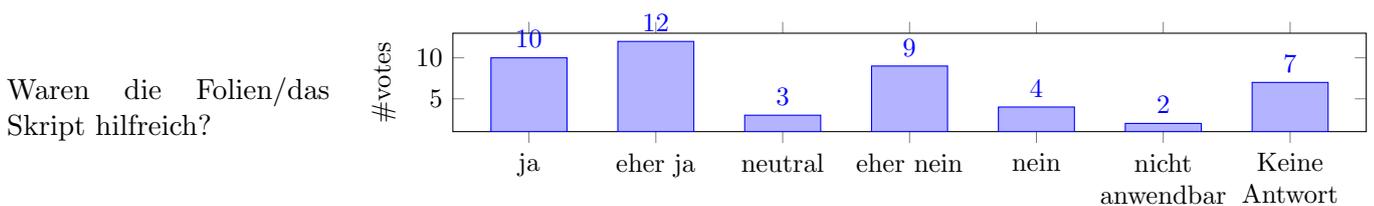
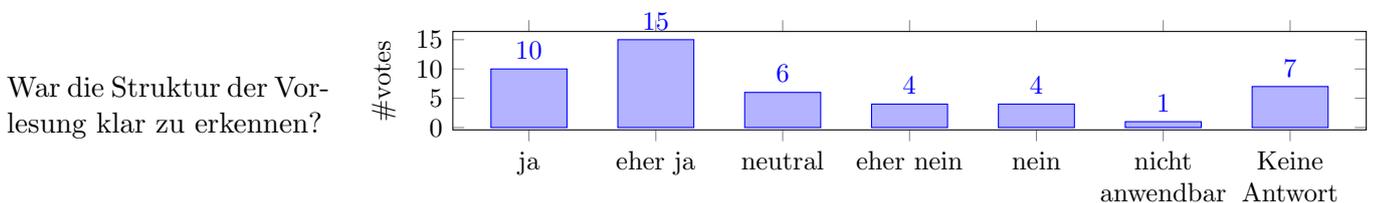
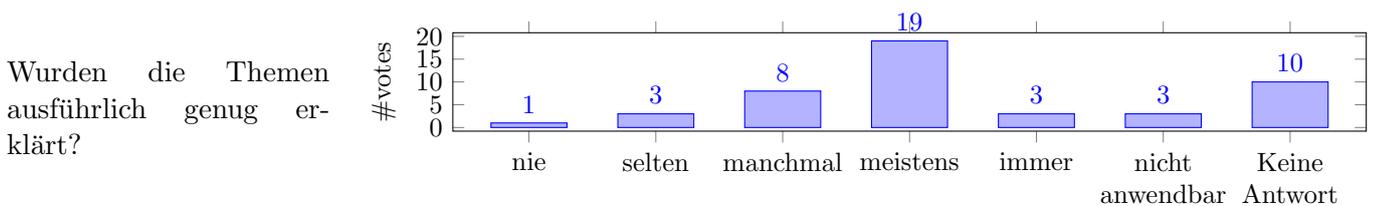
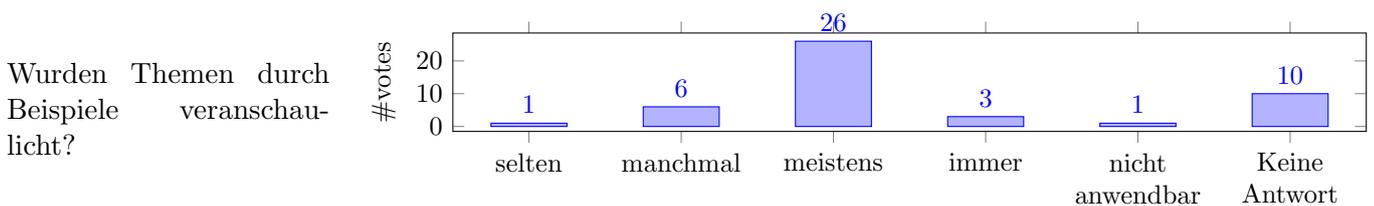
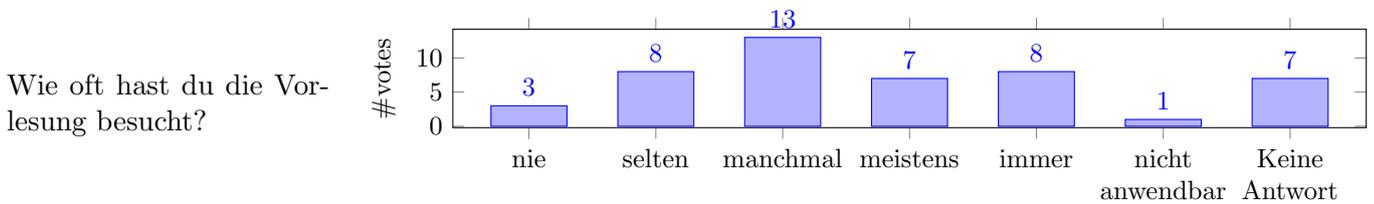
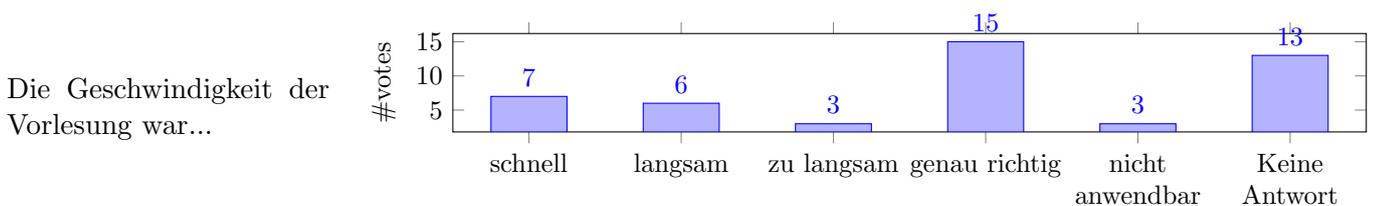


Ergebnis der Online-VLU. Die Umfrage fand in den letzten beiden Vorlesungswochen statt.

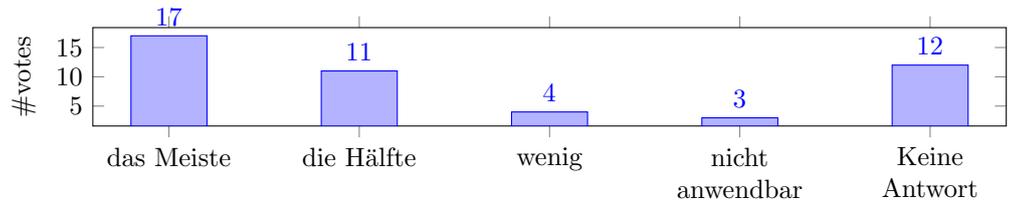
1 Bewertung der Vorlesung



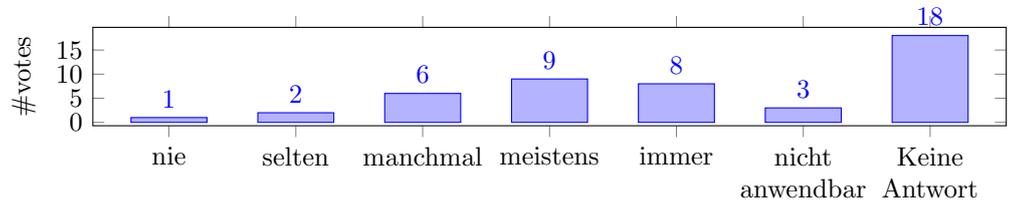
2 Bewertung der Dozierenden



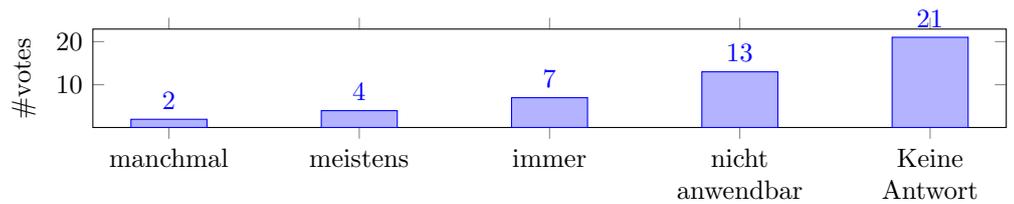
Wie viel verstehst du während der Vorlesung?



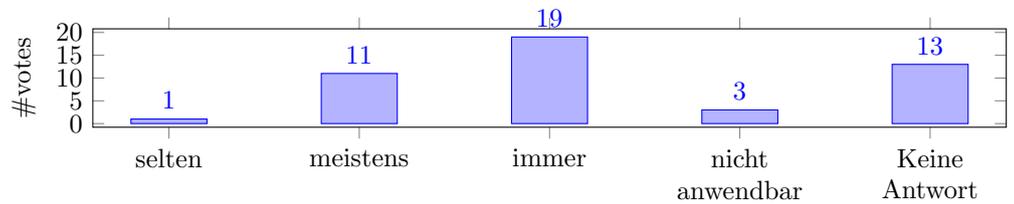
Ist der Dozent/die Dozentin gut auf Fragen eingegangen?



War der Dozent/die Dozentin außerhalb der Vorlesung für Fragen etc. erreichbar?

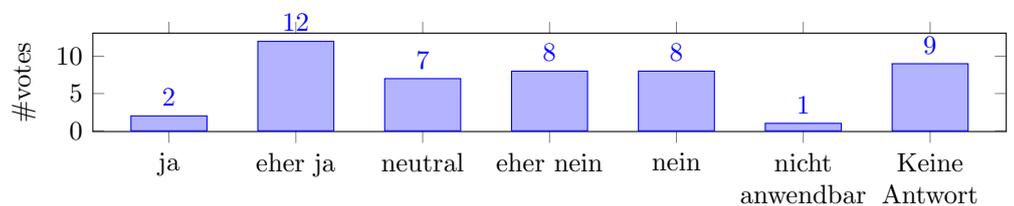


War die Dozentin / der Dozent akustisch gut zu verstehen?

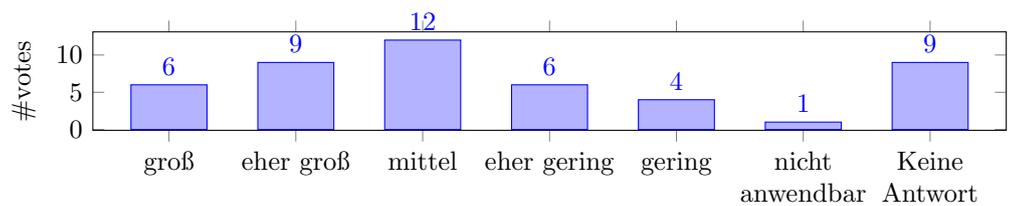


3 Bewertung des Moduls

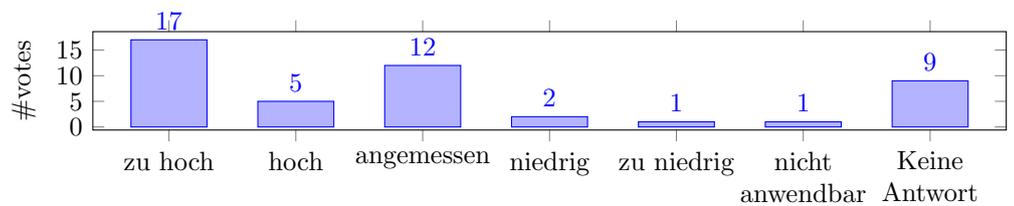
Findest du die verlangten Studienleistungen für dieses Modul angemessen?



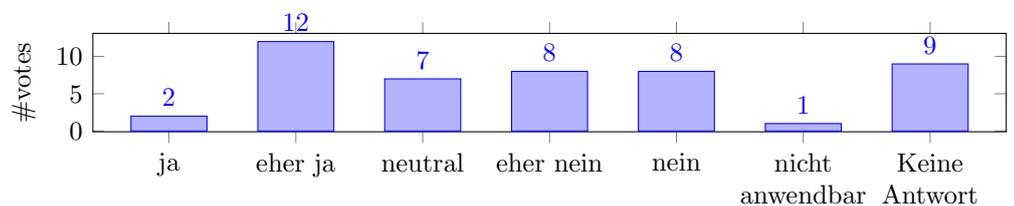
Der Praxisbezug war...



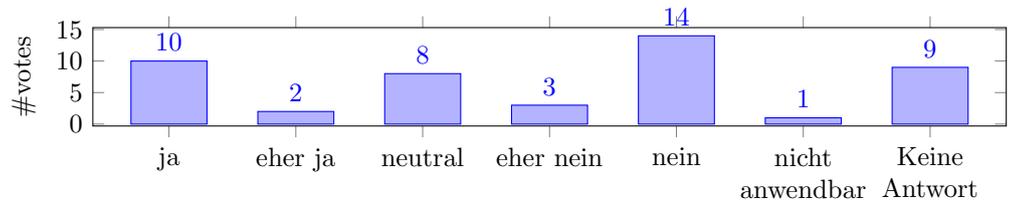
Ist der Arbeitsaufwand für dieses Modul im Hinblick auf die LP-Zahl angemessen?



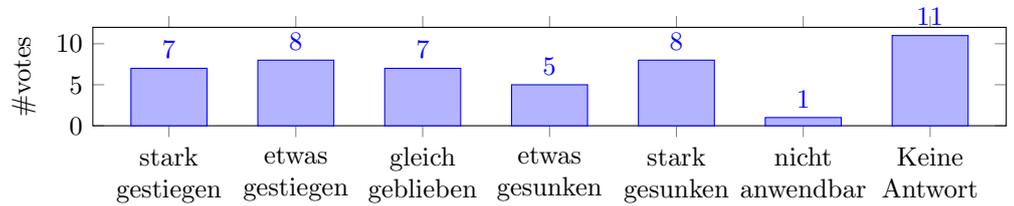
Findest du die verlangten Studienleistungen für dieses Modul angemessen?



Würdest du dieses Modul weiterempfehlen?

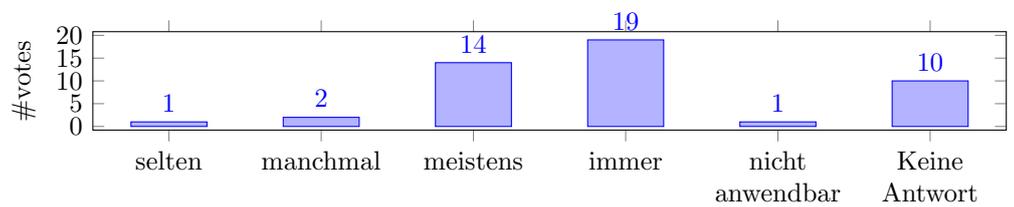


Dein Interesse für dieses Thema ist...

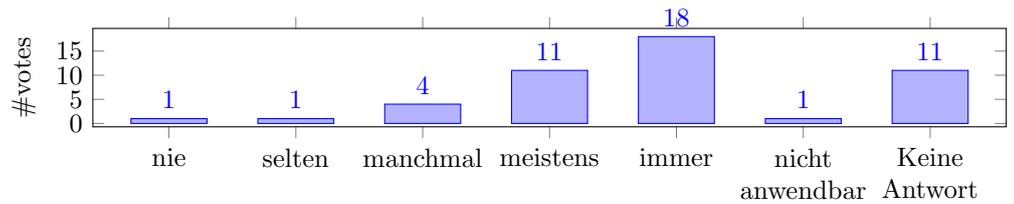


4 Bewertung der Übungsaufgaben

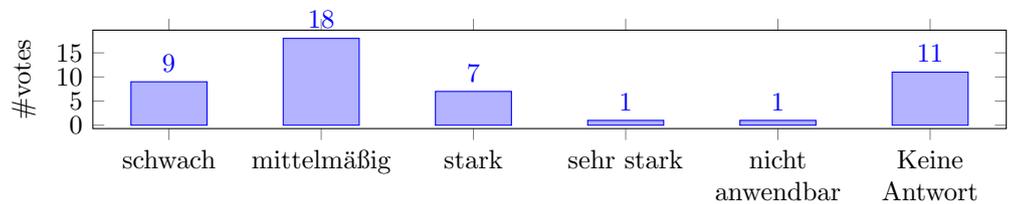
Wie oft hast du die Übungen besucht?



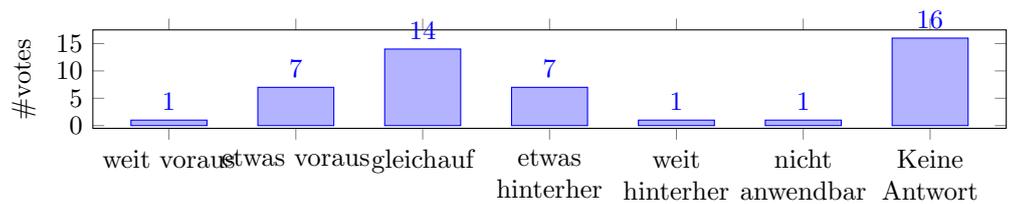
Wurden die Übungsaufgaben rechtzeitig zur Verfügung gestellt?



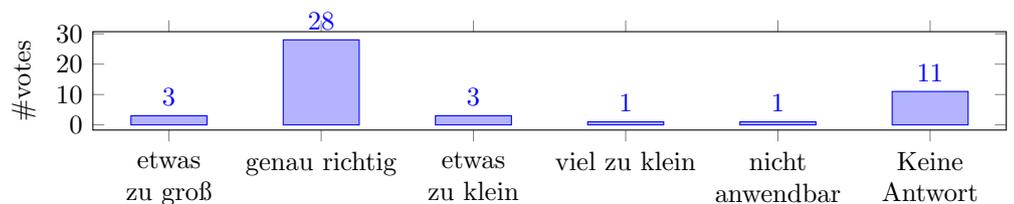
Die Schwierigkeit der Übungsblätter schwankte...



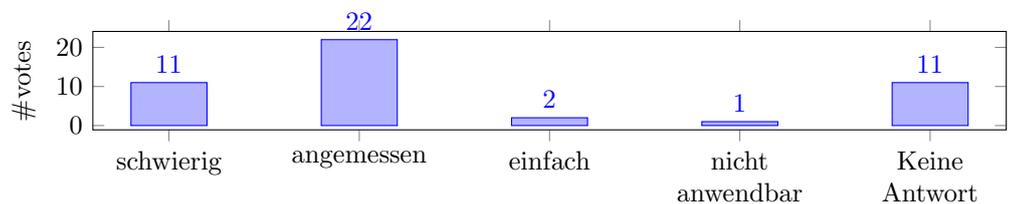
Die Vorlesung war...



Die Übungsgruppe war...

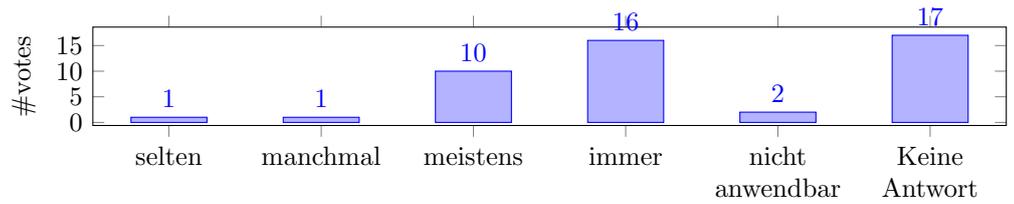


Die Übungsaufgaben waren meistens...

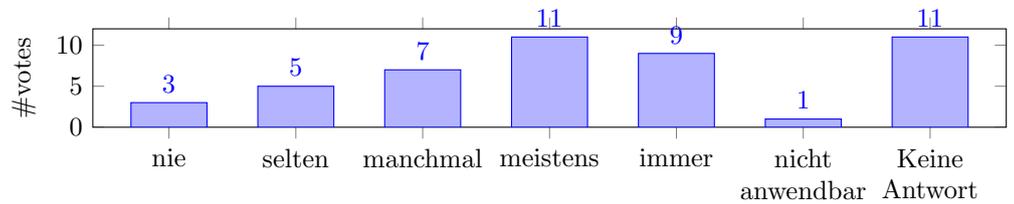


5 Bewertung des Tutoriums

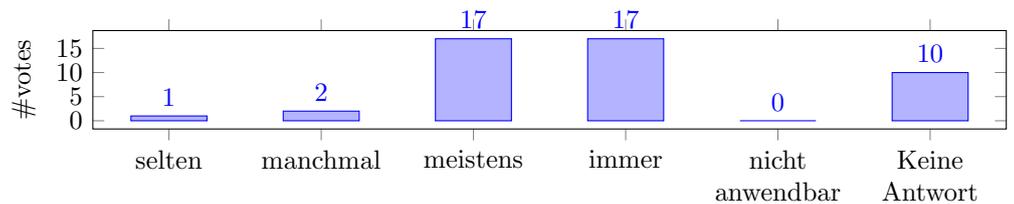
War der Tutor/die Tutorin außerhalb der Übung für Fragen etc. erreichbar?



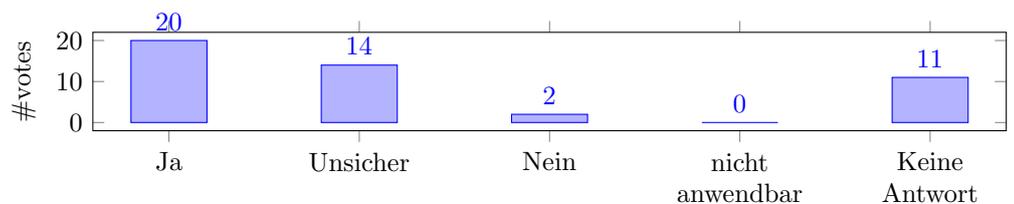
Waren die Korrekturen des Tutors/der Tutorin nachvollziehbar?



Wurde der Tutor/die Tutorin mit dem Stoff der Übung fertig?

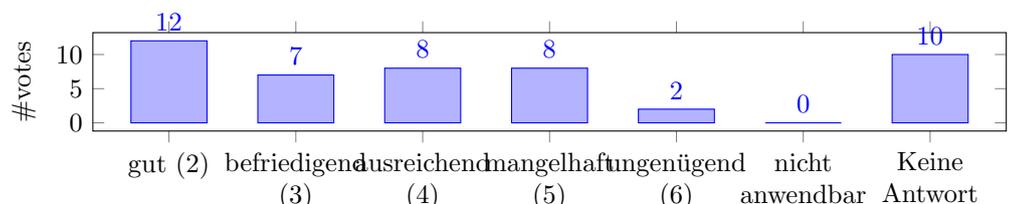


Lohnt sich der Besuch des Tutoriums?

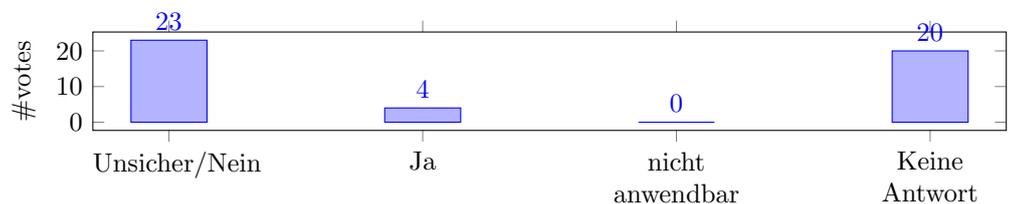


6 Abschließende Bewertung des Moduls

Note:



Hältst du die Vorlesung der Dozent:in für lehrpreiswürdig?



6.1 Wieso?

Ehlicherwise sehe ich nicht, dass ich die Inhalte der Vorlesung in der vermittelten Form verwenden werde. So viele der Patterns/Strukturen kamen mir ziemlich unkonkret vor - besonders, da der gegebene Anwendungsbezug eigentlich gar keine andere Möglichkeit offenlässt. Z.B. die Supermarktkasse oder der Blu-Ray-Player ließen sich meines Erachtens nach einfach auf "Schreibe ein Stück Code pro Komponente" reduzieren. Ob das jetzt technically ein Boundary, ein Decorator oder eine Proxy ist, kommt mir noch immer egal vor. Ist einfach nicht so wirklich für mich, I guess.
Suboptimale Struktur der Vorlesung - Motivation und realer Ablauf der gelernten Inhalte wurde erst in den letzten Wochen der Vorlesung angesprochen
Vorlesungen waren ganz ok aber die Klausuren waren unangemessen. Komische Aufgaben und bauten zu stark aufeinander auf
Die Vorlesung bekommt seit langem nicht die nötige Überarbeitung - der Lehrpreis soll aber Innovation in der Lehre belohnen. Eine Überarbeitung mindestens der Folien (z. B. hinsichtlich Konsistenz, Einhaltung der selbst vorgestellten Standards und Beinhaltung aller später geforderten Inhalte) sowie die Vermeidung der diversen organisatorischen Schwierigkeiten wären wünschenswert, bevor über eine mögliche Qualifizierung für den Lehrpreis nachgedacht wird.
Es gab nichts, was eines Preises würdig gewesen wäre.

7 Freitextkommentare

7.1 Was hat dir an dieser Lehrveranstaltung gefallen?

Fande die besprochen Themen allgemein sehr interessant. Der Dozent hat die Inhalte sehr schön und verständlich übergebracht
behandelte Entwicklungsmuster sind hilfreich und hätten z.B. im Praktikum OOSE sinnvolle Anwendung finden können
Zwar etwas trockene aber sehr wichtige Inhalte.
Der praxisbezug und das vertiefen von UML
Tutorien und Übungsaufgaben
Einblicke in Java und JS
Wovon sich alle anderen Lehrenden mal eine Scheibe abschneiden können: Der Dozent hält die Vorlesung noch immer in Hybrid, nimmt sie auf und stellt die Videos zur Verfügung. Das ist wirklich enorm hilfreich zum lernen und schade, dass das nicht mehr machen. Außerdem fand ich die Vorlesung angenehm zum Zuhören, der Dozent war sehr motiviert und kompetent.
Da ich als Praktikant in einem Softwareentwicklungsunternehmen gearbeitet habe, finde ich den Inhalt der Vorlesungen sehr wichtig und arbeitsrelevant und jeder anständige Entwickler sollte ein gutes Verständnis für den Inhalt haben
Das ich trotz der komischen Klausur eine ganz gute Note geschrieben habe
Einige Inhalte (z. B. Entwurfsmuster, UML) sind wichtige Themen, die man sonst im Studium nicht explizit kennen lernt.
Der Dozent schien zumindest selbst von seinen Themen begeistert zu sein.

7.2 Was könnte noch besser gemacht werden?

Die Punkteverteilung der Übungsaufgaben und der Klausuraufgaben ist einfach nicht nachvollziehbar und für ein so praktisches Modul mit vielen Modellierungen und alternativ Lösungen nicht passend.
Die Struktur wirkte manchmal ein wenig durcheinander da mal ein große Menge Patterns kam dan lange keine Patterns mehr und zum Ende wurden dann nochmal welche eingeschoben
Keine Punkte in der Klausur aufgrund einer Mail verschenken.
Klarere Beschreibung der zu lernenden Inhalte auf einem abstrakten allgemeinen Level. Ich musste mir die Regeln oft anhand von Beispielen herleiten
JavaScript entweder mehr in den Fokus der Vorlesung oder weniger in den Fokus der Klausur rücken
Zu viele Inhalte für 6 LP In der Erstklausur wurde zu 90% nur Stoff der ersten beiden Vorlesungen abgefragt...
Weniger javascript
Aktualität des Inhalts, Klausuraufgaben und ein richtiges Skript
Übungsaufgaben sind schwierig in der Gruppe aufteilbar, da Aufgabenumfang oft sehr schwankt und zum Teil Aufgaben aufeinander aufbauen
Aufgaben relevanter für die Klausur am Ende
<ul style="list-style-type: none">- Das Übungsmodell: Ich finde gut, dass man Übungen nur in Kleingruppen hat und da dann viel besser betreut werden kann. Das Zulassungsmodell habe ich aber als extrem undurchsichtig empfunden, weil man gar nicht nachvollziehen kann, wie viele Punkte man hat. Außerdem kriegt man keine Korrektur und weiß dementsprechend auch nicht, was man im Detail falsch gemacht hat (es gibt zwar Musterlösungen, was ich klasse finde, aber gerade bei den Diagrammen gibt es doch unzählige Lösungsmöglichkeiten...)- Es wurde häufig überzogen. Außerdem fand ich es gar nicht in Ordnung, die Vorlesung erst abzusagen, während sie schon zu 2/3 vorbei ist und alle vergeblich auf den Dozenten warten und dann auch noch ein Video für den Tag hochzuladen, was auch noch 15 Minuten zu lang war, obwohl alle Studierenden schon im Hörsaal unnötig gewartet hatten.- Ich fand schade, dass es in dem Modul thematisch um Dinge geht, die in der Praxis der Softwareentwicklung eigentlich total wichtig sind und man trotzdem so wenig für die Praxis mitnehmen kann. Es wurde ein viel deutlicherer Fokus auf Formalismus als auf eigentliche Umsetzung gelegt (zB bei den Design Pattern: Ich kann jetzt zu jedem ein UML Diagramm zeichnen, aber würde die in der Praxis wohl weder erkennen noch implementieren können). Außerdem fand ich es anstrengend, dass in den Übungen nur Java als klassenbasierte OO-Sprache vorkam. Ich hab Java nie gelernt und konnte viele Eigenarten gar nicht wissen. Das hat viel Eigenrecherche verlangt (was ja an sich okay ist, aber nicht für ein Modul, was nur so wenig LP gibt die schon völlig überreizt werden)
Die Übungsaufgaben könnte wie ein Projekt gestaltet werden, bei dessen Entwicklung wir Software-Engineering-Konzepte anwenden. Eine solche Arbeit könnte uns helfen, uns nach dem Abschluss besser mit den Arbeitsumgebungen für Softwareentwickler in unserem Beruf vertraut zu machen
Keine Sexistischen oder Transfeindliche Kommentare in der Vorlesung vom Dozenten
<ul style="list-style-type: none">- Aktualisierung der Musterlösungen- bessere Organisation von Übungen, Klausur, Einsicht- sorgfältigere Klausurkorrektur- konsistentere und nachvollziehbare Bepunktung, sodass die Noten nicht so willkürlich wirken und in der Einsicht so viel rausgeholt werden kann- vernünftige Versionierung der Folien (nicht einfach nur den Namen mit _final, erweitern)

Die Übungsaufgaben und die Klausur fühlen sich an vielen Stellen leider so an, als bestünde die Hauptaufgabe darin, die Auslegung der Aufgabenstellung durch den Dozenten/die Tutorierenden (die über eine Musterlösung viertelpunktegenau ein Bewertungsraster vorgeschrieben bekommen) zu reverse-engineerieren. Während in der Vorlesung oftmals betont wird, dass viele verschiedene Entwürfe für eine Problemstellung "richtig" sein können, erfolgt die Bewertung in Tutorien und Klausur nach erstaunlich subjektiven Kriterien (nicht "kann der Entwurf Ziel xyz umsetzen", sondern ist xyz im Entwurf gleich wie in der Musterlösung").

Die Vorlesung ist an sich nicht schlecht gehalten, nur etwas zu langsam, bzw. es wird öfters ein Fokus auf die falschen Punkte gelegt. Das ein oder andere Beispiel ist wirklich überflüssig, während an anderer Stelle eine Notation (die später sehr penibel eingehalten werden muss) nicht nachvollziehbar oder (außer in der Musterlösung von Übungen) gar nicht eingeführt wird. Einige wesentliche Konzepte werden leider nicht klar, z. B. im Rahmen von Entwurfsmustern, da keine zielgruppenorientierten Beispiele gegeben werden (erwähnte Namen von Software sind z. B. so alt, dass niemand sie kennt).

Gleichzeitig sind einige Beispiele etwas fragwürdig, z. B. frage ich mich, ob wirklich eine Debatte zu biologischem und selbstbestimmtem Geschlecht aufgemacht werden muss, weil unbedingt dieser Kontext zur Erklärung gewählt werden muss.

Die Folien sind an der ein oder anderen Stelle hilfreich und es lässt sich auch erkennen, dass dort in den letzten Jahren weiter dran gearbeitet wurde. Allerdings lässt sich auf gefühlt fast jeder Folie eine Inkonsistenz finden. Die eigene Notation wird nicht klar eingehalten, deren Einhaltung aber von Studierenden penibel gefordert. Manche Inhalte finden sich einfach nicht auf den Folien und in der Vorlesung. Auch der Umgang mit gegendertem Sprache ist etwas komisch: An manchen Stellen wird gegendert, an dem meisten jedoch nicht. Nur die SSekretärin ist auf einmal weiblich, während alle anderen Bezeichnungen männlich sind usw.

Jede Kleinigkeit für sich genommen ist nicht so schlimm, insgesamt halten die Probleme in den Vorlesungsmaterialien aber vom eigentlichen Inhalt ab, und man fragt sich: Warum? Warum wurde das nie alles einmal überarbeitet?

Inhaltlich könnte sich auch ein Abgleich mit anderen Modulen des Bachelors lohnen. Gerade im ersten Teil der Vorlesung wurden unglaublich langsam die Grundlagen einer objektorientierten Programmiersprache wiederholt, die eigentlich bereits halbwegs bekannt sein sollten.

Organisatorisch sind über das Semester leider einige Schwierigkeiten aufgetreten. Seien es die erst spät klar gewordenen Zulassungskriterien, eine ohne rechtzeitige Ankündigung ausgefallene Vorlesung, ab und an nicht funktionierende Aufzeichnungen, die versehentliche Veröffentlichung der eigenen Passwort txt Datei für Tutorierende, die versehentliche Veröffentlichung der Musterlösungen einer Übungsaufgabe (und dann lange Unklarheit, wie jetzt damit umgegangen wird), die unklare "Versionierung" der Vorlesungsfolien (Bsp.: "02-oom-d V3 (update nach final).pdf" und "04-entwurfsmuster (Stand 22.11.2023 -- Ende Proxy).pdf"), nicht funktionierender Upload der Folien, Übungszettel oder VL-Videos über den Sciebo-Client, und zuletzt die mittlerweile nicht mehr abrufbaren Vorlesungsmaterialien.

Jede Vorlesungswoche fühlt sich leider ein bisschen wie ein Würfelwurf an, was jetzt schon wieder nicht klappt. Jede Schwierigkeit für sich ist auch hier erst mal nicht schlimm und kann passieren, in Summe ist es aber einfach unglaublich nervig.

Viele Informationen, sowohl inhaltlich als auch organisatorisch, wurden sehr schlecht kommuniziert. In der Zukunft würde es sich bspw. anbieten, den Ausfall einer 09:00-Uhr-Vorlesung frühzeitig mitzuteilen und nicht erst 30 Minuten nach Beginn. Außerdem sollten die Folien - insbesondere in einem Modul, in dem penibel auf korrekte Notation geachtet wird - klar strukturiert sein und keinen Interpretationsspielraum bezüglich dieser Notation lassen.

Mehr Praxisnähe statt Theorie

Weniger Java-spezifisches/ Umstieg auf C++

In diesem Modul wird viel mit Java gearbeitet. Da dies nicht mehr Inhalt der Vorlesung AIPro ist, wäre es hier gut, wenn diese Programmiersprache nicht von Anfang an vorausgesetzt werden würde. Vielleicht wäre hier ein Crash-Kurs angebracht für diejenigen, die vorher noch keinen Kontakt zu Java hatten.

Es wäre wünschenswert, wenn der Dozent sein Nicht-Erscheinen zur Vorlesung nicht erst 30 Minuten nach Beginn der Vorlesung mitteilen und einen mit einem Video als Ersatz abspeisen würde.

Meiner Meinung nach sollte außerdem unbedingt die Konzipierung der Klausur überarbeitet werden. Ich finde es sehr schwierig, wenn 5 von 6 Aufgaben aufeinander aufbauen und man auf Ergebnisse aus vorigen Aufgaben zurückgreifen muss. Ich halte dies für besonders kritisch, insbesondere wenn Aufgaben zum Teil voneinander abweichende Informationen zum gleichen Aspekt enthielten. Es wird somit im Übrigen auch denjenigen die Klausur erschwert, die vielleicht bereits bei Aufgabe 1 Verständnisprobleme hatten und daher Aufgabe 2-5 nur schwerlich bearbeiten konnten.

7.3 Hier hast du Platz für weitere Anmerkungen und Feedback zum Modul.

Ich habe noch nie ein so langweiliges Modul gehört. Die Folien sind nicht sonderlich gut. Und ich hatte wirklich kein Spaß, für zukünftige wäre es gut das Modul als nicht Pflicht zu markieren.

bitte den inhaltlichen Umfang des Moduls an die LP-Zahl anpassen und die Vorlesung ggf auf Basis der Umfrageergebnisse umstrukturieren

An sich ist das Modul nicht uninteressant. Jeder Informatiker wird nicht drum rum kommen sich mit der Frage zu beschäftigen wie Software entwickelt, gewartet und verbessert wird. Aber für diese Klausur zu lernen war eine der schlimmsten Uni Erfahrungen, die ich bisher hatte. Die Klausuraufgaben vor allem dieses Jahr waren so komisch und total unproportional dazu, was auf den Blättern drankam. Gefühlt 60 % der Vorlesung beschäftigt sich mit allen UML Diagrammen (Sequenz, Aktivität usw.), nur damit dann am Ende ein Klassendiagramm drankommt (was an sich auch nicht schwer war und viele Punkte gibt) aber dann andere Themen, die zwar auch in der Vorlesung behandelt werden den Rest der Klausur einnehmen. Auf einmal hatte OCL eine große Relevanz, als auch JavaScript. Wenn es nicht so einen riesigen Bereich für 1,0 gäbe würden die Klausuren mit großer Sicherheit nicht so gut ausfallen (mag ja auch die Intention dahinter sein). Es ist einfach ein sehr unbefriedigendes Gefühl, sich perfekt mit allen UML Diagrammen auseinander zusetzen nur damit dann ein Klassendiagramm drankommt, aber ein anderen "Randthema eine komplette Aufgabe ist.

Das Modul sind einfach keine 6 Credits wert. Ich hab es zeitgleich mit zwei 9 LP Modulen gehört die weitaus weniger Arbeit in Anspruch genommen haben. Das finde ich sehr schade.

Ich habe diese Vorlesung nur belegt, weil es ein Pflichtmodul war und würde sie höchstwahrscheinlich nicht als Wahlmodul empfehlen.

Die Übungsaufgaben sind überhaupt nicht schwierig aber trotzdem ermüdet zeitintensiv. Außerdem ist Swt der einzige Kurs in dem es eine faktische Anwesenheitspflicht im Tutorium gibt.

Das Modul verfehlt meiner Meinung nach leider etwas das Lernziel, was schade ist: Es soll die Entwicklung guter Software in einem Team gelehrt werden. Die Anwendung der Inhalte in der Vorlesung auf Fragestellungen in der Praxis bleibt aber leider oft eher unklar. Sinnvoller wäre vermutlich, die Techniken praktisch anwenden zu üben, damit auch etwas hängen bleibt, ähnlich wie es im Modul POOSE umgesetzt ist. Dabei sollten unbedingt sinnvollere Übungsaufgaben gestellt werden, als es momentan der Fall ist.

SWT ist für Informatiker ein Pflichtmodul. Dieser Platz scheint mir momentan unangemessen.

Man hatte bei diesem Modul das Gefühl, dass zwar mit der neuen Prüfungsordnung die LP gekürzt wurden, aber der Stoff dem nicht angepasst wurde. Von der Masse an Stoff und an Aufgaben war es sehr viel. Ohne Aufteilung der Aufgaben in der Abgabegruppe war es meiner Meinung nach nicht zu bewältigen, jede Woche alle Aufgaben zu erledigen. Außerdem waren es zum Lernen am Ende fast 1000 Folien. Diese Lernmasse war schwer zu bewältigen.