Computer Science exchange program

The Computer Science programme provides students with a broad profile in IT. Students will be in a position to develop, update and maintain IT systems on your own initiative, both in public institutions and in companies.

The programme in Computer Science combines theory with practical problem issues and provides relevant job qualifications. You will gain a thorough insight into computer methodology and concepts, business organization, economics, production management, and environmental conditions related to business.

The following elective elements give the student an opportunity to enhance their academic and professional competencies by specialising and putting themes into perspective within the wider scope of information technology.

Availability	Spring semester 2025

Course overview

Course title	Level	ECTS
Mobile Application Development	2nd year/Short Cycle/EQF level 5 (4th semester of the programme)	10
Advanced Software Construction	2nd year/Short Cycle/EQF level 5 (4th semester of the programme)	10
IT Security	2nd year/Short Cycle/EQF level 5 (4th semester of the programme)	10
Machine Learning	2nd year/Short Cycle/EQF level 5 (4th semester of the programme)	10

Semester / Course Prerequisites

A foreign qualification similar to 1½-2 years engineering, IT, SW or computer science studies (comprehensive knowledge of Java and SW-Design & UML and experienced C++ or Java programmer).

Zealand Academy of Technologies and Business

COURSE DESCRIPTION

COURSE: Mobile Application Development	10 ECTS

In this course you will learn how to build apps for Android.
The goal is to learn how to design and implement mobile applications (apps). We will cover best-practices, used in mobile development.

Course Content

This course is intended to qualify student to:
Design and program mobile application mainly for Android.

Android applications using the Kotlin programming language and the Android Studio IDE. An Android device (phone or tablet) and/or an Android emulator is used to run the applications. Cross-platform application development for Android + iOS.

Examination form / Assessment

There is compulsory study activity in this course. The first 10-11 weeks of the semester are general classes. For the last approx. 5 weeks, the student prepares an individual project in a self-selected topic related to the topic. The project work is used for the exam.

Learning Outcome

Knowledge	The student has knowledge about: • The architecture of the Android operating system. • The life-cycle of Android activities and fragments.
Skills	The student can: • Use a modern programming environment to program mobile application. • Design user interfaces for mobile applications. • Make mobile applications for various types of mobile devices. • Make mobile applications communicate with back-end systems using REST. • Use location-based services in mobile applications.
Competencies	• Design and program mobile application

Course: Advanced Software Construction	10 ECTS

The general intention of the course is to provide you with a deeper knowledge on several key topics in modern Software Development. The focus will be on obtaining practical, applicable skills rather than theoretical knowledge. In addition, the focus will primarily be on methodologies, programming techniques and language features rather than on specific dish-of-the-day technologies.

Course Content

• Advanced C# language features relevant and other features for building Frameworks incl. following issues.
• Reflection (dissecting types at run-time)

- A closer look at iteration (indexers, iterators and collection interfaces)
- Operator Overloading (giving meaning to operators like '+' and '-' on object level)
- Anonymous Types (creating new types at run-time)
- A closer look at LINQ
- A deeper dive into Design Patterns
- A deeper dive into the SOLID principles
- How to use state machine for multiple asynchronous input
- Advanced use of synchronous mechanism in C#
- How to use parallelism in C#

Examination form / Assessment

There is compulsory study activity in this course. The first 10-11 weeks of the semester are general classes. For the last approx. 5 weeks, the student prepares an individual project in a self-selected topic related to the topic. The project work is used for the exam.

Learning Outcome

Knowledge	The student get knowledge about: • Several important advanced C# language features (reflection, operator overload, state machines, regular expressions) • Purpose and usability of the SOLID principles • The diversity of Design Patterns • Problems and solutions to concurrency
Skills	After completing the course, you should have acquired the skills to: • Actively use a larger part of the C# language features when developing software e.g. Frameworks. • Develop software that adheres to the SOLID principles deviations from the principles. • Discover more scenarios where use of Design Patterns will be beneficial, and be able to apply specific Design Patterns correctly. Proper use of concurrency control mechanism. • Design and Implement a Framework.
Competencies	After completing the course, you should have acquired the competences to: • Participate in development of larger, modern software systems, involving a broad range of the above methodologies and practices. • Develop Object-Oriented code of high quality.

Course: IT Security	10 ECTS
The purpose of this elective area is to develop the student's competencies in making a qualified choices of It-security rules	

Course Content

- Network security
- Cryptography
- Man-in-the-middle attack
- Password cracking
- VPN
- Hacking and counter attacks
- Database security
- Penetration attack.

Examination form / Assessment

There is compulsory study activity in this course. The first 10-11 weeks of the semester are general classes. For the last approx. 5 weeks, the student prepares an individual project in a self-selected topic related to the topic. The project work is used for the exam.

Learning Outcome

Knowledge	Upon completion of the course, students should have acquired knowledge about theory, principles, and practice of: • Symmetric and asymmetric encryption and their advantages and limitations • Network security in general • Digital certificates and the management of these • The techniques used by hackers • Set up security for web-services, web-tokens (JWT) • IPSec protocol and its use • Secure Virtual Private Network (VPN) and the setup of a VPN • Blockchain principles and setup of e-coins • Penetration tests applied in Metasploit
Skills	Upon completion of the course, students should have acquired the skills to: • Apply various security tools • Defend hacking in integrated environment • Perform password cracking by brutal force • Wireless attacks applied by Cain&Abel
Competencies	Upon completion of the course, students should have acquired the competencies to: • Set up security rules in a company • Install IT-Security tools • Setup a corporate VPN

Course: Machine Learning | 10 ECTS

This course will focus on understanding the fundamentals of ML.

Course Content

- Machine Learning Basic Methods, Data mining
- Training models
- Learning and evaluation Principles

- Python implementations in Jupyter
- Frameworks SciKit Learn

Examination form / Assessment

There is compulsory study activity in this course. The first 10-11 weeks of the semester are general classes. For the last approx. 5 weeks, the student prepares an individual project in a self-selected topic related to the topic. The project work is used for the exam.

Learning Outcome

Knowledge	Upon completion of the course, students should have acquired knowledge about theory, principles, and practice of:
	- Applications of Machine Learning - Machine Learning Principles: supervised unsupervised, model/instance based - Basic Mathematics supporting the training models - Simple Training Models: linear regression, logistic regression, clustering, re-inforcement learning - Classification and regression - Artificial Neural Network and deep Learning - Course may address popular algorithms such as Decision Trees and Support Vector Machines - Machine Learning -up to date- frameworks: Anaconda, Jupyter, Spyder, (Azure&Google) - Libraries: sklearn, pandas, numpy, matplotlib, keras
Skills	Upon completion of the course, students should have acquired the skills to: - Apply various Machine Learning tools and models - Explain Machine Learning Systems - Building and implementation of Machine Learning applications
Competencies	Upon completion of the course, students should have acquired the competencies to: - Apply Machine Learning in a company - Use Machine Learning in specific apps. - Setup a corporate system