

Sommersemester

Vorwort

Auf dieser Wiki-Seite sind die potenziellen Aufgaben eines Tutors für das Sommersemester aufgelistet. Ein Tutor ist nicht verpflichtet, alle diese Aufgaben zu übernehmen, sondern kann eine **Teilmenge** davon übernehmen. Die Tätigkeit als Tutor erstreckt sich in der Regel von Anfang **April** bis Ende **September**, jedoch kann es in Ausnahmefällen zu Abweichungen kommen.

Um als Tutor tätig zu sein, ist die **erfolgreiche Teilnahme** an der Veranstaltung "Funktionale und Objektorientierte Programmierung" oder ähnlichen Kursen, die Java vermitteln, wie z.B. "Einführung in die Programmierung", erforderlich.

Hinweise

Die angegebenen Zeitangaben für die verschiedenen Tätigkeiten dienen lediglich als **Richtwerte** bzw. Durchschnittswerte und können je nach individuellen Gegebenheiten variieren. Zusätzlich zu jeder Tätigkeit wendet ein Tutor 18 Stunden für **organisatorische Tätigkeiten** wie Meetings usw. auf.

Inhaltsverzeichnis

- [Gestaltung von Übungsblätter](#)
- [Testen von Übungsblätter](#)
- [Gestaltung des Programmierprojekts](#)
- [Gestaltung von Praxis-Übung für das Recap](#)
- [Weiterentwicklung des Testframeworks Util](#)
- [Orga-Team](#)
- [Technik-Leitung](#)

Tätigkeiten



Gestaltung von Übungsblättern (d.h. Erstellen der Übungsblätter)

Beschreibung

Die Aufgabe besteht in der Erstellung von Übungsblättern zur Vertiefung des Stoffes aus der Vorlesung. Die Tutoren sind für die Erstellung der Übungsblätter verantwortlich. Es wird vorgegeben, welche Inhalte jedes Übungsblatt abdecken muss. Der Tutor ist dann dafür verantwortlich, diese Inhalte in der Übung umzusetzen, einschließlich Konzeption, Aufgabenentwicklung, Bewertungsschema und Musterlösung.

Anzahl an Tutoren

Insgesamt ca. 8

Anzahl an Übungsblätter

14 (H00 - H13)

Zeitlicher Aufwand

22h/Monat

Voraussetzungen

Für die Erstellung der Übungsblätter sollten erfahrene Tutoren verantwortlich sein, da dies sowohl ein hohes Maß an Fachkenntnissen als auch didaktische Fähigkeiten erfordert.

Empfohlene Voraussetzungen

Es wird empfohlen, dass die Tutoren über Kenntnisse in der Testbarkeit von Code verfügen, um sicherzustellen, dass der erstellte Code testbar ist. Dies beinhaltet die Fähigkeit, Code so zu strukturieren und zu dokumentieren, dass er effektiv getestet werden kann. Darüber hinaus sollten die Tutoren über LaTeX-Kenntnisse verfügen, um die Übungsblätter professionell zu gestalten.

“

Reviewen von Übungsblättern

Beschreibung

Die Aufgabe besteht darin, die erstellten Übungsblätter zu reviewen, um sicherzustellen, dass sie verständlich und angemessen vom Umfang für Studierende sind und den Anforderungen entsprechen.

Anzahl an Übungsblätter

14 (H00 - H12 und Projekt)

Anzahl an Tutoren

4

Zeitlicher Aufwand

12h/Monat

“

Voraussetzungen

Keine spezifischen Voraussetzungen nötig.

Empfohlene Voraussetzungen

Fachwissen.

“

Entwicklung von JUnit-Tests (Testen von Übungsblättern)

Beschreibung

Die Bewertungskriterien jedes Übungsblattes und Projekts müssen überprüft werden. Dazu werden JUnit-Tests geschrieben, um sicherzustellen, dass diese Kriterien ordnungsgemäß erfüllt sind.

Anzahl an Übungsblätter

14 (H00 - H13)

Anzahl an Tutoren

Zeitlicher Aufwand

20h/Monat

Voraussetzungen

Die erfolgreiche Teilnahme an Veranstaltungen wie "Funktionale und Objektorientierte Programmierung" oder vergleichbaren Kursen, die Java vermitteln, wie z.B. "Einführung in die Programmierung", ist Voraussetzung. Zusätzlich sind Kenntnisse in JUnit und Testbarkeit erforderlich, um effektive Tests zu entwickeln.

Empfohlene Voraussetzungen

Erfahrung mit Mockito und Spoon wird empfohlen, um zusätzliche Möglichkeiten für automatisierte Tests zu nutzen und die Testabdeckung zu verbessern, einschließlich Bytecode-Transformationen. Grundkenntnisse in Java Reflection sind ebenfalls empfohlen.

“

Gestaltung von Praxis-Übung für das Recap

Beschreibung

Die Gestaltung von Praxis-Übungen für das Recap beinhaltet die Erstellung von Materialien, die darauf abzielen, mehr und einfachere Beispiele bereitzustellen. Dabei sollen Themen vorgestellt werden, die im Spiralcurriculum immer wieder auftauchen. Die Übungen sollen die Studierenden beim Debugging, Refactoring und der Verbesserung der Code-Qualität unterstützen. Des Weiteren sollen die Studierenden dazu ermutigt werden, Selbsttests durchzuführen und vorhandene Tests auszuführen. Diese Praxis-Übungen dienen als Grundlage für das Live-Coding im Recap.

Anzahl an Tutoren

1

Zeitlicher Aufwand

Variabel plus organisatorische Tätigkeiten.

Voraussetzungen

Ein erfahrener Tutor mit fundierten Kenntnissen im Debugging und in der Code-Qualität (Software Engineering), sowie Fachwissen in den relevanten Inhalten der Vorlesung.

Empfohlene Voraussetzungen

Es wird empfohlen, dass die Tutoren über Kenntnisse in LaTeX verfügt, um die Folien professionell zu gestalten.

“

Technik-Leitung

Beschreibung

Ein erfahrener Tutor, der sich speziell um die technischen Belange des Kurses kümmert. Zu den Aufgaben gehören die Verwaltung des Techstacks, Hosting, Continuous Integration (CI), den Grading-Server usw. Im Sommersemester wird der Tutor in diese Aufgaben eingeführt und eingeleitet, um sicherzustellen, dass er für das Wintersemester gut vorbereitet ist. Dies bedeutet, dass der Tutor auch im Wintersemester als Tutor fungieren wird.

Anzahl an Tutoren

1

Zeitlicher Aufwand

10 Stunden pro Monat

Voraussetzungen

Tutor im Sommersemester und im kommenden Wintersemester

Empfohlene Voraussetzungen

Erfahrung mit Docker, Linux und Git