

Übung 03

Hangman mit Vue

SWP-W

5abHWII

November 3, 2022



Übungsleiter: Albert Greinöcker

Ziel der Übung:

- Erstes Erstellen einer Vue-Anwendung (noch ohne `Node.js`)
- Anwenden der Kernkonzepte von Vue

1 Allgemeine Aufgabenstellung

Es soll mittels `Vue.js` ein einfaches Hangman-Spiel mit ein paar Zusatzfeatures erstellt werden. Wenn die Spielregeln nicht klar sind, dann bitte unter <https://de.wikipedia.org/wiki/Galgenm%C3%A4nnchen> nachlesen.

Eine Wortliste als `JSON`-Array und Bilder, die verwendet werden können, liegen bei der Angabe auf Moodle.

2 Vorschläge für die Implementierung

2.1 Einlesen der Wörter

Eine Möglichkeit wäre, die bestehende Wortliste¹ so einzubinden, wie es im ersten Beispiel (FIFA) gemacht wurde. Dafür ist die aktuelle Datei so zu verändern, dass der Inhalt des `JSON`-Arrays einer Variable zugeordnet wird, die dann verwendbar ist. Achtung!: Dann ist sie allerdings noch nicht im Vue-Kontext zu verwenden, das geschieht erst, wenn die Variable im `data`-Bereich nochmals einem eigenen Attribut zugewiesen wird (siehe dazu Bsp.: `01_first/10_fifa_example`).

2.2 Auswahl der Buchstaben

Es besteht die Möglichkeit, die Buchstaben einfach in ein Textfeld eingeben zu lassen. Möchten man allerdings pro Buchstabe einen Button anbieten, kann man sich die Buchstaben von A-Z mit der altbekannten Methode `map` erzeugen lassen:

```
1 letters : Array.from(Array(26)).map((e, i) => String.fromCharCode(i + 65)),
```

¹Es kann natürlich auch eine eigene erzeugt und verwendet werden

Wenn ein bestimmter Buchstabe gewählt wird, wird dieser in der Spiellogik weiterverarbeitet und kann danach aus der Liste `letters` entfernt werden (z.B. mit der `filter`-Methode). Vue sollte dann der Button automatisch verschwinden lassen.

Die Buttons werden dann z.B. so erzeugt:

```
1 <button v-for="letter in letters">{{letter}}</button>
```

2.3 Highscore

Wie auch schon im Spiel davor soll ein Highscore basierend auf der benötigten Zeit und dem erreichten Level berechnet und angezeigt werden. Dazu ist es auch wieder notwendig, einen Timer mit `setInterval` laufen zu lassen.

2.4 Highscore-Tabelle

Es soll eine Highscore - Tabelle im Localstore des Browsers abgelegt werden. Es ist empfohlen diese Daten als JSON-Strings abzuspeichern (siehe Folienbeispiel Javascript). So können auch ältere Highscore-Einträge dargestellt werden.

2.5 Cheating

Um die Entwicklung leichter zu machen, kann man mittels cheat-Taste (frei wählbar - Tastendruck siehe voriges Beispiel) das zu erratende Wort anzeigen. Vorgangsweise: In der Vue-Instanz gibt es eine Variable (z.B. `show`). Die Anzeige basiert dann abhängig von dieser Variable:

```
1 <span v-if="cheat">{{word}}</span>
```

Besser ist es hier gleich das Attribut `key` und nicht `keyCode` (wie im vorigen Beispiel) zu verwenden.

Das Registrieren des `keyListeners` sollte am Besten in der methode `mounted()` passieren.

2.6 Masking und Anzeige der bereits geratenen Buchstaben

Natürlich sollen nur die Buchstaben angezeigt werden, die auch schon ausgewählt wurden. Die Maskierung Zusätzlich sollten alle Buchstaben, die schon geraten wurden, auch angezeigt werden.

Grundsätzlich ist bei der Umsetzung alles möglich, es soll halt die bekannte Spiellogik beinhalten.