



Anleitungen zur Nutzung und Adaptierung von interaktiven Lernspielen auf HTML-Basis mittels Vibe-Coding

Die Digitale Flucht

Schulstufe	Alle Schulstufen
Unterrichtsfach	Alle Unterrichtsgegenstände
Dauer	Flexibel nutzbar (mind. 15 Minuten)
Material	<ul style="list-style-type: none"> • Digitales Endgerät • Eventuell Papier und Bleistift

Inhalt

Die Spielanleitung.....	1
Nutzung durch Lernende.....	1
Anwendungskontext und Rahmenbedingungen.....	2
Anleitung zur Individualisierung des Spielinhalts.....	3
Dokumentation der Erstellung und Weiterentwicklung	5
Verfügbarkeit & Zugang	7

Die Spielanleitung

Im Spiel „Digitale Flucht“ stellen sich die Schüler*innen in Einzelarbeit der Herausforderung, um die Sicherheit ihrer Schule zu gewährleisten. Das Szenario: Das gesamte Schulnetzwerk ist durch eine digitale Bedrohung blockiert und kann nur dann wieder freigeschaltet werden, wenn alle Rätsel gelöst werden. Der Weg führt die Schüler*innen durch insgesamt acht Themenblöcke, in denen jeweils drei Fragen beantwortet werden müssen.

Aus den korrekten Lösungen jedes Blocks müssen die Schüler*innen anschließend einen drei- bis vierstelligen Code ermitteln. Erst durch die Eingabe des richtigen Codes wird der Zugang zum jeweils nächsten Themenbereich freigegeben. Das Ziel ist erreicht, sobald alle acht Blöcke durchgearbeitet wurden: Mit dem Lösen des letzten Rätsels ist die Bedrohung abgewendet und das Schulnetzwerk für alle wieder frei.

Nutzung durch Lernende

Das Spiel steht als HTML-Dateien auf GitHub <https://tina-spi.github.io/OER-Spiele/> sowie im Repository der PH Niederösterreich unter folgendem Link <https://pub.ph-noe.ac.at/id/eprint/85/> zum Download zur Verfügung. Auf den meisten digitalen Endgeräten können diese Dateien direkt in einem Webbrowser geöffnet werden, wenn diese lokal gespeichert sind.

Für den Einsatz im Unterricht empfiehlt sich folgender Ablauf: Die HTML-Datei wird den Schüler*innen über eine Lernplattform zur Verfügung gestellt. Die Schüler*innen laden die Datei auf ihr digitales Endgerät herunter und können sie anschließend per Doppelklick in einem Webbrowser öffnen. Eine Internetverbindung ist für die Nutzung nicht erforderlich, wenn die Datei lokal gespeichert ist.

Einige Betriebssysteme (zB iOS, iPadOS) verhindern aus Sicherheitsgründen das lokale Ausführen von interaktiven HTML-Dateien. In diesen Fällen muss das Spiel über einen Server bereitgestellt werden. Eine einfache Möglichkeit dafür bietet **GitHub Pages**. Die HTML-Dateien können so gehostet und von den Schüler*innen über den Link direkt im Webbrowser geöffnet werden. Eine Anleitung zur Einrichtung von GitHub Pages ist im Repository der PH Niederösterreich verfügbar: <https://pub.ph-noe.ac.at/id/eprint/83>

Anwendungskontext und Rahmenbedingungen

Lizenzbestimmungen

Diese Anwendung sowie die dazugehörige Dokumentation sind als Open Educational Resources (OER) veröffentlicht. Die Veröffentlichung erfolgt unter der Creative-Commons-Lizenz **CC BY-NC-SA 4.0**. Dadurch die freie Nutzung, Vervielfältigung und Bearbeitung der Inhalte unter folgenden Bedingungen möglich:

- **Namensnennung:** Bei der Nutzung und Weitergabe sind die Urheberinnen zu nennen.
- **Weitergabe unter gleichen Bedingungen:** Remixe oder auf diesem Material basierende Bearbeitungen müssen unter derselben Lizenz veröffentlicht werden.
- **Nicht-kommerzielle Nutzung:** Als OER ist die Anwendung für freie Nutzung im Bildungsbereich bestimmt. Eine kommerzielle Verwertung oder der Verkauf der Inhalte ist ausgeschlossen, um den offenen Zugang dauerhaft zu gewährleisten.

Mit der Nutzung der Applikation werden die Lizenzbedingungen anerkannt. Insbesondere bei der im Leitfaden beschriebenen inhaltlichen Anpassung (Individualisierung) ist sicherzustellen, dass die entstandenen Werke ebenfalls unter der Lizenz CC BY-NC-SA 4.0 genutzt und veröffentlicht werden.

Weitere Informationen zu CC-Lizenzen sind auf <https://de.creativecommons.net/start/> abrufbar.

Einsatzmöglichkeit des vorliegenden Spiels

Die aktuell vorliegende Version eignet sich für den Einsatz in der 8. Schulstufe im Fach Digitale Grundbildung.

Einsatzszenarien nach Individualisierung des Spielinhalts

Durch die individuelle Anpassung der Spielinhalte lässt sich die Applikation in nahezu allen Schulstufen und Gegenständen einsetzen. Die didaktische Einbettung ist dabei flexibel: Je nach Komplexität des Fragenpools und dem Vorwissen der Lernenden ist eine entsprechende Zeitspanne im Unterrichtsverlauf einzuplanen. Es empfiehlt sich, die Anwendung in mehreren Webbrowsers sowie auf verschiedenen Endgeräten zu testen, um eine fehlerfreie Nutzung auf unterschiedlichen Endgeräten sowie in verschiedenen Webbrowsers gewährleisten.

Anleitung zur Individualisierung des Spielinhalts

Geeignete Applikationen zur Bearbeitung

Die Inhalte des Spiels lassen sich mit jedem Editor, mit dem HTML-Code bearbeitet werden kann, anpassen. Eine geeignete Applikation ist beispielsweise **Notepad++**.

Bearbeitet man HTML-Code im kostenfreien Editor Notepad++ (Open-Source-Software; GNU General Public License), wird der Syntax farblich hervorgehoben, sodass die Struktur gut erkennbar ist. Notepad++ verfügt über eine automatische Vervollständigung, sodass beim Bearbeiten passenden Tags und Attribute vorgeschlagen werden. Weiters sind die Zeilen des Codes durchgängig nummeriert, was die Navigation im Dokument erleichtert.

Unter der URL <https://notepad-plus-plus.org/> kann das Programm heruntergeladen werden. Auf dieser Seite ist auch eine umfangreiche Dokumentation samt User-Manual verfügbar.

Hinweise zu relevanten Codezeilen

Die in den folgenden Abschnitten angegebenen Zeilennummern beziehen sich auf die Zeilennummerierung, wie sie in Notepad++ angezeigt wird. Bei der Adaptierung des HTML-Codes ist darauf zu achten, da andere Editoren oder Darstellungsformen abweichende Nummerierungen aufweisen können.

Schrittweise Anleitung

1. Generierung des fachspezifischen Fragenpools

Unter Verwendung eines generativen KI-Modells (genKI) wird ein individueller Fragenkatalog erstellt. Dabei ist auf eine präzise Prompt-Formulierung zu achten, damit der generierte Fragenpool dem aktuellen Code-Format entsprechen.

Mit folgendem **Musterprompt** kann dies KI-gestützt erzeugt werden:

Rolle: Du bist eine Expertin für **[THEMA]** in der **[SCHULSTUFE]**.

Aufgabe: Erstelle für die folgenden 8 Themenblöcke jeweils genau 3 Testfragen. Das Ziel ist es, dass die Schülerinnen pro Block einen 3 bis 4-stelligen Zahlencode generieren, um ihre Antworten zu überprüfen.

Themenblöcke: **[THEMEN AUFLISTEN]**

Methodik der Code-Erstellung (pro Block gemischt anwenden) :

- Zähl-Methode: „Wie viele der folgenden Aussagen sind richtig?“ (Die Antwort ist die Ziffer).
- Multiple-Choice-Methode: „Welche Antwort ist korrekt? 1) X, 2) Y, 3) Z.“ (Die Nummer der richtigen Antwort ist die Ziffer).
- Buchstaben-Wert-Methode: „Wie lautet der erste Buchstabe des gesuchten Begriffs? Wandle ihn in seine Position im Alphabet um (A=1, B=2...) und nimm die Quersumme oder die letzte Ziffer, falls er zweistellig ist.“

Anforderung: Erstelle eine übersichtliche Liste. Nenne das Thema, die drei Fragen inklusive Antwortmöglichkeiten und am Ende des Blocks den 3-stelligen Lösungscode.

Abbildung 1: Musterprompt für das Erstellen eines formatierten Fragenpools

2. Fachliche und inhaltliche Prüfung

Die durch die KI-gestützt generierten Inhalte sind einer fachlichen Prüfung zu unterziehen. Hierbei muss die fachliche Relevanz sowie die Korrektheit der Antworten überprüft und bei Bedarf manuell angepasst werden.

3. Integration in den Quellcode

Der finale Fragenpool wird in den dafür vorgegebenen Bereich des Quelltextes übertragen. Diese sind im Quellcode in den **Zeilen 186 bis 410** vorzunehmen.

Der Titel und das Intro sowie der Final-Screen können in den **Zeilen 147 bis 163** bzw. in den **Zeilen 174 bis 181** adaptiert werden.

4. Aktualisierung der Metadaten und Lizenzangaben

Gemäß der Bedingungen der CC BY-NC-SA 4.0-Lizenzierung sind die Urheberangaben im Quellcode zu adaptieren. Diese sind im Quellcode in den **Zeilen 183 bis 186** vorzunehmen.

Die derzeit vergebene Lizenz muss wie folgt abgeändert werden:

[TITEL NACH BEARBEITUNG] © **[Jahr]** by **[EIGENER NAME]** is licensed under CC BY-NC-SA 4.0. Basierend auf „Die Digitale Flucht“ © 2026 by Martina Spitaler & Claudia Petz (CC BY-NC-SA 4.0, <https://creativecommons.org/licenses/by-nc-sa/4.0/>). Es wurden Änderungen vorgenommen.

Hinweis: Die **[HERVORGEHOBENEN TEXTPASSAGEN]** müssen durch eigene Inhalte ersetzt werden.

5. Finalisierung und Sicherung

Nach Abschluss der Modifikation wird die HTML-Datei lokal gesichert. Zur Überprüfung der Funktionalität muss ein Testlauf in einem gängigen Webbrowser durchgeführt werden.

Dokumentation der Erstellung und Weiterentwicklung

Verwendetes genKI-Modell

Das vorliegende interaktive Unterrichtsspiel wurde unter Verwendung der generativen KI-Applikation Google Gemini 3.1 Pro (Google, 2026) am 16.03.2026 erstellt. Die Umsetzung erfolgte nach dem Prinzip des Vibe Codings, bei dem natürliche Sprache zur KI-gestützten Generierung von Programmiercode genutzt wird. Dabei wird das Ergebnis in Form eines Prompts beschrieben, der KI-generierte Output geprüft und gegebenenfalls iterativ verbessert.

Verwendeter Prompt

Rolle & Ziel:

Du bist ein erfahrener Programmierer. Erstelle eine interaktive Single-Page-Webanwendung (HTML, CSS, JavaScript) für ein pädagogisches Escape-Game namens "Digitale Flucht".

Design & Styling (CSS):

- **Thema:** Dark Mode, modernes Hacker-Thema.
- **Farben:** Dunkle Hintergründe, Akzentfarben in Neon für Buttons und Fehler, Hellblau für Überschriften, Neongrün für Erfolgsmeldungen.
- **Layout:** Zentrierter Container, responsiv. Abgerundete Ecken, sanfte Schatten und halbtransparente Boxen.
- **Typografie:** Serifenlose, gut lesbare Schriftart.

Spiellogik (JavaScript & HTML-Struktur):

1. **Intro-Screen:** Zeigt eine Story ("Alarm im Computerraum: Das Schulnetzwerk wird von Hackern bedroht"), eine kurze Spielanleitung und einen "Mission starten"-Button.
2. **Spiel-Screen:**
 - Beinhaltet oben eine Statusleiste (Mission, Fortschritt z.B. "1 / 8", Freigeschaltete Codes).
 - Zeigt immer nur einen Fragenblock.
3. **Aufbau eines Fragenblocks:**
 - Titel mit passendem Emoji.
 - 3 Fragen pro Block (Mix aus Multiple-Choice mit Radio-Buttons und reinen Zahlen-Textfeldern).
 - Unten ein Eingabefeld für den "Zifferencode" (3-4 Zeichen) und einen Button "Block prüfen".
4. **Prüf-Logik:**
 - Der Code ergibt sich, indem man die richtigen Antworten der 3 Fragen eingibt (z.B. Antwort 1 = "4", Antwort 2 = "4", Antwort 3 = "16" -> Code: "4416").
 - Wenn der Code falsch ist oder Antworten falsch markiert sind, erscheint eine rote Fehlermeldung.
 - Wenn alles stimmt, erscheint kurz eine grüne Erfolgsmeldung, dann lädt automatisch der nächste Block.
5. **Erfolgs-Screen:** Wenn alle 8 Blöcke gelöst sind, erscheint eine Gratulation ("Netzwerk gesichert") und ein Button zum erneuten Spielen (Page Reload).

Inhalte (Die 8 Fragenblöcke): siehe Dokument

Schreibe sauberen, gut lesbaren Code und stelle sicher, dass alle JavaScript-Funktionen reibungslos ineinandergreifen.

Abbildung 2: verwendeter Prompt für das Erstellen des Spiels

Versionshistorie und Änderungsverlauf

Element	Art der Änderung	Methode / Werkzeug
Fragenpool: Ideensammlung	Generierung von Fragen für das Spiel	Google Gemini 3.1 Pro (Google, 2026)
Fragenpool: erstellen	Erstellung des Fragenpools	Manuell
Vibe-Coding	Erstellen des Spiels	Google Gemini 3.1 Pro (Google, 2026)
Layout der Fragen	Adaption der Darstellung der Fragen (Zeilenumbrüche, Hilfestellungen)	Manuelle Adaption
Rechtliche Hinweise	Ergänzung der Creative-Commons-Lizenz (CC) im Footer	Manuelle Ergänzung

Grenzen des Vibe-Coding

Trotz der Vorteile des Vibe Codings sind dessen Grenzen zu berücksichtigen. KI-generierter Code entspricht nicht immer den gewünschten Anforderungen und muss sorgfältig geprüft werden. Bei komplexen oder umfangreichen Projekten steigt die Fehleranfälligkeit, da KI-Applikationen den Gesamtkontext nur begrenzt erfassen können. Ohne grundlegende Kenntnisse in der Webentwicklung ist es zudem schwierig, den generierten Output kritisch zu bewerten oder gezielt zu korrigieren insbesondere hinsichtlich Code-Qualität und Sicherheit. Vibe Coding ersetzt daher keine fundierten Programmierkenntnisse.

Verfügbarkeit & Zugang

Das beschriebene Lernspiel sowie die zugehörige Dokumentation werden als Open Educational Resource (OER) offen zugänglich bereitgestellt. Sie können über die GitHub-Plattform unter <https://tina-spi.github.io/OER-Spiele/> sowie über das Open Access Repository der Pädagogischen Hochschule Niederösterreich unter <https://pub.ph-noe.ac.at/id/eprint/85/> abgerufen, heruntergeladen und weiterverwendet werden.