

# Lehrveranstaltungshandbuch GUI

Graphische Oberflächen und Interaktion

Version: 1 | Letzte Änderung: 10.09.2019 16:43 | Entwurf: 0 | Status: vom verantwortlichen Dozent freigegeben

## – Allgemeine Informationen

**Langname** Graphische Oberflächen und Interaktion

**Anerkennende LModule** [GUI\\_BaTIN](#)

**Verantwortlich** Prof. Dr. Dieter Rosenthal  
Professor Fakultät IME

**Gültig ab** Wintersemester 2021/22

**Niveau** Bachelor

**Semester im Jahr** Wintersemester

**Dauer** Semester

**Stunden im Selbststudium** 60

**ECTS** 5

**Dozenten** Prof. Dr. Dieter Rosenthal  
Professor Fakultät IME

**Voraussetzungen** Prozedurale und objektorientierte Programmierung in C/C++ und Java  
Struktur und Funktionalität von Betriebssystemen

**Unterrichtssprache** deutsch

**separate Abschlussprüfung** Ja

### Literatur

keine

### Abschlussprüfung

**Details**

Regelfall (bei großer  
Prüfungszahl:  
schriftliche Klausur)

Die Studierenden sollen  
im mündlichen  
Fachgespräch  
(gegebenenfalls auch in  
einer schriftlichen  
Klausur) folgende  
Kompetenzen  
nachweisen: 1.) Sicherer  
Umgang mit  
grundlegenden  
Begrifflichkeiten, 2.)  
Anwendung  
programmiersprachlicher  
Konstrukte zur Lösung  
von  
Anwendungsproblemen  
im Bereich der  
Programmierung  
graphischer  
Benutzeroberflächen,  
(Bei schriftlichen  
Prüfungen auch: 3.)  
Prüfung von Aussagen  
und  
programmiersprachlicher  
Konstrukte auf  
Korrektheit. Typische  
Aufgabenformen  
schriftlich sind zu 1.)  
sind Multiple-Choice-  
Fragen, Lückentexte,  
Bewertung von  
Aussagen hinsichtlich  
ihrer Korrektheit, zu 2.)  
Lösung kleinerer  
umgangssprachlich  
formulierter Probleme  
durch Programmstücke  
und zu 3.) das Finden  
von Fehlern in  
Aussagen und in  
vorgegebenen  
Programmstücken.)

---

**Mindeststandard**

Mindestens 50% der  
möglichen  
Gesamtpunktzahl.

---

**Prüfungstyp**

mündliche Prüfung,  
strukturierte Befragung

## – Vorlesung / Übungen

### Lernziele

Zieltyp	Beschreibung
Kenntnisse	grundlegende Begriffe und Techniken der Erstellung graphischer Benutzeroberflächen unter Microsoft Windows Fensterkonzept Nachrichtenkonzept
Kenntnisse	Anwenden der durch das Betriebssystem gegebenen API in Java Einbinden der Pakete AWT und Swing Diskussion der Vor- und Nachteile
Fertigkeiten	Umgang mit der Microsoft Windows API
Fertigkeiten	Umgang mit AWT- und Swing-Klassen zur Erstellung von graphischen Benutzeroberflächen in Java

### Aufwand Präsenzlehre

Typ	Präsenzzeit (h/Wo.)
Vorlesung	2
Übungen (ganzer Kurs)	1
Übungen (geteilter Kurs)	1
Tutorium (freiwillig)	0

### Besondere Voraussetzungen

prozedurale und objektorientierte Programmierung in C/C++ und Java, Struktur und Funktionalität von Betriebssystemen

### Begleitmaterial

elektronische Vortragsfolien zur Vorlesung, elektronische Sammlung von Übungsaufgaben, elektronische Sammlung von Beispielprogrammen, links auf relevante Webseiten (insbesondere Schnittstellendokumentationen, Software-Beispiele und frei verfügbare Softwareentwicklungsumgebungen)

### Separate Prüfung

Nein

## – Praktikum

### Lernziele

Zieltyp	Beschreibung
Kenntnisse	Programmierung von Graphischen Benutzeroberflächen am Beispiel individueller Aufgaben - Details wie unter "Vorlesung/Übung" angegeben
Fertigkeiten	Umgang mit Softwareentwicklungsumgebungen für GUIs
Fertigkeiten	Realisierung von GUI-Applikationen mittlerer Komplexität in kleinen Teams

### Aufwand Präsenzlehre

Typ	Präsenzzeit (h/Wo.)
Praktikum	1
Tutorium (freiwillig)	0

### Besondere Voraussetzungen

Prozedurale und objektorientierte Programmierung in C/C++ und Java

### Begleitmaterial

elektronische Vortragsfolien zur Vorlesung, elektronische Sammlung von Beispielprogrammen, Links auf relevante Webseiten (insbesondere Schnittstellendokumentationen, Software-Beispiele und frei verfügbare Softwareentwicklungsumgebungen)

### Separate Prüfung

Ja

### Separate Prüfung

### Prüfungstyp

praxisnahes Szenario bearbeiten (z.B. im Praktikum)

**Details**

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.

---

**Mindeststandard**

Erfolgreiche Teilnahme an allen Laborterminen, d.h. insbesondere selbstständige (ggf. mit Hilfestellung) Lösung der Programmieraufgaben.