März  
2017

Projekt «Lernspiel»

Projektteam: Marcel Mathis, Pascal Niederberger, Amol Ming

Modul 431 – Gruppenarbeit Projektauftrag

Auftraggeber: Peter Brunner

Inhalt

[Administratives 2](#_Toc477894389)

[Zweck dieses Dokuments 2](#_Toc477894390)

[Projektorganisation 2](#_Toc477894391)

[Wichtige Termindaten 2](#_Toc477894392)

[Ablage 2](#_Toc477894393)

[Änderungsverfolgung 2](#_Toc477894394)

[Handelt es sich um ein Projekt? 3](#_Toc477894395)

[Projektauftrag erstellen 4](#_Toc477894396)

[Kosten-Nutzen Berechnung 5](#_Toc477894397)

[Terminplan erstellen 6](#_Toc477894398)

[Projektüberwachung 8](#_Toc477894399)

# Administratives

## Zweck dieses Dokuments

Dokumentation und Definition des Projektes «Spiel».

## Projektorganisation

Arbeitgeber: Peter Brunner (PBR)

Projektleitung: Amol Ming (AMI)

Projektteam: Marcel Mathis (MMA), Pascal Niederberger (PNI), Amol Ming (AMI)

## Wichtige Termindaten

Projektstart: 08.03.2017

Projektende: 03.05.2017

## Ablage

<https://github.com/JFK422/Modul431>

## Änderungsverfolgung

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Beschreibung** | **Ersteller** | **Version** |
| 10.03.2017 | Dieses Dokument erstellt | AMI | 1.00 |
| 10.03.2017 | Erstellung Projektauftrag | PNI | 1.01 |
| 12.03.2017 | Erstellung Terminplan | MMA | 1.02 |
| 13.03.2017 | Visuelle Überarbeitung | AMI | 1.03 |
| 15.03.2017 | Abgabe an Lenkungsausschuss | Team | 1.04 |
| 20.03.2017 | Visuelle Überarbeitung (Tabelle) | Team | 1.05 |
| 21.03.2017 | Feinschliff | Team | 1.06 |

# Handelt es sich um ein Projekt?

Ein Projekt ist ein einmaliges, zeitliches begrenztes Vorhaben, das bezüglich Umfang, einbezogener Personen und Risikofaktoren eine gewisse Komplexität aufweist.  
Die folgende Tabelle definiert, ob unser Projekt «Lernspiel» tatsächlich ein Projekt ist oder lediglich eine Aufgabe/Auftrag.

|  |  |  |  |
| --- | --- | --- | --- |
| **Kriterium** | **Punktzahl** | | |
|  | **1** | **2** | **3** |
| Projektgrösse | klein | mittel | gross |
| Anzahl Wiederholungen | monatlich | 1-mal jährlich | 1-mal in 5 Jahren |
| Einbezogene Personen | wenige | mittel | viele |
| Know-how vorhanden | ja | ein wenig | nein |
| Risiko | gering | mittel | gross |
|  |  |  |  |
| Total Punkte |  |  | 10 Punkte |

Richtwert: Beträgt die Summe der Punkte für die einzelnen Kriterien mehr als 9 Punkte, so ist die Aufgabe Projektwürdig.

🗹 das Projekt «Lernspiel» ist projektwürdig!

# Projektauftrag erstellen

**Ausgangslage:** Es wird ein Spiel benötigt, welches verschiedene Themen der Informatik behandelt. Das Spiel soll so eingesetzt werden können, dass es zur Erweiterung und Vertiefung des Fachwissens motivierend beiträgt.

**Stärken:** Gute Programmierkenntnisse als auch ein breites Fachwissen sind von Vorteil.

**Schwächen:** Programmieren in HTML.

**Perspektiven:** Wie umfangreich soll das Spiel programmiert werden? Und welcher Kompromiss aus Qualität und Quantität soll beachtet werden?

**Risiken:** Wird das programmierte Spiel funktionieren? Welche Programmiersprache eignet sich dazu am besten?

**Hauptdaten**: Projektname: «Lernspiel»

**Auftraggeber**: Brunner Peter, Lehrperson INF16B, Modul 431

**Projektteam**: Marcel Mathis, Pascal Niederberger, Amol Ming

**Start**: Mittwoch, 08.03.2017

**Ende**: Mittwoch, 03.05.2017

**Projektziel:** Das Spiel muss bis am 03.05.2017 fertiggestellt werden, fehlerfrei funktionieren und den Zweck erfüllen.

**Nutzen:** Es wird einen schnellen und effizienten Weg geschaffen, gezielt Themen der Informatik zu Lernen und verstehen.

**Kosten:** 0 Fr.

# Kosten-Nutzen Berechnung

**Nutzen**: Es wird einen schnellen und effizienten Weg geschaffen, gezielt Themen der Informatik zu lernen und verstehen. Unsere Programmierkenntnisse werden im Verlauf des Projektes erweitert und unser Fachwissen über Projektbearbeitung wird gestärkt.

**Kosten:** Es werden keine finanziellen Mittel benötigt, lediglich ein Zeitaufwand von 34 Tagen, welcher noch detaillierter im Zeitplan nachlesbar ist.

# Terminplan erstellen

|  |  |  |
| --- | --- | --- |
| **Tätigkeit** | **Aufwand** | **Verantwortlich** |
| Spielstruktur besprechen & planen | 3 Tage | Team |
| Programmiersprache definieren | 1 Tag | Team |
| Lernressourcen beschaffen (Spielinhalt) | 7 Tage | Amol Ming |
| Grundstruktur des Programms entwickeln | 10 Tage | Marcel Mathis |
| Lernressourcen ins Programm integrieren | 7 Tage | Marcel Mathis |
| Programm testen | 2 Tage | Pascal Niederberger |
| Programm verfeinern, optimieren | 3 Tage | Pascal Niederberger |
| Programm fertigstellen | 1 Tag | Pascal Niederberger |



# Projektüberwachung

**Terminkontrolle**

Balkendiagramm des Terminplans wird täglich überprüft, um den Zeitplan gegebenenfalls anzugleichen. Wöchentlich wird Teamintern nochmals nachgefasst.  
Der Fortschritt wird regelmässig überprüft, um einen Rückstand zu vermeiden.

**Qualitätskontrolle**

Nach Abschluss einer Tätigkeit wird mit dem 4-Augen Prinzip gearbeitet, um die gemachte Arbeit in technischer Sicht zu überprüfen. In diesem Prozess, werden die beiden Teammitglieder die Arbeit überprüfen, welche nicht in die Tätigkeit involviert sind.

**Kostenkontrolle**

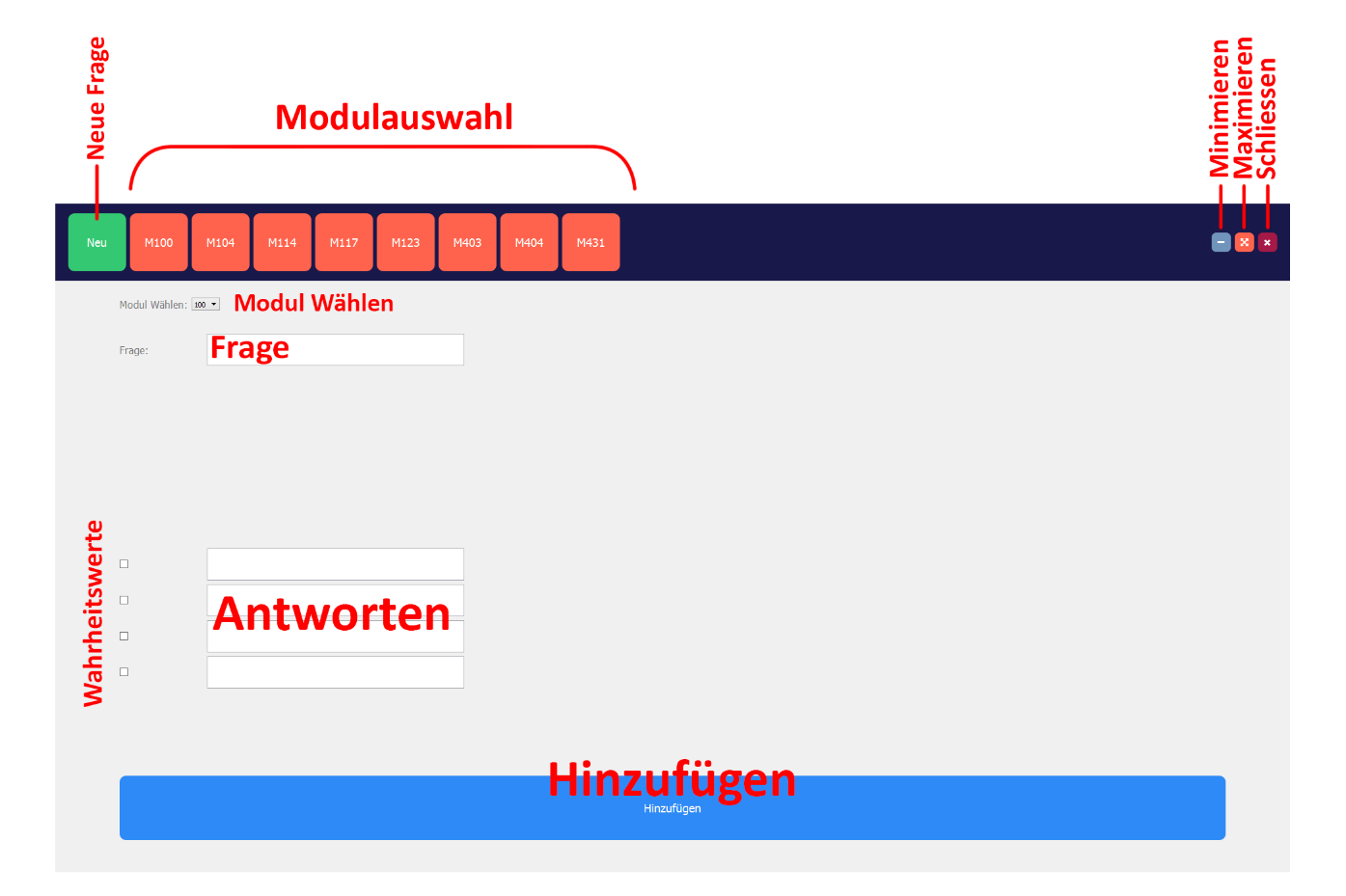
Da keine finanziellen Kosten generiert werden, fällt die Kostenkontrolle weg.

# Dokumentation

## Allgemeines zum Lernspiel

Unser Lernspiel ist sehr einfach aufgebaut. Man kann Fragen hinzufügen, welche jeweils 4 Antwortmöglichkeiten haben. Die Fragen sind unterteilt in die acht Module, welche wir bereits kennen. Man kann Fragen hinzufügen, ohne den Programm-Code zu verändern. Das Programm macht also alles automatisch. Nachdem man Fragen zu einem bestimmten Modul beantwortet hat, bekommt man eine Auswertung, in der steht, wie viele Fragen man richtig und wie viele man falsch beantwortet hat.

## Die Programmoberfläche



## Funktionsweise des Programms

Das Programm wurde in der Programmiersprache Python programmiert. Dabei wurden noch die Module PyQt5 (Gui Erstellung), QtAwesome (Icons) und cx\_freeze (ausgeben des Skripts als .exe) verwendet. Das Skript wurde hierfür in zwei Teilen programmiert. Diese sind index.py als Hauptskript und fileManager.py als Skript zum Speichern und Laden der Fragen. Wenn man eine Frage hinzufügt, wird diese mittels dem fileManager im Ordner des Moduls, nach Frage, Optionen und Antworten getrennt gespeichert. Das Skript wurde für 32Bit Betriebssysteme ausgegeben. Welches auch auf 64Bit Systemen läuft. Das Lernspiel umfasst etwa 500 Zeilen an Code.