

# Modulhandbuch BVS1

## Betriebssysteme und Verteilte Systeme 1

Bachelor Technische Informatik 2020

---

Version: 1 | Letzte Änderung: 02.08.2019 11:33 | Entwurf: 0 | Status: vom Modulverantwortlichen freigegeben |  
Verantwortlich: Vogt

### – Allgemeine Informationen

<b>Anerkannte Lehrveranstaltungen</b>	<u>BVS1_Vogt</u>
---	------------------

---

<b>Gültig ab</b>	Wintersemester 2021/22
------------------	---------------------------

---

<b>Dauer</b>	1 Semester
--------------	------------

---

<b>ECTS</b>	5
-------------	---

---

<b>Zeugnistext (de)</b>	Betriebssysteme und Verteilte Systeme 1
-------------------------	--

---

<b>Zeugnistext (en)</b>	Operating Systems and Distributed Systems 1
-------------------------	--

---

<b>Unterrichtssprache</b>	deutsch oder englisch
---------------------------	-----------------------

---

<b>abschließende Modulprüfung</b>	Ja
---------------------------------------	----

### Modulprüfung

---

<b>Benotet</b>	Ja
----------------	----

**Konzept**

Die Studierenden sollen in einer schriftlichen Klausur folgende Fähigkeiten nachweisen, die die o.a. Kompetenzen K1-K4 und K6-K9 abdecken: 1.) Sicherer Umgang mit grundlegenden Begrifflichkeiten, Konzepten und Techniken, 2.) Anwendung programmiersprachlicher und abstrakterer Konstrukte zur Lösung von Anwendungsproblemen bei der nebenläufigen und verteilten Programmierung, 3.) Prüfung von Lösungsvorschlägen auf Korrektheit. Typische Aufgabenformen zu 1.) sind Multiple-Choice-Fragen, Lückentexte, Bewertung von Aussagen hinsichtlich ihrer Korrektheit, zu 2.) Lösung kleinerer umgangssprachlich formulierter Probleme durch Programmstücke oder in abstrakterer Form und zu 3.) das Finden von Fehlern in Aussagen und vorgegebenen Programmstücken. Die übrigen Kompetenzen (K10, K13, K15, K16) lassen sich in einer schriftlichen Klausur begrenzter Dauer nur schwer sinnvoll prüfen. Sie werden daher im Praktikum geprüft, das als studienbegleitende Vorleistung absolviert werden muss.

---

**Frequenz**

Jedes Semester

## – Allgemeine Informationen

### Inhaltliche Voraussetzungen

**PI1 -  
Praktische  
Informatik 1**                      Sicherer Umgang mit einer  
   Programmiersprache.

---

**PI2 -  
Praktische  
Informatik 2**                      Sicherer Umgang mit einer  
   Programmiersprache.

---

**DR -  
Digitalrechner**                      Kenntnisse über Aufbau und  
   Funktionalität eines  
   Digitalrechners.

---

**GSP -  
Grundlagen  
der  
Systemprogrammierung**                      Grundkenntnisse über die  
   hardwarenahe Programmierung  
   eines Digitalrechners.

---

**NP -  
Netze und  
Protokolle**                      Grundkenntnisse über Internet-  
   Protokolle.

### Handlungsfelder

Systeme zur Verarbeitung, Übertragung und  
Speicherung von Informationen für technische  
Anwendungen planen, realisieren und integrieren

---

Anforderungen, Konzepte und Systeme analysieren  
und bewerten

---

Informationstechnische Systeme und Prozesse  
organisieren und betreiben

---

Mit Auftraggebern, Anwendern, gesellschaftlichem  
Umfeld und Teammitgliedern interagieren

### Learning Outcomes

<b>ID</b>	<b>Learning Outcome</b>
LO1	<p>Was: Das Modul vermittelt Kompetenzen zum Umgang mit Betriebssystemen und Diensten in verteilten Systemen sowie ein tieferes Verständnis der zugrundeliegenden Konzepte und Techniken. Im Fokus dieses ersten Moduls (gefolgt von BVS2) steht die Nutzung von Programmier- und -schnittstellen, die eine Systemsoftware typischerweise zur Realisierung nebenläufiger, kooperierender Software im lokalen und verteilten Umfeld anbietet. In praktischer Arbeit analysieren die Studierenden Problemstellungen im Systemumfeld (K1, K2, K4), implementieren Lösungen auf der Grundlage anerkannter Konzepte und Methoden (K3) mit Hilfe von Standardwerkzeugen (K6, K9) und prüfen sie (K7). Sie recherchieren dazu in Dokumentationen (K8, K15) und passen vorhandene Software an (K10).</p> <p>Womit: Der Dozent vermittelt Wissen und Basisfertigkeiten in einem Vorlesungs-/Übungsteil und betreut darauf aufbauend ein Praktikum. In den Übungen und insbesondere im Praktikum arbeiten die Studierenden in Kleingruppen und verteidigen ihre Lösungen (K8, K13, K16).</p> <p>Wozu: Systemsoftware, also Betriebssysteme und Dienstsoftware für verteilte Systeme, bietet die Plattform zur Erstellung von Anwendungen, die nebenläufig und verteilt arbeiten - Eigenschaften, die für heutige komplexe Softwaresysteme typisch sind. Entsprechende Programmierkenntnisse und Wissen über die zugehörigen Grundlagen sind somit essentiell für die Erstellung moderner Software (HF1). Durch ihre praktische Programmierarbeit</p>

erwerben die Studierenden Erfahrungen, die wichtig sind für die Erfassung von Anforderungen, die Entwicklung von Konzepten zur technischen Lösung und zu ihrer Bewertung (HF2) sowie zur Organisation von Prozessen und zum Betrieb von Systemen, die nebenläufig und verteilt arbeiten (HF3). Die Durchführung im Team mit dem Dozenten als "Auftraggeber" stärkt die Interaktionsfähigkeit der Studierenden (HF 4).

## Kompetenzen

Kompetenz	Ausprägung
Typische Werkzeuge, Standards und Best Practices der industriellen Praxis kennen und einsetzen	diese Kompetenz wird vermittelt
In Systemen denken	diese Kompetenz wird vermittelt
fachliche Probleme abstrahieren und formalisieren	diese Kompetenz wird vermittelt
Konzepte und Methoden der Informatik, Mathematik und Technik kennen und anwenden	diese Kompetenz wird vermittelt
In vorhandene Systeme einarbeiten und vorhandene Komponenten sinnvoll nutzen	diese Kompetenz wird vermittelt
Systeme analysieren	diese Kompetenz wird vermittelt
Systeme realisieren	diese Kompetenz wird vermittelt
Systeme prüfen	diese Kompetenz wird vermittelt
Informationen beschaffen und auswerten; Technische Zusammenhänge darstellen und erläutern	diese Kompetenz wird vermittelt

Komplexe technische  
Aufgaben im Team  
bearbeiten

diese Kompetenz wird  
vermittelt

Befähigung zum  
lebenslangen Lernen

diese Kompetenz wird  
vermittelt

Kommunikative und  
interkulturelle  
Fähigkeiten anwenden

diese Kompetenz wird  
vermittelt

## – Vorlesung / Übungen

<b>Typ</b>	Vorlesung / Übungen
------------	---------------------

<b>Separate Prüfung</b>	Nein
-------------------------	------

<b>Exemplarische inhaltliche Operationalisierung</b>	Die grundlegenden Komponenten und Aufgaben von Betriebssystemen und verteilten Systemen werden eingeführt und am Beispiel aktueller Systeme illustriert. Aufbauend darauf werden Programmierschnittstellen und Dienste zur Implementierung nebenläufiger Vorgänge betrachtet. Die Studierenden werden damit in die Lage versetzt, nebenläufig arbeitende, kommunizierende und kooperierende Programme zu verstehen und selbst zu erstellen.
--	---

## – Praktikum

<b>Typ</b>	Praktikum
------------	-----------

<b>Separate Prüfung</b>	Ja
-------------------------	----

<b>Exemplarische inhaltliche Operationalisierung</b>	Die Studierenden lernen, Konzepte und Techniken der nebenläufigen und verteilten Programmierung selbstständig zur Lösung von Anwendungsproblemen anzuwenden und dabei die Programmierschnittstellen von Systemsoftware zu nutzen. Dazu beschaffen sie auch selbstständig Informationen und passen vorhandene Software an.
--	---

### Separate Prüfung

<b>Benotet</b>	Nein
----------------	------

<b>Frequenz</b>	Einmal im Jahr
-----------------	----------------

<b>Voraussetzung für Teilnahme an Modulprüfung</b>	Ja
--	----

**Konzept**

Die Studierenden schließen sich zu Kleingruppen zusammen. Jede Kleingruppe absolviert mehrere "Praktikumsrunden" mit zugewiesenen Laborterminen. In jeder Runde werden Programmieraufgaben gelöst. Zur Vorbereitung eines Labortermins muss ein "Vorbereitungsblatt" praktisch gelöst werden. Die dabei erworbenen Kenntnisse werden zu Beginn des Termins geprüft (kurzer schriftlicher Eingangstest, persönliches Gespräch mit dem Betreuer). Wird diese Prüfung nicht bestanden, so muss ein Folgetermin wahrgenommen werden; im Wiederholungsfall führt dies zum Nichtbestehen des Praktikums. Im Erfolgsfall wird ein "Laborarbeitsblatt" mit weiteren Aufgaben unter Aufsicht (und ggf. mit Hilfestellung) bearbeitet. Durch diese stufenweise Vorgehensweise können sämtliche aufgeführte Kompetenzen individuell abgeprüft werden.