

COURSE DESCRIPTION – ACADEMIC YEAR 2024/2025

Course title	Ubiquitous Computing
Course code	42427
Scientific sector	ING-INF/05
Degree	Bachelor in Elektrotechnik und cyberphysischen Systemen
Semester	2
Year	I
Credits	6
Modular	Yes

Total lecturing hours	40
Total lab hours	20
Attendance	Die Teilnahme ist nicht verpflichtend, wird jedoch dringend empfohlen, da viele Laborarbeiten eine entsprechend Software- und Hardware-Infrastruktur erfordern. Nicht anwesende Studierende müssen zu Beginn des Kurses den Dozenten kontaktieren, um Unterstützung zu den Modalitäten des eigenständigen Studiums zu erhalten.
Prerequisites	-
Course page	TEAMS

Specific educational objectives	Von der Idee bis zur Entwicklung eines funktionalen Prototyps – das ist das zentrale Thema dieses Kurses. Durch die Integration drahtloser Kommunikationstechnologien können Geräte miteinander sowie mit ihrer Umgebung interagieren, was die Möglichkeiten mobiler Anwendungen erheblich erweitert. In diesem Kurs wird gezeigt, wie smarte Sensoreinheiten von der Planung bis zur Umsetzung implementiert werden können. Die Teilnehmenden erlernen die Grundlagen der Elektronik, Mikrocontroller-Programmierung und das Prototyping mithilfe der Arduino/ESP32-Plattform. Zudem erhalten sie einen Einblick in den Einsatz von digitalen und analogen Sensoren, LEDs und Motoren, um intelligente Prototypen zu entwickeln. Auf diese Weise erwerben die Studierenden ein umfassendes Verständnis der drahtlosen Netzwerktechnologien sowie der grundlegenden Sensortechnik.
--	--

Lecturer	Prof. Michael Haller
Contact	michael.haller@unibz.it
Scientific sector of lecturer	ING-INF/05
Teaching language	German
Office hours	
Lecturing assistant (if any)	
Contact LA	
Office hours LA	
List of topics	<ul style="list-style-type: none"> • Arduino Output (mit einem starken Fokus auf Programmierung, Digital, Analog, PWM, ADC) • Arduino Input: Schalter, Arbeiten mit Sensoren • AdvancedIO: FSR, Dehnungssensoren, Empfindlichkeit, Offset, Genauigkeit, Dynamikbereich, Linearität und Rauschen, Signalfilterung (Mittelwertfilter, EMA, WEMA etc.)

	<ul style="list-style-type: none"> • Motoren: ERM/LRA, Interrupts, Speicherverwaltung • Serielle Kommunikation (Sensoren zu Mikrocontroller, Mikrocontroller zu Computer, etc.), I2C, Serielles Peripherie-Interface (SPI) • Drahtlose Sensornetzwerke (z. B. Kommunikationsarchitektur, Protokolle, Sensorknoten etc.), WLAN/IEEE 802.11, Bluetooth/IEEE • MIDI/OSC-Kommunikation • ESP32-Architektur & Graphik am ESP32-Display • Designrichtlinien & Prototyping-Fähigkeiten für die Gestaltung und Entwicklung vernetzter Sensoreinheiten • Technologietrends
Teaching format	Frontalunterricht und Laborarbeiten (Workshops)
Learning outcomes	<ul style="list-style-type: none"> • Arduino Output (mit einem starken Fokus auf Programmierung, Digital, Analog, PWM, ADC) • Arduino Input: Schalter, Arbeiten mit Sensoren • AdvancedIO: FSR, Dehnungssensoren, Empfindlichkeit, Offset, Genauigkeit, Dynamikbereich, Linearität und Rauschen, Signalfilterung (Mittelwertfilter, EMA, WEMA etc.) • Motoren: ERM/LRA, Interrupts, Speicherverwaltung • Serielle Kommunikation (Sensoren zu Mikrocontroller, Mikrocontroller zu Computer, etc.), I2C, Serielles Peripherie-Interface (SPI) • Drahtlose Sensornetzwerke (z. B. Kommunikationsarchitektur, Protokolle, Sensorknoten etc.), WLAN/IEEE 802.11, Bluetooth/IEEE • MIDI/OSC-Kommunikation • ESP32-Architektur & Graphik am ESP32-Display • Designrichtlinien & Prototyping-Fähigkeiten für die Gestaltung und Entwicklung vernetzter Sensoreinheiten • Technologietrends
Assessment	Die Aufgaben werden in kleineren Teams (jeweils zu zweit) bearbeitet, wobei es zu Beginn des Semesters auch individuelle Aufgaben geben wird. Diese haben das Ziel, allen Teilnehmenden die nötigen Erfahrungen und ein vertieftes Verständnis für die Gestaltung und Umsetzung vernetzter Sensoreinheiten zu vermitteln. Ohne diese Grundlagen wäre es später im Semester schwierig, das geplante Projekt erfolgreich umzusetzen.
Assessment language	Deutsch
Assessment Typology	
Evaluation criteria and criteria for awarding marks	Jede*r Studierende erhält ein Hardware-Kit, um den Umgang mit Sensoren, Displays und Aktuatoren zu erlernen. Durch praxisnahe Beispiele während des Unterrichts erwerben die Teilnehmenden grundlegende Fähigkeiten und lernen, typische Schaltungen zu erstellen, die miteinander kommunizieren können. Parallel dazu

	<p>arbeiten sie im Rahmen eines semesterlangen Gruppenprojekts daran, ein komplexes Vorhaben von der Idee bis zur Umsetzung zu entwickeln. Die Studierenden werden ermutigt, frühzeitig einen funktionsfähigen Prototyp zu erstellen, den sie anschließend durch Tests weiter verfeinern können. Am Semesterende präsentieren sie ihre Projekte den anderen Teilnehmenden.</p>
<p>Required readings</p>	<ul style="list-style-type: none"> • Steffen Wendze; IT-Sicherheit für TCP/IP- und IoT-Netzwerke: Grundlagen, Konzepte, Protokolle, Härtung; ISBN: 3658334223 • Walter Trojan; Das MQTT-Praxisbuch: Mit ESP8266 und Node-RED; ISBN: 3895763241 • Pradeeka Seneviratne; Beginning LoRa Radio Networks with Arduino: Build Long Range, Low Power Wireless IoT Networks; ISBN: 1484243560 • Ayhan Polat; Das Internet of Things. Ein Literatur Review zum aktuellen Forschungsstand; ISBN: 3668536503 • The Official Raspberry Pi Projects Bool; WEB: https://magpi.raspberrypi.org/books/projects-1
<p>Supplementary readings</p>	<p>-</p>
<p>Software used</p>	<p>Verschiedene Mikrocontroller und Kits werden verwendet. Nur teilnehmende Studierende, die am Unterricht teilnehmen, können diese während der Unterrichtszeit nutzen. Außerdem werden wir hauptsächlich ProtoPie, Visual Studio Code, Arduino IDE, PlatformIO und Unity verwenden, die den Studierenden zur Verfügung stehen. Weitere Informationen sind ausschließlich auf TEAMS verfügbar.</p>

COURSE DESCRIPTION – ACADEMIC YEAR 2024/2025

Course title	Ubiquitous Computing
Course code	42427
Scientific sector	ING-INF/05
Degree	Bachelor in Electronics and Cyberphysical Systems (L-8)
Semester	2 nd
Year	I
Credits	6
Modular	Yes

Total lecturing hours	40
Total lab hours	20
Attendance	Attendance is not compulsory, but highly recommended as many labs require an adequate software and hardware infrastructure; non-attending students may contact the lecturer at the start of the course to get support on the modalities of the independent study.
Prerequisites	-
Course page	TEAMS

Specific educational objectives	By building an idea, designers are challenged to "build to think" and thus gain deeper insights. The integration of wireless communication technologies gives devices the ability to interact with each other and their environment, extending the possibilities of mobile applications further. This course will go beyond early physical prototyping and show how to implement smart sensing devices that can communicate together (from the design to the implementation). Participants learn basic electronics, microcontroller programming, and physical prototyping using the Arduino/ESP32 platform, then use digital and analog sensors, LED lights and motors to build, program and customize smart prototypes. Moreover, students will get enough theoretical background for designing & developing their own physical prototypes that at the end can communicate together. Therefore, students will gain a profound understanding of wireless network technologies and sensor technology fundamentals.
--	---

Lecturer	Prof. Michael Haller
Contact	michael.haller@unibz.it
Scientific sector of lecturer	ING-INF/05
Teaching language	German
Office hours	
Lecturing assistant (if any)	
Contact LA	
Office hours LA	
List of topics	<ul style="list-style-type: none"> • Arduino Output (with a strong focus on programming, digital, analog, PWM, ADC) • Arduino Input: Switches, Debouncing, Playing with sensors • AdvancedIO: FSR, stretch sensors, sensitivity, offset, accuracy, dynamic range, linearity and noise, filtering the signals (moving mean filter, EMA, WEMA etc.) • Motors: ERM/LRA, Interrupts, Memory Handling

	<ul style="list-style-type: none"> Serial Communication (Sensors to Microcontroller, Microcontrollers to Computer, etc.), i2C, Series Peripheral Interface (SPI) Wireless Sensor Networks (e.g. communication architecture, protocols, sensor nodes etc.), WLAN/IEEE 802.11, Bluetooth/IEEE) MIDI/OSC communication ESP32 architecture & How to render widgets on a ESP32-Display Design Guidelines & Prototyping Skills for the design & development of interconnected sensing devices Technology Trends
Teaching format	Frontal lectures & labs

Learning outcomes	<ul style="list-style-type: none"> Arduino Output (with a strong focus on programming, digital, analog, PWM, ADC) Arduino Input: Switches, Debouncing, Playing with sensors AdvancedIO: FSR, stretch sensors, sensitivity, offset, accuracy, dynamic range, linearity and noise, filtering the signals (moving mean filter, EMA, WEMA etc.) Motors: ERM/LRA, Interrupts, Memory Handling Serial Communication (Sensors to Microcontroller, Microcontrollers to Computer, etc.), i2C, Series Peripheral Interface (SPI) Wireless Sensor Networks (e.g. communication architecture, protocols, sensor nodes etc.), WLAN/IEEE 802.11, Bluetooth/IEEE) MIDI/OSC communication ESP32 architecture & How to render widgets on a ESP32-Display Design Guidelines & Prototyping Skills for the design & development of interconnected sensing devices Technology Trends
--------------------------	---

Assessment	The major activity of the class is centered around a group project (in pairs of two), but there will be individual assignments early in the semester. The goal of these assignments is to ensure everyone in the class gains experience and understanding of the design and implementation of connected sensing devices, without which creating an interesting and sophisticated project will be difficult.
Assessment language	German
Assessment Typology	
Evaluation criteria and criteria for awarding marks	Each student group is provided with a physical computing kit including an Arduino/ESP32 compatible board as well as everything needed to learn how to use sensors, displays, and actuators. Through hands-on experiences during class periods, students acquire basic skills and learn to build a range of typical circuits that will communicate to each other. Along with basic skill acquisition, students are involved in a

	<p>semester-long group assignment in which they develop a complex project from start to finish. Students are encouraged to quickly arrive at a working prototype at which point they can fine-tune their project through testing. At the end of the semester, the projects are presented to the rest of the students.</p>
<p>Required readings</p>	<ul style="list-style-type: none"> • Steffen Wendze; IT-Sicherheit für TCP/IP- und IoT-Netzwerke: Grundlagen, Konzepte, Protokolle, Härtung; ISBN: 3658334223 • Walter Trojan; Das MQTT-Praxisbuch: Mit ESP8266 und Node-RED; ISBN: 3895763241 • Pradeeka Seneviratne; Beginning LoRa Radio Networks with Arduino: Build Long Range, Low Power Wireless IoT Networks; ISBN: 1484243560 • Ayhan Polat; Das Internet of Things. Ein Literatur Review zum aktuellen Forschungsstand; ISBN: 3668536503 • The Official Raspberry PI Projects Bool; WEB: https://magpi.raspberrypi.org/books/projects-1
<p>Supplementary readings</p>	<p>-</p>
<p>Software used</p>	<p>Different microcontrollers and microelectronics kits are used. Only participant students, who attend classes, can use them during class time. Moreover, we will mainly use ProtoPie, Visual Studio Code, Arduino IDE, PlatformIO, and Unity, all of which are available for the students. Further information is provided are on TEAMS.</p>