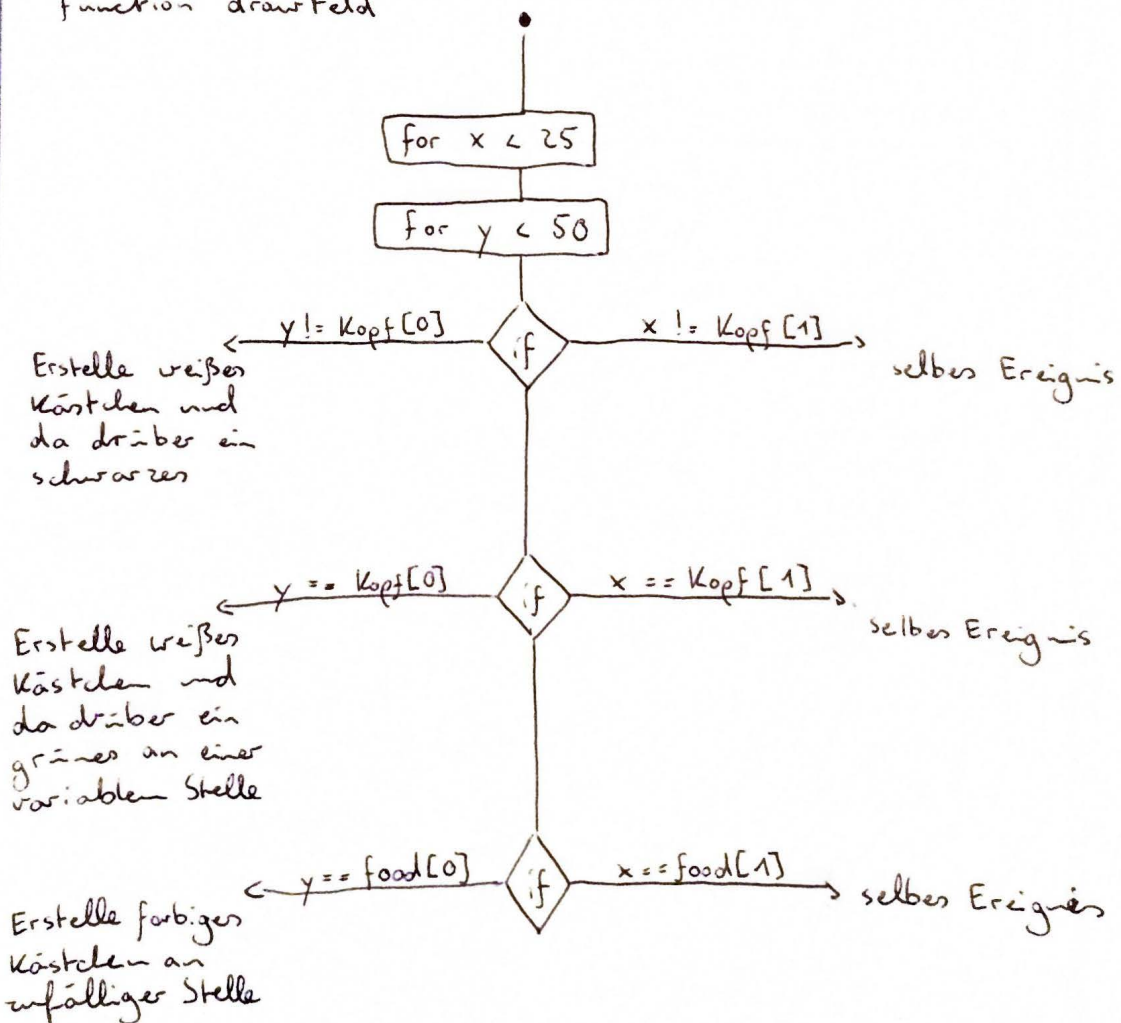
Klasse Kopf



## Steuerung



## function drawField



function generateFood

food = [Math.random \* 49, Math.random \* 24]

function eat

neuer Punkt

neuer Punkt = [Schwanz[i][0], Schwanz[i][1]]

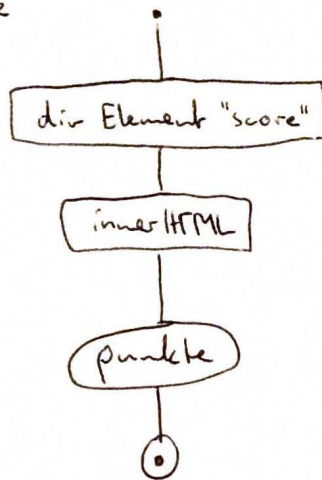
push(neuer Punkt) an KopfAlt und Schwanz

if generate food

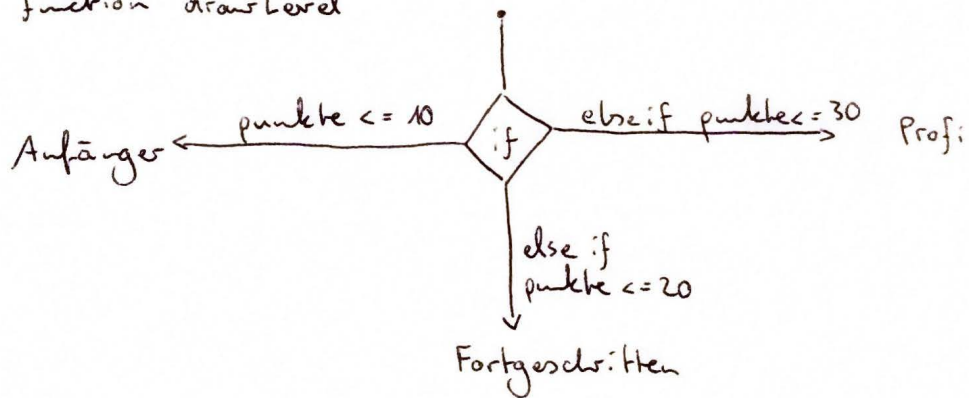
time -= 8

punkte ++

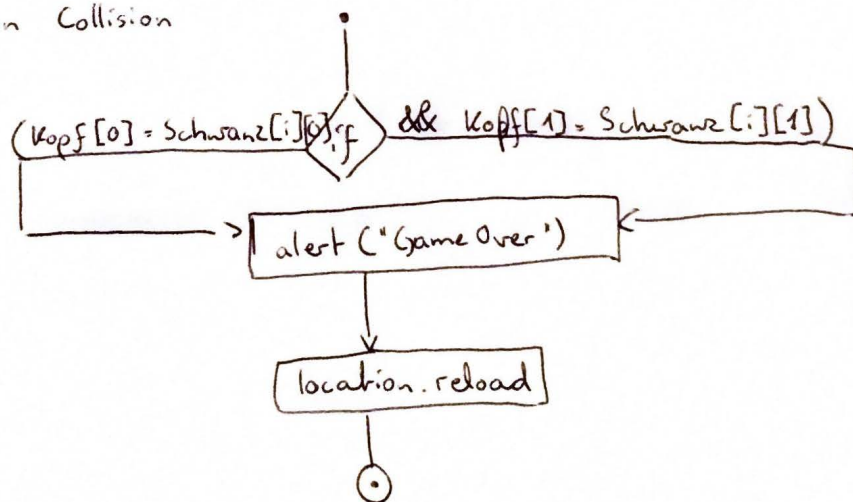
function drawScore



function drawLevel



function Collision



stian animate

