

Wegleitung PyGame-Project



In diesem Dokument finden Sie die notwendigen Informationen, die Sie zur Vorbereitung und Durchführung dieser Projektarbeit benötigen. Bei Fragen wenden Sie sich an die Lehrperson.

1) Allgemeine Rahmenbedingungen

Ziel dieser Arbeit ist es, ein Programmierprojekt in einer Gruppe zu realisieren und dokumentieren. Die Abgabe des Projektes besteht aus 3 Teilen, den sogenannten «Meilensteinen». Diese sollen Sie durch das Projekt leiten und dabei helfen, den Fortschritt des Projektes zu sichern. Alle Meilensteine fließen in die Bewertung ein.

Meilenstein 1	Abgabe Projektskizze & Logfiles 01	Fr, 24.11.23 17:30 Uhr
Meilenstein 2	Abgabe Logfiles 02, Game-Demo (Prototyp)	Mi, 13.12.23 23:59 Uhr
Meilenstein 3	Finale Abgabe: Code, Actionvideo und Logfiles 01-03	Fr, 19.01.24 23:59 Uhr

2) Produktanforderungen

Sie entwickeln mit PyGame eine eigene Variation eines Games. Ob Sie dabei eine eigene Spielidee von Grund auf kreieren oder sich anhand eines Tutorials an einem bestehenden Spiel orientieren, ist Ihnen freigestellt.

Ihr Game muss folgende Kriterien erfüllen:

1. Ein Spiel, welches durch **Interaktion** mit einem*einer oder mehreren Spieler*innen gespielt wird.
2. Ein individueller **Startscreen**, auf welchem das Spiel gestartet werden kann und mögliche weitere Aktionen (Spielerklärung o.Ä.) bereithält.
3. Ein individueller **Endscreen** bei Beendigung des Spiels (Score, Game-Over o.Ä. anzeigen) mit möglichen Aktionen (Neustart, Beenden, Highscore etc.).
4. Die drei Stadien (Startscreen, Game und Endscreen) sind logisch über **Buttons** miteinander verknüpft. Das Spiel lässt sich über ein Hauptdokument namens **main.py** starten.
5. Eigene **akustische und visuelle Elemente** (Musik, Bilder, etc.)
6. Bei **3er-Gruppen** ist zusätzlich ein **Fullscreen-Modus** und ein beständiger **Highscore** vorhanden.
7. **Der persönliche Anstrich:** Eigene Komponenten, welche das Basisspiel erweitern.¹

Genauere Anforderungen zu den einzelnen Meilensteinabgaben sind im nächsten Abschnitt beschrieben.



Alle Quellen (Webquellen, AI- oder andere Tools, helfende Personen) sind strikt in den Logfiles und im Code anzugeben!

Eine 1:1 Umsetzung eines Tutorials mit unvollständiger Quellenangabe wird nicht akzeptiert. Ein Plagiat führt zu Notenabzug und kann bis zur Note 1 gewertet werden. Beim Nutzen von AI-Tools geben Sie zu der Quelle den Befehl (Prompt) an, welcher Sie zum verwendeten Code geführt hat.

¹ Erweiterungsmöglichkeiten wären z.B. Zusatzfunktionen im Spiel oder Features wie einen Highscore, zusätzliche Zeitangabe, Fullscreen, Spiel pausieren, Möglichkeit einer Spielanleitung auf dem Startscreen, verschiedene Schwierigkeitsstufen des Spiels usw. Diese Erweiterung müssen Sie nicht von Grund auf selbst programmieren. Sie können wiederum Tutorials nutzen und mit ihrem bisherigen Spiel kombinieren.

3) Projektskizze

In der Projektskizze sind Ihre Zielsetzung (inkl. Link), Ihr geplanter Arbeitsplan, die geplante Arbeitsaufteilung unter den Gruppenmitgliedern sowie eine grobe Skizze der äusseren Erscheinung des Produktes enthalten. Eine Auseinandersetzung mit der Funktionsweise (Spiellogik, formale Bedingungen) des ausgewählten Spieles wird verlangt.

Sie erstellen einen Arbeitsplan für die Projektarbeit. Die vorgegebenen Meilensteine dienen als grobe Orientierung. Denken Sie daran, Zeitpuffer für erwartete Probleme und Zwischenfälle einzuplanen.

Formelles: Verfassen Sie die Inhalte in ganzen Sätzen. Nutzen Sie die Vorlage, welche im Wordformat auf MS-Teams zur Verfügung steht.

4) Logfiles - Projektjournal

Sie notieren alle Arbeitsschritte in den Logfiles. Vermerken Sie im Dokument jeweils welches Gruppenmitglied sich, wann um welche Aufgabe gekümmert hat. Beschreiben Sie, auf welche Probleme Sie gestossen sind und wie Sie diese lösen konnten. Geben Sie alle verwendeten Quellen an.

Formelles: Verfassen Sie die Inhalte in der Vergangenheitsform und in ganzen Sätzen. Wählen Sie eine einheitliche Struktur (z.B. Fliesstext, Stichworte, ...) und verwenden Sie diese im ganzen Dokument. Nutzen Sie die Vorlage, welche im Wordformat auf MS-Teams zur Verfügung steht.

5) Code

- Im Code sind sämtliche Quellen angegeben.
- Generelle Quellen (z.B. Link zum Tutorial) können auf der ersten Zeile des Python Dokuments aufgeführt werden.
- Alle Zeilen, welche Sie von einer solchen Quelle abgeändert haben, werden folgendermassen markiert:

```
111      # Punktezähler
112      if not car_crash:
113          # Alter Code: score += 1 # erhöhe den Punktestand
114          score = score + 2 # Eigene Änderung, Erhöhung um 2
```

Längere Codeabschnitte können mit `'''code'''` über mehrere Zeilen auskommentiert werden

```
138      '''elif event.key == pygame.K_RIGHT: # bewege das Auto nach rechts
139          if my_car_position_x < 2:
140              my_car_position_x += 1'''
```

- Alle eigenen Implementierungen sind mit Kommentaren als solche zu kennzeichnen.

6) Actionvideo

Das Actionvideo befindet sich in einem zeitlichen Rahmen zwischen minimal 3 und maximal 6 Minuten. Dabei zeigen Sie die implementierten Komponenten Ihres Produktes (im Code und im Spielmodus) und kommentieren diese sinnvoll. **Sie müssen nicht den ganzen Code in allen Files zeigen und kommentieren. Gehen Sie hauptsächlich auf die von Ihnen bearbeiteten und ergänzten Stellen ein.**

Beschreiben Sie zudem kurz die kniffligsten Stellen im Arbeitsprozess. Was hat gut funktioniert, was weniger? Eine **kritische Betrachtung der eigenen Arbeit** und des Projektes wird erwartet. Was haben Sie gut gemacht, was weniger? Wo würden Sie rückblickend die eigene Arbeitsweise anpassen?

7) Bewertung

Ihre Lehrperson bewertet Ihr Projekt und teilt Ihnen die erreichte Punktzahl der Gruppe mit. Mit der Hälfte der Maximalpunktzahl wird die Note 6 erreicht (2er Gruppe). Bei der finalen Abgabe geben Sie im Logfile eine prozentuale Aufteilung unter den Gruppenmitgliedern an. Die erreichten Punkte werden anhand dieses

Verhältnisses aufgeteilt, was zur individuellen Note der Gruppenmitglieder führt. Ihre Aufteilung ist als Vorschlag im Verhältnis des geleisteten Arbeitseinsatzes zu verstehen. Die Lehrperson kann insistieren und die Punkte gleichmässig verteilen.

8) Checklisten

Meilenstein 1

- Alle Dokumente sind korrekt beschriftet und im richtigen Format (Snake-Case).
 - Projektskizze: **projektskizze_projektname_name1_name2.pdf**
 - Logfile: **logfile_01_projektname_name1_name2.pdf**
- Die Projektskizze ist vollständig und enthält den «Hauptlink» Ihres Game-Tutorials. Verantwortlichkeiten sind grob eingeteilt (Spiellogik, Start -& Endscreen, Bilder, Sound).
- Sämtliche Quellen (Links und AI-Prompts) sind in Logfiles und im Code an den entsprechenden Stellen angegeben. Eigenleistungen sind als solche gekennzeichnet.

Meilenstein 2

- Alle Dokumente sind korrekt beschriftet und im richtigen Format (Snake-Case).
 - Logfile: **logfile_02_projektname_name1_name2.pdf**
- Game-Architektur: Sie verfügen über ganz einfach gehaltene Start-, Game- und Endscreens, welche eventuell bereits miteinander verknüpft sind.
- Im Gamescreen haben Sie ein lauffähiger Prototyp mit den grundlegendsten Komponenten erstellt.
- Projektplan (Arbeitsschritte) wurde gegebenenfalls angepasst.
- Das Spiel lässt sich jederzeit über die Escape-Taste und den Kreuz-Button schliessen.
- Eigenanteil (eigene Funktionen, Kombinationen von verschiedenen Tutorials als Erweiterung der Spiellogik) wird thematisiert und konkrete Ideen sind vorhanden (inkl. Recherche).
- Sie notieren den Projektfortschritt in den Logfiles und nennen dabei Probleme und deren Lösungsansätze.
- Sämtliche Quellen (Links und AI-Prompts) sind in Logfiles und Code an den entsprechenden Stellen angegeben. Eigenleistungen sind als solche gekennzeichnet.

Meilenstein 3

- Alle Dokumente sind korrekt beschriftet und im richtigen Format (Snake-Case). Speichern Sie sämtliche Dokumente und den gesamten Projektcode in einem Zip-Ordner.
 - Logfile: **logfile_03_projektname_name1_name2.pdf**
 - Actionvideo: **video_projektname_name1_name2.mov oder .mp4**
 - Komprimierter Ordner: **abgabe_projektname_name1_name2.zip**
- Start- und Endscreen zeigen jeweils sinnvolle Möglichkeiten (Start, Neues Spiel, Schliessen, Highscore, Menu, etc.) an und sind nicht identisch.
- Visuelle und akustische Elemente sind passend eingebaut.
- Eigenanteil (eigene Funktionen, Kombinationen von verschiedenen Tutorials als Erweiterung der Spiellogik) sind im Code und den Logfiles ersichtlich und werden im Video explizit beschrieben.
- Spiel lässt sich jederzeit über die Escape-Taste und den Kreuz-Button schliessen.
- Sämtliche Quellen (Links und AI-Prompts) sind in Logfiles und Code an den entsprechenden Stellen angegeben. Eigenleistungen sind als solche gekennzeichnet.
- Der Code ist fehlerfrei und aufgeräumt (eigene Implementierungsansätze, welche nicht funktionsfähig sind, aber viel Arbeit gewesen sind, können Sie auskommentiert mit einem entsprechenden erklärenden Kommentar im Code belassen). Sie kommentieren sämtliche Funktionen und Codeblöcke sinnvoll.
- Das Projekt ist aufgeräumt und enthält keine überflüssigen Files. Sie verwenden Unterordner für zusätzliche Dateien wie Bilder und Sound.