
Modulbezeichnung: **Praktikum angewandte Systemsoftwaretechnik (PASST)** **10 ECTS**
 (Laboratory on Applied Systems Software Technology)

Modulverantwortliche/r: Jürgen Kleinöder

Lehrende: Maximilian Ott, Tobias Langer

Startsemester: SS 2020

Dauer: 1 Semester

Turnus: halbjährlich (WS+SS)

Präsenzzeit: 150 Std.

Eigenstudium: 150 Std.

Sprache: Deutsch

Lehrveranstaltungen:

ACHTUNG: Die notwendigen Veranstaltungen fallen im Sommersemester 2020 aufgrund der Aussetzung aller Präsenzveranstaltungen aus. Das Modul ist somit nicht belegbar.

Praktikum angewandte Systemsoftwaretechnik (fällt aus) (SS 2020, Praktikum, Anwesenheitspflicht, Tobias Langer et al.)

Es wird empfohlen, folgende Module zu absolvieren, bevor dieses Modul belegt wird:

Systemprogrammierung

Inhalt:

Im Praktikum erlernen und verwenden die Studierenden verschiedene Techniken und Prozesse der systemnahen Softwareentwicklung im OpenSource-Umfeld. Nähere Informationen zum konkreten Inhalt in diesem Semester finden sich auf der unter "Weitere Informationen" verlinkten Veranstaltungsseite.

Lernziele und Kompetenzen:

Fachkompetenz

Anwenden

Die Studenten

- verstehen Kodierrichtlinien und sind in der Lage diese situationsgemäß anzuwenden

Analysieren

Die Studenten

- erkunden gegebenen Programmcode grossen Umfangs und hoher Komplexität
- bestimmen dessen Funktionalität und beschreiben und diskutieren dies

Evaluiere (Beurteilen)

Die Studenten

- beurteilen Qualität, Korrektheit und Richtlinienkonformität fremder Programme
- analysieren in Softwaresystemen, insbesondere dem Linux-Kernel auftretende Fehler und Situationsbilder und können solche bewerten und auf ihr Vorkommen testen
- evaluieren und verwenden geeignete Mittel zur Erkennung der Fehlerursachen
- verifizieren die korrekte Behebung eines erkannten Fehlers
- beschreiben, bewerten und kritisieren das eigene und das Vorgehen Dritter bei der Programmentwicklung, Fehlersuche und Integration

Erschaffen

Die Studenten

- planen und entwickeln USB-Geräte und Dateisystemtreiber für das Betriebssystem Linux
- planen und entwickeln systemnahe Programme zur Interaktion mit Geräten im Zusammenspiel mit den entwickelten Treibern
- konzipieren, planen und entwickeln systemnahe Software, Systemsoftware oder Bestandteile eines Betriebssystemkerns; erstellen Dokumentation und präsentieren ihr Vorgehen
- erstellen geeignete Maßnahmen (Patches) zur Behebung erkannter Fehler und Probleme
- erzeugen textuelle Beschreibungen der Fehlerbedingungen, -symptome und -ursachen, der Fehlersuche sowie des Patches nach Kriterien relevanter Open-Source-Projekte in englischer Sprache, die geeignet sind, solche Patches und Beschreibungen an diese Open-Source-Projekte weiterzugeben
- entwickeln vorbeschriebene Patches und Beschreibungen und reichen diese bei relevanten Open-Source-Projekten ein

- interagieren mit den externen Entwicklern dieser Projekte mit dem Ziel eingereichtes Material akzeptiert zu bekommen
- erstellen Analysen und Präsentationen eigener und fremder Arbeit und tragen diese in geeigneter Weise vor einem Fachpublikum vor

Lern- bzw. Methodenkompetenz

Die Studenten

- verwenden gängige Werkzeuge der Softwareentwicklung im Open-Source-Bereich wie git, gdb, kgdb, qemu/kvm und cscope
- verstehen deren Funktionsweise
- verwenden diese erfolgreich in internen Aufgaben und Projekten sowie in der Interaktion mit externen Entwicklern
- interpretieren Code im Hinblick auf dessen vorgesehene Funktion, mögliche Fehler, sinnvolle Erweiterungspunkte und Qualitätsaspekte
- stellen technische, methodische und soziale Sachverhalte geeignet dar

Selbstkompetenz

Die Studenten

- sind in der Lage mit Kritik und Änderungswünschen umzugehen
- überwinden Berührungsängste im Kontakt mit externen Dritten
- bringen sich konstruktiv und produktiv in Open-Source-Projekte ein

Sozialkompetenz

Die Studenten

- organisieren selbständig die gemeinsame Bearbeitung der Übungsaufgaben und lösen diese kooperativ in kleinen Gruppen
- kommunizieren erfolgreich in englischer Sprache mit Betreuern und mit externen Entwicklern unter Einhaltung relevanter Protokolle im Open-Source-Umfeld
- gehen professionell mit Kritik an eigener Arbeit um und beziehen berechtigte Kritik in ihre zukünftige Arbeitsweise ein
- verhalten sich angemessen beim kritisieren fremder Arbeit gegenüber dem Ersteller dieser Arbeit oder Dritten
- erkennen und befolgen geschriebene und ungeschriebene Regeln im Umfeld relevanter Open-Source-Projekte; verhalten sich angemessen bei möglichen Konfliktsituationen

Verwendbarkeit des Moduls / Einpassung in den Musterstudienplan:

Das Modul ist im Kontext der folgenden Studienfächer/Vertiefungsrichtungen verwendbar:

[1] Berufspädagogik Technik (Master of Education)

(Po-Vers. 2018w | TechFak | Berufspädagogik Technik (Master of Education) | Gesamtkonto | Unterrichtsfach (Zweifach) inkl. Fachdidaktik | Informatik | Praktikum Informatik | Praktikum angewandte Systemsoftwaretechnik)

[2] Berufspädagogik Technik (Master of Education)

(Po-Vers. 2018w | TechFak | Berufspädagogik Technik (Master of Education) | Gesamtkonto | Unterrichtsfach (Zweifach) inkl. Fachdidaktik | Informatik | Praktikum Informatik | Praktikum angewandte Systemsoftwaretechnik)

Studien-/Prüfungsleistungen:

Praktikum angewandte Systemsoftwaretechnik (Prüfungsnummer: 113845)

(englische Bezeichnung: Laboratory on Applied Systems Software Technology)

Prüfungsleistung, Praktikumsleistung

Anteil an der Berechnung der Modulnote: 100%

weitere Erläuterungen:

Es sind semesterbegleitend sechs Übungsaufgaben zu bearbeiten. In der vorlesungsfreien Zeit findet ein Blockpraktikum statt (2 Wochen, Programmierung und zwei Vorträge a ca. 15 Minuten). Arbeit in 2er-Teams.

Übungsaufgaben, Programmierung im Blockpraktikum und Vorträge werden bepunktet. Die Note ergibt sich auf Basis der erreichten Punkte. Gewichtung Übungsaufgaben/Blockpraktikum 40/60.

Erstablingung: SS 2020, 1. Wdh.: keine Angabe

1. Prüfer: Jürgen Kleinöder
