



Fight Fakes

Entwicklerdokumentation | Call 18 | Projekt ID 6823

Lizenz: CC BY 4.0.

1 Projektziel

Laut der von Saferinternet.at durchgeführten Studie [„Jugendliche und Falschinformationen im Internet“](#) fällt es 70% der Jugendlichen in Österreich schwer Informationen im Internet zu bewerten. Erst wenn Informationen unglaubwürdig erscheinen, werden sie von ihnen überprüft. Das ist wenig überraschend angesichts der Informationsflut durch TikTok & Co. – es wäre nicht machbar, alle Inhalte zu prüfen. Der Boom generativer Künstlicher Intelligenz verschärft das Problem durch das vermehrte Aufkommen von Deep Fakes.

Gleichzeitig fehlen geeignete deutschsprachige Materialien, um mit Jugendlichen an der bessere Informationsbewertung in Zeiten sozialer Medien und Deep Fakes zu arbeiten. Das Ziel des vorliegenden Projekts ist diese Lücke zu schließen, indem eine Lernumgebung entwickelt wird, in der Jugendliche spielerisch lernen, Informationen schnell und intuitiv zu bewerten.

Hier setzen wir mit dem Projekt „Fight Fakes“ an und konnten im Projektzeitraum eine quizähnliche Spielumgebung entwickeln, in der Jugendliche spielerisch lernen, Informationen schnell und intuitiv zu bewerten und Lehrkräfte ein einfach zu bedienendes Tool an die Hand geben, um das Thema spielerisch im Unterricht zu behandeln.

Die Spiele können spontan - also ohne Vorkenntnisse und Vorbereitung - eingesetzt werden. Der modulare Aufbau mit vielen kurzen Spielen ermöglicht zeitliche Flexibilität - optional können auch weiterführende Begleitmaterialien zur Vertiefung des Themas eingesetzt werden.

„Fight Fakes“ ist somit ein Lernspiel, das Jugendlichen hilft, ihre Intuition zu schulen und digitale Desinformation besser zu erkennen. Es verbindet aktuelles Wissen über Deep Fakes mit praktischen Übungen in einer spielbasierten Lernumgebung.

Es wurden zwei unterschiedliche Spielmechaniken entwickelt, die sich an der Lebenswirklichkeit von Jugendlichen orientieren: das Swipe Game und das Chat Game. Im Folgenden werden die Konfigurationsmöglichkeiten der einzelnen Spiele dokumentiert.

2 Swipe Game

<https://github.com/oiat/SwipeGame/blob/main/README.md>

Das Swipe Game präsentiert nacheinander Karten, die die Spieler:innen nach rechts (richtig) oder links (falsch) swipen können.

Konfiguration

Design

The style of the game can be customised in the settings.json file. The following options are available:

- main-color `CSS Color`
background colour
- text-color `CSS Color`
text colour
- colorfeedbacktrue `CSS Color`
feedback color for correct answers
- colorfeedbackfalse `CSS Color`
feedback color for wrong answers
- welcome
splash screen: `banner` is optional, set to `false` to hide

The images, icons, and background music that are included in the game folder can be replaced with other files of the same format.

Settings

The game settings can also be customised in the settings.json file.

- randomizeQuestions `bool`
shows cards in random order
- questionwait `int`
time between two cards
- timer `bool`
activate time limit for cards
- questiontime `int`
time limit for cards in ms if timer true
- colorfeedback `bool`
displays coloured feedback for correct/incorrect answers
- showscore `bool`
displays the score in the game

- truebutton / falsebutton `string`
emoji for correct/incorrect buttons
- penalty `bool`
wrong answers will result in penalty points
- maxquestions `int`
displays a maximum of x cards, only useful in conjunction with randomizeQuestions
- mute `bool`
deactivates background music

Questions and answers

The cards can be configured using the questions.json file. The file contains a 'questions' object which can contain any number of card objects. Structure of a card object:

```
1. 1. {
2. 2.   "question": "Pixelshakes is the coolest thing ever",
3. 3.   "img": "D0C4F3E7-CDA6-492A-9494-37D7ED09182A.png",
4. 4.   "feedback": {
5. 5.     "true": {
6. 6.       "text": "Heck yeah it is!"
7. 7.     },
8. 8.     "false": {
9. 9.       "text": "Oh... well, to each their own..."
10.10.    }
11.11.  },
12.12.  "points": 100,
13.13.  "correct": true
14.14. }
```

- question `string`
text on the card
- img `string optional`
Image on the card. Ignored if video is set
- vid `string optional`
video on the card
- feedback `object optional`
displays feedback depending on whether the question was answered correctly or incorrectly, can contain text and/or img
- points `int`
points for this card
- correct `bool`
card should be swiped to the right (correct)

3 Chat Game

<https://github.com/oiat/ChatGame/blob/main/README-de.md>

In diesem Spiel wird ein Chat simuliert um Textnachrichten, Bilder und Videos zu versenden und von den Spieler:innen beantworten zu lassen.

Konfiguration

Design

In der Datei settings.json kann der Style des Games angepasst werden. Folgende Einstellungen stehen zur Verfügung:

- main-color `CSS Color`
Farbe des Hintergrunds und deiner Nachrichten
- accent-color `CSS Color`
Farbe für Akzente und Spielernachrichten
- text-color `CSS Color`
Schriftfarbe deiner Nachrichten
- text-color2 `CSS Color`
Schriftfarbe der Spielernachrichten
- welcome
Inhalt des Startbildschirm. `banner` ist optional, zum Ausblenden auf `false` setzen.

Die im Spielordner enthaltenen Bilder, Icons und Hintergrundmusik können durch andere Dateien im selben Format ersetzt werden.

Einstellungen

In der Datei settings.json können ebenfalls die Einstellungen des Spiels angepasst werden.

- sender `string`
Name des Chatpartners, wird im Spiel angezeigt
- randomizeanswer `bool`
Zeigt die Antworten in zufälliger Reihenfolge an
- questionwait `int`
Zeit zwischen 2 Fragen in ms
- colorfeedback `bool`
Zeigt Emoji-Reaktionen für richtige/falsche Antworten
- showscore `bool`
Zeigt den Punktestand im Spiel an
- colorfeedbacktrue / colorfeedbackfalse `string`
Emoji das als jeweilige Reaktion verwendet wird

- mute `bool`
deaktiviert die Hintergrundmusik

Fragen und Antworten

Die Chatnachrichten können über die Datei `questions.json` konfiguriert werden. Die Datei enthält ein object "questions" das beliebig viele Frage-Objekte enthalten kann. Struktur eines Fragenobjekts:

```
1. {
2.   "question": "Hey there! Do you know what Pixelshakes is?",
3.   "feedback": {
4.     "true": {
5.       "text": "Yes exactly!"
6.     },
7.     "false": {
8.       "text": "Pixelshakes is an easy to use tool for creating custom
marketing games."
9.     }
10.  },
11.  "answers": [
12.    {
13.      "text": "Yes, it\u0027s an easy to use tool for creating custom
marketing games.",
14.      "points": 5
15.    },
16.    {
17.      "text": "Pixelshakes? Sounds yummy, i think it\u0027s like a desert?",
18.      "points": -3
19.    },
20.    {
21.      "text": "Never heard of it...",
22.      "points": -1
23.    }
24.  ]
25. }
```

- question `string`
Nachricht an den Spieler
- img `string` optional
Bild als Nachricht. Wird ignoriert falls Video gesetzt wird
- vid `string` optional
Video als Nachricht
- feedback `object` optional
Sendet eine Nachricht, je nachdem ob die Frage richtig oder falsch beantwortet wurde.
kann text und/oder img enthalten

- answers **array**
Antwortmöglichkeiten die dem Spieler zur Verfügung stehen. Jede Antwort als object :
 - text **string** optional
Nachricht
 - img **string** optional
Bild. Text und Bild können kombiniert werden
 - points **int** Punkte dieser Antwortmöglichkeit. > 0 wird als richtige Antwort gewertet (siehe feedback)