

Best Practice Beispiel

Ereignisorientierte prüfungsrelevante Lernziele (EPL) im Modul Informatik

Eckdaten zur Veranstaltung

| | |
|---------------------------|---|
| <u>Lehrveranstaltung:</u> | Vorlesung und Praktikum bzw. Übung (ganzes Modul) |
| <u>Zeitlicher Umfang:</u> | 2 Semester |
| <u>Lehrperson:</u> | Prof. Dr. Barbara Wieczorek |
| <u>Fachbereich:</u> | Grundlagenwissenschaften |

Kurzbeschreibung

Die Methode EPL wird ausführlich beschrieben in: (Wieczorek und Ulrich 2022). Sie ist an das „klassische“ Constructive Alignment angelehnt, setzt dieses jedoch systematisch

- je Lerneinheit (bezeichnet als Ereignis),
- je individuellem Lernziel

um. Dies soll zu einer hohen Transparenz hinsichtlich der Prüfungsanforderungen führen und eine zielgenaue Prüfungsvorbereitung ermöglichen.

Jedem Lernziel der Veranstaltung werden dabei die Ereignisse zugeordnet, die im Dienste der Erreichung dieses Lernziels stehen.

Den Studierenden liegt diese Zuordnung in zwei Formaten vor:

- Es steht eine tabellarische Übersicht in einem pdf-Dokument zur Verfügung (s. Abbildung 1).
- Zusätzlich gibt es einen Fragebogen in Moodle, wo für jedes Lernziel die Selbsteinschätzung zum Stand der Erreichung gespeichert werden kann (s. Abbildung 6). Insbesondere in diesem Format können Studierende die EPL als *konstantes Feedbackinstrument* zu ihrem Lernstand nutzen, was sich positiv auf ihr Lernergebnis auswirken kann.

Wesentlich ist dabei die operationalisierte Formulierung der Lernziele, um unmittelbar mögliche Prüfungsaufgaben ableiten zu können.

Zur Beschreibung der verschiedenen in den EPL verwendeten Ereignisse:

Im konkreten Beispiel werden sowohl Ereignisse des Präsenz- wie auch des Selbststudiums einbezogen. Der Einbezug des Selbststudiums ist im Allgemeinen aber nicht notwendig, um EPL zu erstellen.

Im hier genannten Modul erfolgt die hauptsächliche Vermittlung von Inhalten in **Vorlesungen (V)**. Dabei werden aktivierende Elemente wie kurze Programmierübungen und Klicker-Fragen einbezogen.

Eine Vertiefung und Erweiterung erfolgt im Selbststudium zunächst in Form von

Nachbereitungsaufgaben (VNb) zur Vorlesung, welche im Learning-Management-System Moodle zur Verfügung stehen und automatisiert ausgewertet werden.

Der Schwerpunkt auf praktische Programmierung wird in Praktika bzw. Übungen gelegt, die Aufgaben befinden sich auf Übungsserien. Diese bestehen aus **Vorbereitungsaufgaben (ÜVb)**, die im Selbststudium bearbeitet werden, und **Präsenzaufgaben (ÜPr)**.

Daraus ergeben sich die Bezeichnungen **V**, **VNb**, **ÜVb** sowie **ÜPr** für die Ereignisse des Moduls. Aufgaben in der Prüfung (hier: Klausur) werden schließlich anhand der Lernziele aus den EPL formuliert.

Vorgehensweise/Einsatzszenario

Verfahrensschritte

Die EPL müssen von der Lehrperson erstellt und anschließend den Studierenden zur Verfügung gestellt werden. Im Idealfall findet dies alles im Rahmen der Planung der Lehrveranstaltung statt. Realistischer ist allerdings, zumindest bei der erstmaligen Erstellung der EPL, die Lernziele und zugehörigen Ereignisse während der Durchführung eines Moduls schrittweise zu erfassen. So kann man diese auch für bereits lange bestehende Module einführen.

Die Erstellung der EPL kann in vier Schritten erfolgen:

1. Ereignisse definieren

Ein Ereignis steht im Dienst der Erreichung eines oder mehrerer Lernziele. Es sollte konkret genug sein, dass dessen Verknüpfung mit einem Lernziel die effiziente Prüfungsvorbereitung unterstützt. Je nachdem, ob sich eine Modulprüfung nur über einen Veranstaltungstyp erstreckt, z.B. eine Vorlesung, oder ob mehrere Typen wie Vorlesung, Seminar oder Praktikum inbegriffen sind, können zum Beispiel einzelne Termine (z.B. V 15) oder konkrete Aufgaben (z.B. Ü 10.1) daraus als Ereignis dienen. Ebenso können Kapitel eines Skripts adressiert werden. Sind konkrete Aufgaben für das Selbststudium (z.B. VNb 15.1) vorhanden, können auch diese einbezogen werden.

2. Lernziele erfassen

Die ermittelten Ereignisse werden nun hinsichtlich ihrer Lernziele analysiert. Dabei werden für jedes Ereignis die dort adressierten Lernziele erfasst. Um danach die prüfungsrelevanten Lernziele zu extrahieren, ist generell auf eine operationalisierte Formulierung zu achten, d.h. ein Lernziel muss so formuliert sein, dass messbar ist, ob es erreicht wurde (Ulrich, 2020, S. 47).

3. Prüfungsrelevante Lernziele extrahieren und Ereignisse zuordnen

Aus den vorhandenen Lernzielen werden nun die prüfungsrelevanten extrahiert. Für jedes dieser **prüfungsrelevanten Lernziele** werden alle **zugehörigen Ereignisse** zusammengefasst. Im Rahmen dieses Schrittes kann auch ermittelt werden, ob Ereignisse vorliegen, z.B. bestimmte Übungsaufgaben, die kein prüfungsrelevantes Lernziel adressieren. So kann entschieden werden, ob dieses Ereignis eventuell entfernt wird. Auch Lernziele, die nicht ausreichend mit Ereignissen untersetzt sind, können so identifiziert werden.

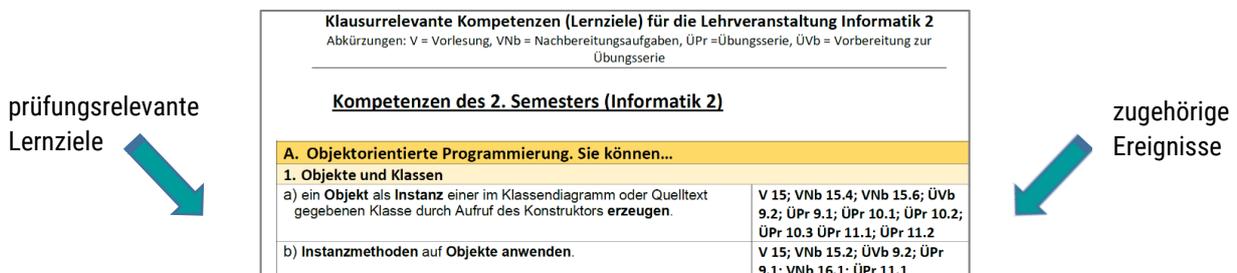


Abbildung 1: Auszug aus der tabellarischen Darstellung der EPL für das Modul Informatik 2 (Medizintechnik),
Quelle: (Wieczorek und Ulrich 2022)

4. Prüfungsaufgaben anhand der EPL entwerfen

Aus operationalisiert formulierten Lernzielen können unmittelbar Prüfungsaufgaben abgeleitet werden.

Beispiel am Lernziel A.1.a

Das Lernziel A.1.a) lautet „Sie können ein Objekt als Instanz einer im Klassendiagramm oder Quelltext gegebenen Klasse durch Aufruf des Konstruktors erzeugen“. Im nachfolgenden Beispiel wird dieses Lernziel in verschiedenen Ereignissen erarbeitet.

Die Einführung der Inhalte erfolgt in **Vorlesung V 15**, eine erste Sicherung des Lernergebnisses erfolgt während V 15 mit einer sich anschließenden interaktiven Aufgabe mit Klickern (s. Abbildung 2).

3. Erzeugung von Objekten: Aufruf der Konstruktormethode Mitschrift

- Syntax in Python allgemein
`objektname = Klassenname (Parameter)`
- Im Beispiel:

benötigte Parameter

`myKonto = Konto (1, "Max Muster")`

beliebige Variable, die Konto-Objekt bezeichnet Ergebnis des Aufrufes der Konstruktormethode: Konto-Objekt

| | | | | | | | |
|--|--|--------------------|-----------|------------|-----------------|----------------------------|-------------------|
| <p>Vorlesungsaufgabe</p> <p>Wählen Sie nun anhand dieses Klassendiagramms den korrekten Befehl zur Erzeugung eines Kaffeetasse-Objektes aus.</p> <p>A. <code>myCup = Kaffeetasse (200, 60)</code> B. <code>myCup = Kaffeetasse ("200 ml", "60 C")</code> C. <code>myCup = Kaffeetasse ()</code></p> | <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">Kaffeetasse</td> </tr> <tr> <td style="padding: 2px;">fuelstand</td> </tr> <tr> <td style="padding: 2px;">temperatur</td> </tr> <tr> <td style="padding: 2px;">Kaffeetasse ()</td> </tr> <tr> <td style="padding: 2px;">fuelleAuf(wieviel,wieWarm)</td> </tr> <tr> <td style="padding: 2px;">kuehleAb(wieviel)</td> </tr> </table> | Kaffeetasse | fuelstand | temperatur | Kaffeetasse () | fuelleAuf(wieviel,wieWarm) | kuehleAb(wieviel) |
| Kaffeetasse | | | | | | | |
| fuelstand | | | | | | | |
| temperatur | | | | | | | |
| Kaffeetasse () | | | | | | | |
| fuelleAuf(wieviel,wieWarm) | | | | | | | |
| kuehleAb(wieviel) | | | | | | | |

Abbildung 2: Erarbeitung von Lernziel A.1.a) in Vorlesung V 15, Quelle: (Wieczorek und Ulrich 2022)

In der **Nachbereitung** von Vorlesung 15 werden interaktive Moodle-Tests verwendet (s. Abbildung 3), hier ist der korrekte Befehl aus einer Reihe von vorgegebenen auszuwählen.

Gegeben ist wieder die Klasse `Verordnung` (Quelltext zum Download und selbst Testen [hier](#)):

```
class Verordnung:
    def __init__(self, einPatientenname, einMedikament, eineDosis):
        self.patientenname=einPatientenname
        self.medikament=einMedikament
        self.dosis=eineDosis

    def getPatient(self):
        return self.patientenname

    def getMedikament(self):
        return self.medikament

    def getDosis(self):
        return self.dosis

    def aendereDosis(self, neueDosis):
        self.dosis=neueDosis

    def aendereMedikament(self, neuesMedikament):
        self.medikament=neuesMedikament
```

Geben Sie an, welche Befehle zur Erzeugung eines Objektes vom Typ `Verordnung` korrekt sind.

Wählen Sie eine oder mehrere Antworten:

- `v=Verordnung("Lisa",'Aspirin',500)`
- `dieVerordnung=Verordnung("Horst",'Ibuprophen',400)`
- `dieVerordnung=Verordnung("Horst','Ibuprophen')`
- `v=Verordnung()`

Abbildung 3: Lernziel A.1.a) in Nachbereitungsaufgabe VNb 15.6,
Quelle: (Wieczorek und Ulrich 2022)

In **Übungsserie 9** (s. Abbildung 4) ist der entsprechende Programmier-Befehl selbst zu erstellen.

Aufgabe 1. Konto-Objekte definieren und Methoden anwenden

- a) Falls Sie die Aufgaben nicht im Online-Classroom lösen: Speichern Sie die Datei [kontoklasse.py](#) in Ihrem Verzeichnis. (Diese ist **ähnlich** dem Beispiel aus der Vorlesung, aber etwas **ergänzt**). Erstellen Sie dann eine **neue** Datei **main.py**. Importieren Sie dort das Modul `kontoklasse`.
- b) Erstellen Sie in `main.py` ein Konto-Objekt, welches Sie mit `meinKonto` bezeichnen.**
- c) Zahlen Sie den Betrag 100 auf das Objekt `meinKonto` ein.
- d) Lassen Sie sich den Kontostand anzeigen.
- e) Erzeugen Sie ein zweites Konto `deinKonto`. Überweisen Sie den Betrag 20 von `meinKonto` auf `deinKonto`. (Achten Sie hier darauf, für welches Konto Sie dann den Pin eingeben müssen).

Abbildung 4: Lernziel A.1.a) in ÜPr 9.1,
Quelle: (Wieczorek und Ulrich 2022)

In Abbildung 5 ist eine entsprechende **Prüfungsaufgabe** dargestellt.

```

class Reagenzglas:
    __anzahl = 0

    def __init__(self):
        Reagenzglas.__anzahl += 1
        self.__fuellstand = 0

    def getAnzahl():
        return Reagenzglas.__anzahl

    def fuelleAuf(self, wieviel):
        self.__fuellstand = self.__fuellstand + wieviel

    def schuetteAus(self, wieviel):
        if wieviel <= self.__fuellstand:
            self.__fuellstand = self.__fuellstand - wieviel
        else:
            print("So viel können Sie nicht ausschütten.")

    def getFuellstand(self):
        return self.__fuellstand
    
```

Geben Sie für die Aufgaben a) bis d) jeweils den Python-Quelltext an.

a) Erzeugen Sie ein Objekt `x` vom Typ `Reagenzglas`.

Abbildung 5: Lernziel A.1.a) als Prüfungsaufgabe,
Quelle: (Wieczorek und Ulrich 2022)

Durch diese und die weiteren in den EPL genannten Ereignisse kann das Lernziel erreicht werden und die Messung ist möglich.

Fragebogen zur Selbsteinschätzung für Studierende

Um die Studierenden bei der Prüfungsvorbereitung zu unterstützen, liegen die EPL auch in einem Fragebogen-Format im Moodle-Kurs vor. Die Studierenden können hiermit bereits ab Beginn des Moduls ihre Selbsteinschätzung zu den einzelnen Lernzielen erfassen und speichern. Der Fragebogen ist nur für die Studierenden selbst gedacht und soll nicht abgegeben werden.

Selbsteinschätzung zu klausurrelevanten Kompetenzen [Druckversion \(ohne Antworten\)](#)

1 Objekte und Klassen: Sie können ... (1 = sehr gut, 5 = gar nicht)

| | 1 | 2 | 3 | 4 | 5 |
|---|----------------------------------|----------------------------------|-----------------------|----------------------------------|-----------------------|
| a) Ein Objekt als Instanz einer im Klassendiagramm oder Quelltext gegebenen Klasse durch Aufruf des Konstruktors erzeugen (V 15; VNb 15.4; VNb 15.6; ÜVb 9.2; ÜPr 9.1; ÜPr 10.1; ÜPr 10.2; ÜPr 10.3; Ü11.1; ÜPr 11.2) | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| b) Instanzmethoden auf Objekte anwenden. (V 15; VNb 15.2; ÜVb 9.2; ÜPr 9.1; VNb 16.1; ÜPr 11.1) | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| c) Klassenmethoden korrekt aufrufen. (V 16; VNb 16.1; VNb 16.3; ÜVb 11.1; ÜPr 11.1) | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| d) Attribute, Methoden und Parameter von Methoden anhand einer Klassendefinition in Python benennen. (V 15; ÜPr 9.2; ÜPr 9.3; ÜPr 11.1) | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/> |
| e) Auswirkungen von Methodenaufrufen anhand der Klassendefinition beschreiben (unter korrekter Verwendung der Begriffe Attribut, Parameter, Wert, Rückgabe, Objekt, lesender und schreibender Zugriff). (V 15; VNb 15.7; ÜVb 9.1; ÜVb 9.2; ÜPr 9.2; ÜPr 11.1) | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

Abbildung 6: Selbsteinschätzung für Studierende als Moodle-Befragung,
Quelle: (Wieczorek und Ulrich 2022)

Didaktische Herausforderungen

- Analyse der Ereignisse eines Moduls und Erfassung von Lernzielen, die durch die Ereignisse adressiert werden
- Operationalisierte Formulierung der Lernziele
- Genügend konkrete Formulierung der Ereignisse

Medien

- Lernmanagementsystem Moodle zur Bereitstellung von EPL in zwei Formaten:
 - als pdf-Dokument (als Übersicht zum Ausdrucken, vgl. Abbildung 1)
 - Fragebogen (Moodle-Aktivität „Befragung“ – zur Speicherung des aktuellen Lernstands für die Studierenden zur Selbsteinschätzung vgl. Abbildung 6)
- Klicker zur Aktivierung während des Ereignisses „Vorlesung“

Wie stark werden folgende Kompetenzen auf einer Skala von 1-10 gefördert?

Die Methode dient dazu, die in der jeweiligen Lehrveranstaltung angestrebten Kompetenzen zu fördern, indem diese explizit genannt und mit entsprechenden Ereignissen verknüpft werden. Prinzipiell können daher beliebige Kompetenzen gefördert werden, wenn diese in einer Veranstaltung mit entsprechenden Ereignissen untersetzt sind.

Mit welchem Ziel wurde dieses Konzept erstellt?

Die EPL zielen darauf ab, einerseits den Studierenden eine exakte Übersicht der notwendigen, zu erreichenden Lernziele für die Leistungsprüfung zu geben.

- Frühzeitige größtmögliche Transparenz hinsichtlich der Prüfungsanforderungen
- Ermöglichung gezielter Vorbereitung der Prüfung durch Studierende bereits ab Beginn des Moduls (durch Lernziele, wo zugehörige Ereignisse am Anfang stehen), dadurch Reduzierung des Aufwands unmittelbar vor der Prüfung
- Für Studierende Erfassung der Selbsteinschätzung (in übersichtlichem Fragebogen-Format) zur Erhöhung der Effizienz der Vorbereitung. Lücken können gezielt geschlossen werden.
- Daneben bieten sie den Lehrenden
 - ein Werkzeug zur effizienten Erstellung von Prüfungen
 - eine Planungshilfe, weil sichtbar ist,
 - ob die aufgeführten Lernziele genügend Ereignisse besitzen
 - ob die aufgeführten Lernziele tatsächlich den Kompetenzen entsprechen, die im Modul erreicht werden sollen.

Arbeitsaufwand für Lehrende

Vor der Veranstaltung



Während der Veranstaltung



Nach der Veranstaltung



Bewertung/Prüfungsleistung: Wie wurde der Leistungsnachweis erbracht?

- Klausur mit Prüfungsaufgaben, die anhand der EPL entworfen wurden (siehe Abbildung 5 im Punkt ‚Vorgehensweise/Einsatzszenario‘)

Erste Evaluationsergebnisse der Methode

Die Auswertung von zwei Studierendenbefragungen im Sommersemester 2020 und 2021 ergab, dass die EPL ähnlich intensiv genutzt werden wie die Übungsserien und Nachbereitungsaufgaben, mehr genutzt werden als Vorlesungsvideos, Fachbücher und Internet-Tutorials, und weniger als Beispielklausuren und eigene Mitschriften. Dies deutet darauf hin, dass die EPL eine sinnvolle Ergänzung der bisherigen Lernmedien zur Prüfungsvorbereitung darstellen.

Literatur

Ulrich, Immanuel. Gute Lehre in der Hochschule. Praxistipps zur Planung und Gestaltung von Lehrveranstaltungen. 2. Auflage. Wiesbaden: Springer, 2020.

[Wieczorek, Barbara, und Immanuel Ulrich. „Ereignisorientierte prüfungsrelevante Lernziele als Werkzeug im Sinne des Constructive Alignment: Konzeption, Umsetzung und Evaluation.“ die hochschullehre, 2022: 228-242.](#)

Dieses Werk ist lizenziert unter [Namensnennung - keine Bearbeitung 4.0](#) Ausgenommen von der Lizenz sind Logos und anders gekennzeichnete Inhalte.
Best Practice Beispiel „Ereignisorientierte prüfungsrelevante Lernziele (EPL) im Modul Informatik“; Prof. Dr. Barbara Wieczorek sowie Karolin Freund und Sandra Dietzel (für Team INSPIRE); Urheberrecht bei Ernst-Abbe-Hochschule Jena; Prof. Dr. Barbara Wieczorek; erstellt am: 12.06.2023



[CC BY-ND 4.0](#)

