

# Mitteilungsblatt

der Universität Innsbruck

<https://www.uibk.ac.at/universitaet/mitteilungsblatt/>

---

Studienjahr 2024/2025

Ausgegeben am 17. Juli 2025

89. Stück

---

792. Curriculum für den Universitätslehrgang **Universitätsstudiengang Programming and Software Engineering Foundations** an der Universität Innsbruck

---

*Das Mitteilungsblatt erscheint jeweils am 1. und 3. Mittwoch jeden Monats.*

*Eigentümer, Herausgeber, Vervielfältigung und Vertrieb: Büro der Rektorin der Universität Innsbruck, Innrain 52, A-6020 Innsbruck. Für den Inhalt verantwortlich: Dr. Veronika Allerberger-Schuller*

Beschluss der Curriculum-Kommission an der Fakultät für Mathematik, Informatik und Physik vom 22.04.2025, genehmigt mit Beschluss des Senats vom 11.07.2025.

Aufgrund des § 25 Abs. 1 Z 10a des Universitätsgesetzes 2002, BGBl. I Nr. 120, idgF, und des § 48a des Satzungsteiles „Studienrechtliche Bestimmungen“, verlautbart im Mitteilungsblatt der Universität Innsbruck vom 10.02.2022, 17. Stück, Nr. 277, idgF, wird verordnet:

Curriculum für den Universitätslehrgang  
**Universitätsstudiengang Programming and Software Engineering Foundations**  
an der Universität Innsbruck

**Inhaltsverzeichnis**

- § 1 Zulassung und Aufnahme
- § 2 Qualifikationsprofil
- § 3 Umfang und Dauer
- § 4 Lehrveranstaltungsarten und Teilungszahlen
- § 5 Pflichtmodule
- § 6 Prüfungsordnung
- § 7 Abschlusszeugnis
- § 8 Inkrafttreten

## **§ 1 Zulassung und Aufnahme**

- (1) Die Zulassung zum Universitätsstudiengang setzt die Reifeprüfung und eine mindestens 1.5-jährige Berufserfahrung voraus.
- (2) Über die Aufnahme in den Universitätsstudiengang entscheidet die Leiterin oder der Leiter des Universitätsstudiengangs. Informationen über die Modalitäten der Bewerbung und das Auswahlverfahren werden auf der Homepage der Universität Innsbruck veröffentlicht.
- (3) Personen, die in den Universitätsstudiengang aufgenommen wurden und die Lehrgangsgebühr entrichtet haben, werden vom Rektorat der Universität Innsbruck als außerordentliche Studierende zugelassen.

## **§ 2 Qualifikationsprofil**

- (1) Die Absolventinnen und Absolventen des Universitätsstudiengangs verfügen über umfassende Programmierkenntnisse, insbesondere in der objektorientierten Programmierung (OOP). Sie sind in der Lage, Anwendungen mit einfachen Benutzungsoberflächen umzusetzen. Sie können effiziente Algorithmen für konkrete Problemstellungen entwerfen und implementieren. Darüber hinaus besitzen sie die Fähigkeit, ihre Herangehensweise an technische Problemstellungen systematisch und strukturiert zu gestalten. Neben der Implementierung können Studierende Fehler in eigenem und fremdem Code identifizieren und diese beheben. Zudem entwickeln Studierende die Fähigkeit zu logisch-abstraktem und vernetztem Denken.
- (2) Die Absolventinnen und Absolventen sind mit klassischen sowie modernen Methoden der Softwareentwicklung vertraut und in der Lage, diese gezielt anzuwenden. Sie besitzen Kenntnisse im Umgang mit Versionskontrollsystemen, beherrschen den Einsatz von Build-Tools zur Automatisierung des Entwicklungsprozesses und wenden Tools zur Sicherstellung der Codequalität an. Darüber hinaus sind sie in der Lage, Testkonzepte zu erstellen und verschiedene Teststrategien zur Validierung von Softwarelösungen praktisch umzusetzen. Zudem erwerben die Studierenden Kompetenzen im zielgerichteten Einsatz von KI-Tools in verschiedenen Phasen des Softwareentwicklungslebenszyklus. Dies ermöglicht ihnen, die Effizienz der Softwareentwicklung durch Automatisierung und kontinuierliche Optimierung zu steigern. Beispiele hierfür sind die automatisierte Extraktion von Anforderungen aus Dokumenten, oder die Generierung von Codeabschnitten. Darüber hinaus erlernen sie, Tools und Methoden aus unterschiedlichen Quellen und Disziplinen selbstständig zu recherchieren, relevante Inhalte zu identifizieren, diese kritisch zu bewerten, auszuwählen und zielgerichtet für die Softwareentwicklung einzusetzen.
- (3) Die Absolventinnen und Absolventen sind in der Lage, sowohl agile als auch klassische Projektmanagementmethoden anzuwenden und anspruchsvolle sowie vielschichtige Softwareprojekte eigenständig zu realisieren – von der Anforderungsanalyse über die Planung und Umsetzung bis hin zur Testung und dem Deployment. Sie übernehmen dabei verschiedene Rollen und Verantwortlichkeiten wie Projektmanagerinnen und Projektmanager, Entwicklerinnen und Entwickler, Testerinnen und Tester und DevOps-Ingenieurinnen und Ingenieure. Sie können dabei die jeweiligen Methoden flexibel an die Anforderungen des Projekts anpassen, Ressourcen effizient managen, Risiken identifizieren, Normen, Vorschriften und Regeln einhalten und entsprechende Maßnahmen zur Risikominimierung ergreifen, um eine termingerechte und qualitativ hochwertige Fertigstellung des Projekts sicherzustellen. Darüber hinaus entwickeln sie Soft-Skills, um im Team zu arbeiten, Konflikte konstruktiv zu lösen und effektiv zu kommunizieren.
- (4) Die erworbenen Qualifikationen eröffnen Absolventinnen und Absolventen vielfältige berufliche Perspektiven in der digitalen Arbeitswelt, einschließlich der Übernahme leitender Positionen. Dabei sind Absolventinnen und Absolventen in der Lage, Zusammenhänge zwischen ökologischen, ökonomischen und sozialen Mechanismen zu verstehen, Querverbindungen zu schaffen, die daraus gewonnenen Erkenntnisse in verschiedenen Situationen anzuwenden sowie zu neuen Sachverhalten Stellung zu beziehen, dabei den eigenen Standpunkt zu erläutern und adressatengerecht unter Verwendung der Fachsprache zu präsentieren. Dies sind insbesondere Aufgabenbereiche mit Fokus auf einzelne Phasen des Softwarelebenszyklus sowie koordinierende Tätigkeiten im Zusammenspiel zwischen IT-Systemen und organisatorischen Prozessen. Zu den relevanten Berufsbildern gehören unter anderem Produktmanagement, Softwareentwicklung, Projektleitung,

Verantwortlichkeit für Digitalisierungsvorhaben, IT-Beratung sowie die Rolle als IT-Unternehmerinnen und Unternehmer.

### § 3 Umfang und Dauer

Der Universitätsstudiengang umfasst 30 ECTS-Anrechnungspunkten (im Folgenden: ECTS-AP). Ein ECTS-AP entspricht einer Arbeitsbelastung von 25 Stunden. Der Universitätsstudiengang wird berufsbegleitend angeboten und erstreckt sich über zwei Semester.

### § 4 Lehrveranstaltungsarten und Teilungszahlen

Prüfungsimmanente Lehrveranstaltungen:

- (1) Proseminare (PS) führen interaktiv in ein Fachgebiet ein und vermitteln Kenntnisse und Methoden des wissenschaftlichen Arbeitens. Teilungszahl: 25.
- (2) Vorlesungen verbunden mit Übungen (VU) dienen zur praktischen Bearbeitung konkreter Aufgaben eines Fachgebiets, die sich im Rahmen des Vorlesungsteils stellen. Teilungszahl: 25.

### § 5 Pflichtmodule

Es sind folgende Pflichtmodule im Ausmaß von insgesamt 30 ECTS-AP zu absolvieren:

1.	Pflichtmodul: Grundlagen der Programmierung	SSt	ECTS-AP
a.	<b>VU Grundlagen der Programmierung</b> Die Lehrveranstaltung vermittelt grundlegende Programmierkonzepte und -techniken, sowie fördert das algorithmische Denken.	3	5
b.	<b>VU Erweiterte Programmierkonzepte</b> Die Lehrveranstaltung baut auf die VU Grundlagen der Programmierung auf und erweitert diese um fortgeschrittene Programmierkonzepte. Insbesondere werden objektorientierte Prinzipien behandelt und einzelne Themen aus dem Bereich Algorithmen und Datenstrukturen vermittelt.	3	5
	<b>Summe</b>	<b>6</b>	<b>10</b>
	<b>Lernergebnisse:</b> ad a: Die Studierenden <ul style="list-style-type: none"> <li>sind in der Lage, die Bedeutung von Syntax und Semantik in einer Programmiersprache zu verstehen;</li> <li>sind in der Lage, eine Entwicklungsumgebung zu verwenden und können diese effizient zur Umsetzung von Programmen einsetzen;</li> <li>sind in der Lage, grundlegende Programmierkonzepte wie Datentypen, Operatoren, Variablen, Kontrollstrukturen, Methoden, und Datenstrukturen anzuwenden;</li> <li>sind in der Lage, die erlernten Programmierkonzepte zu kombinieren und anzuwenden, um Algorithmen zu erstellen und damit Problemstellung zu lösen;</li> <li>sind in der Lage, Probleme im Code schrittweise zu erfassen und selbständig zu beheben.</li> </ul> ad b: Die Studierenden <ul style="list-style-type: none"> <li>verfügen über Grundkenntnisse der Konzepte der objektorientierten Programmierung;</li> <li>können die verschiedenen Konzepte der Vererbung, einschließlich Klassen, Interfaces und abstrakter Klassen, in objektorientierten Programmiersprachen anwenden;</li> <li>sind befähigt weiterführende Programmierkonzepte bei diversen Problemstellungen anzuwenden;</li> <li>fördern ihr algorithmisches Denken und sind in der Lage, die Effizienz einfacher Algorithmen zu analysieren und ihre Komplexität zu bewerten.</li> </ul>		

	<b>Anmeldungsvoraussetzung/en:</b> keine
--	--

2.	<b>Pflichtmodul: Moderne Software Engineering Prozesse &amp; Methoden</b>	<b>SSt</b>	<b>ECTS-AP</b>
<b>a.</b>	<b>VU Software Engineering Tools &amp; KI Praktiken</b> Die Lehrveranstaltung vermittelt den Studierenden praxisorientierte Kenntnisse und Fähigkeiten im Umgang mit Software Engineering Tools und modernen (KI-)Technologien zur Unterstützung von Softwareentwicklungsprozessen.	3	5
<b>b.</b>	<b>VU Moderne Software Engineering Prozesse &amp; Methoden</b> Die LV baut auf der VU Software Engineering Tools & KI Praktiken auf und vermittelt vertiefte Kenntnisse, die für die ganzheitliche Umsetzung eines Softwareprojekts erforderlich sind. Dabei spielen sowohl erweiterte Software Engineering-Praktiken als auch Konzepte des Projektmanagements eine entscheidende Rolle.	3	5
	<b>Summe</b>	<b>6</b>	<b>10</b>
	<b>Lernergebnisse:</b> ad a: Die Studierenden <ul style="list-style-type: none"> <li>sind in der Lage, verschiedene Konzepte zur Codeverwaltung zu verstehen, und Code damit zu verwalten;</li> <li>sind in der Lage, Build-Prozesse zu automatisieren, indem sie geeignete Tools und Skripte einsetzen;</li> <li>sind in der Lage, Unit-Tests zu entwickeln, zur Sicherstellung der korrekten Funktionsweise von Softwarekomponenten;</li> <li>sind in der Lage, einen einfachen UI-Prototyp zu entwickeln und diesen in einer funktionalen, benutzerfreundlichen Oberfläche umzusetzen;</li> <li>sind befähigt, moderne (KI-)Tools im Bereich Software Engineering situationsgerecht einzusetzen.</li> </ul> ad b: Die Studierenden <ul style="list-style-type: none"> <li>sind in der Lage, die grundlegenden Prozesse und Vorgehensmodelle im Software-Design und Planung zu verstehen;</li> <li>sind in der Lage, unterschiedliche Arten von Anforderungen zu erheben, und sind befähigt Anforderungen selbständig zu formulieren;</li> <li>sind befähigt, verschiedene Maßnahmen zur Sicherstellung der Codequalität, sowie geeignete Tools zur Analyse und Bewertung zu benutzen;</li> <li>sind in der Lage, UML-Diagramme zu erstellen und zu interpretieren, um die Struktur und das Verhalten von Softwaresystemen zu modellieren;</li> <li>sind befähigt, verschiedene Testkonzepte und Strategien zu beurteilen.</li> </ul>		
	<b>Anmeldungsvoraussetzung/en:</b> keine		

3.	<b>Pflichtmodul: Software Engineering Lab</b>	<b>SSt</b>	<b>ECTS-AP</b>
	<b>PS Software Engineering Lab</b> Ziel dieses Moduls ist es, dass die Studierenden ein Softwareprodukt von der Konzeption über die Entwicklung bis hin zur Abnahme eigenständig	4	10

	umzusetzen. Dabei lernen die Studierenden wichtige Schritte der Softwareentwicklung praxisnah kennen und wenden die erlernten Fähigkeiten aus VU Software Engineering Tools & KI Praktiken und VU Moderne Software Engineering Prozesse & Methoden an.		
	<b>Summe</b>	<b>4</b>	<b>10</b>
	<b>Die Studierenden</b> <ul style="list-style-type: none"> <li>• sind in der Lage, erworbenen Fähigkeiten – wie das Entwerfen von Algorithmen, das Erstellen von Tests und Anwenden von Maßnahmen zur Sicherstellung der Softwarequalität – in der Praxis anzuwenden und zu kombinieren, um ein Softwareprojekt in mehreren Iterationszyklen eigenständig umzusetzen;</li> <li>• sind in der Lage, in Softwareprojekten in verschiedenen Rollen verantwortlich mitzuwirken und teamorientiert zu arbeiten;</li> <li>• sind in der Lage, Projektmanagement-Tools praxisorientiert anzuwenden, um Projekte effizient zu planen, zu organisieren und zu steuern.</li> </ul>		
	<b>Anmeldungsvoraussetzung/en:</b> keine		

## § 6 Prüfungsordnung

- (1) Die Leistungsbeurteilung der Lehrveranstaltungen der Module erfolgt durch Lehrveranstaltungsprüfungen, wobei bei prüfungsimmanenten Lehrveranstaltungen die Beurteilung aufgrund von mindestens zwei schriftlichen, mündlichen und/oder praktischen Beiträgen der Teilnehmerinnen und Teilnehmer erfolgt.
- (2) Die Lehrveranstaltungsleiterin bzw. der Lehrveranstaltungsleiter hat vor Beginn des Universitätsstudiengangs die Methode der Leistungsbeurteilung und die Beurteilungskriterien festzulegen und bekanntzugeben.

## § 7 Abschlusszeugnis

Nach erfolgreichem Abschluss wird den Absolventinnen und Absolventen des Universitätsstudiengangs ein Abschlusszeugnis ausgestellt.

## § 8 Inkrafttreten

Dieses Curriculum tritt einen Monat nach Kundmachung im Mitteilungsblatt in Kraft.

Für die Curriculum-Kommission:

Univ.-Prof. Dr. Gerhard Kirchmair

Für den Senat:

Univ.-Prof. Mag. Dr. Walter Obwexer