Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Semester 1

<table>
<thead>
<tr>
<th>Module 1.1</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Algebraic Structures</td>
<td>30</td>
<td>15</td>
<td>3</td>
</tr>
<tr>
<td>Foundations of Computing</td>
<td>30</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 1.2</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication Theory</td>
<td>30</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Networking</td>
<td>30</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 1.3</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distributed Systems</td>
<td>30</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 1.4</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intelligent Systems - Agents and Reasoning</td>
<td>30</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Intelligent Systems - Information Retrieval and Learning</td>
<td>30</td>
<td></td>
<td>3</td>
</tr>
<tr>
<td>Intelligent Systems - Problem Solving</td>
<td>20</td>
<td>10</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 1.5</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information Security Basics</td>
<td>30</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Module 1.6</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principles of Software Development</td>
<td>30</td>
<td>15</td>
<td>3</td>
</tr>
</tbody>
</table>

Semester 2

<table>
<thead>
<tr>
<th>Module 2.1</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithmic Decision Theory (Optional)</td>
<td>45</td>
<td></td>
<td>5</td>
</tr>
</tbody>
</table>
### Module 2.2
Algorithms for Numbers and Public-Key Cryptography (Optional)  
Lecture: 45 ECTS: 5

### Module 2.3
Dependable Systems (Optional)  
Lecture: 45 ECTS: 5

### Module 2.4
Formal Methods (Optional)  
Lecture: 45 ECTS: 5

### Module 2.5
Big Data Analytics (Optional)  
Lecture: 30 Exercise: 30 ECTS: 5

### Module 2.6
Information Theory and Coding (Optional)  
Lecture: 45 ECTS: 5

### Module 2.7
Intelligent Agents I (Optional)  
Lecture: 45 Exercise: 14 ECTS: 5

### Module 2.8
Knowledge Discovery and Data Mining (Optional)  
Lecture: 30 Exercise: 15 ECTS: 5

### Module 2.10
Networked Feedback Systems (Optional)  
Lecture: 30 Exercise: 15 ECTS: 5

### Module 2.11
Optimisation for Computer Science (Optional)  
Lecture: 45 ECTS: 5

### Module 2.12
Principles of Security Engineering (Optional)  
Lecture: 45 ECTS: 5

### Module 2.13
Quality of Service in Computer Networks (Optional)  
Lecture: 45 ECTS: 5

### Module 2.14
Symmetric Key Cryptography and Security of Communications (Optional)  
Lecture: 45 ECTS: 5
### Module 2.15
Introduction to Static Program Analysis (Optional)
- **Lecture (UE):** 45
- **Exercise (UE):** 0
- **ECTS:** 5

### Module 2.16
Software Vulnerabilities: Exploitation and Mitigation (Optional)
- **Lecture (UE):** 45
- **Exercise (UE):** 0
- **ECTS:** 5

### Semester 3

#### Module 3.1
Intellectual Property
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 3

#### Module 3.2
Advanced Project Management
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 3

#### Module 3.3
Management of Information Security (Optional)
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 4

#### Module 3.4
Computational Statistics (Optional)
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 4

#### Module 3.5
Cryptocurrencies and the Cryptographic Blockchain (Optional)
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 4

#### Module 3.7
Advanced Database Topics (Optional)
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 4

#### Module 3.8
Autonomous Robot Software (Optional)
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 4

#### Module 3.9
Intelligent Agents II (Optional)
- **Lecture (UE):** 30
- **Exercise (UE):** 0
- **ECTS:** 4
<table>
<thead>
<tr>
<th>Module</th>
<th>Lecture (UE)</th>
<th>Exercise (UE)</th>
<th>ECTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.10</td>
<td>Machine Learning (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>3.11</td>
<td>Management of Information Security (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>3.12</td>
<td>Model-Driven Software Development (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>3.13</td>
<td>Selected Topics in Network and System Security (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>3.14</td>
<td>Open Network Security (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>3.15</td>
<td>Parallel and Grid Computing (Optional)</td>
<td>16</td>
<td>30</td>
</tr>
<tr>
<td>3.16</td>
<td>Security Modelling (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>3.17</td>
<td>Security Protocols (Optional)</td>
<td>24</td>
<td>6</td>
</tr>
<tr>
<td>3.18</td>
<td>Selected topics in Artificial Intelligence (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>3.19</td>
<td>Software Engineering Environments (Optional)</td>
<td>10</td>
<td>20</td>
</tr>
<tr>
<td>3.21</td>
<td>Testing and Validation (Optional)</td>
<td>26</td>
<td>4</td>
</tr>
<tr>
<td>3.22</td>
<td>Ubiquitous Computing (Optional)</td>
<td>30</td>
<td>4</td>
</tr>
<tr>
<td>Module</td>
<td>Lecture (UE)</td>
<td>Exercise (UE)</td>
<td>ECTS</td>
</tr>
<tr>
<td>-------------------------</td>
<td>--------------</td>
<td>---------------</td>
<td>------</td>
</tr>
<tr>
<td>Fault and Intrusion Tolerance (Optional)</td>
<td>30</td>
<td>22</td>
<td>4</td>
</tr>
<tr>
<td>Master thesis</td>
<td>30</td>
<td></td>
<td>30</td>
</tr>
</tbody>
</table>

Semester 4
Semester 1

### Basic Algebraic Structures

<table>
<thead>
<tr>
<th>Module:</th>
<th>Module 1.1 (Semester 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECTS:</td>
<td>3</td>
</tr>
<tr>
<td>Objective:</td>
<td>The course supplies those basic notions of algebra which are necessary for following higher courses in the frame of the Master MICS</td>
</tr>
<tr>
<td>Course learning outcomes:</td>
<td>* Explain basic algebraic structures such as groups and fields</td>
</tr>
<tr>
<td></td>
<td>* Present proofs of properties of algebraic structures</td>
</tr>
<tr>
<td>Description:</td>
<td>1. Groups</td>
</tr>
<tr>
<td></td>
<td>2. Rings and Fields</td>
</tr>
<tr>
<td></td>
<td>3. An application: Public key encryption</td>
</tr>
<tr>
<td></td>
<td>4. Coset spaces and quotient groups</td>
</tr>
<tr>
<td></td>
<td>5. Polynomial rings over fields</td>
</tr>
<tr>
<td></td>
<td>6. Construction of field extensions.</td>
</tr>
<tr>
<td></td>
<td>7. Finite fields</td>
</tr>
<tr>
<td></td>
<td>8. Applications to Coding Theory (if time permits)</td>
</tr>
<tr>
<td>Teaching modality:</td>
<td>The course consists of lectures and exercises.</td>
</tr>
<tr>
<td></td>
<td>The exercise will take place every second week and are related to the topics of the course.</td>
</tr>
<tr>
<td>Language:</td>
<td>Anglais</td>
</tr>
<tr>
<td>Mandatory:</td>
<td>Oui</td>
</tr>
<tr>
<td>Evaluation:</td>
<td>There will be a final written exam which will count for 100 percent. The exam will cover the lectures and the exercises</td>
</tr>
<tr>
<td>Remark:</td>
<td>(1) The individual work on the exercises is highly recommended.</td>
</tr>
<tr>
<td></td>
<td>(2) There is a typeset manuscript available for the lecture.</td>
</tr>
<tr>
<td>Professor:</td>
<td>SCHLICHENMAIER Martin</td>
</tr>
</tbody>
</table>

### Foundations of Computing

<table>
<thead>
<tr>
<th>Module:</th>
<th>Module 1.1 (Semester 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECTS:</td>
<td>3</td>
</tr>
<tr>
<td>Objective:</td>
<td>Basic knowledge of computability and computational complexity theory.</td>
</tr>
<tr>
<td>Course learning outcomes:</td>
<td>* List various computational models and their properties</td>
</tr>
<tr>
<td></td>
<td>* Analyze and compare the complexity of problems</td>
</tr>
<tr>
<td>Description:</td>
<td>Mathematical Background/Definitions</td>
</tr>
</tbody>
</table>
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

- Problems/Solving Problems
- Computational Models
- Algorithms/Procedures

Computability Theory
- Unlimited Register Machine (URM)
- URM-Computability
- Partial /Primitive Recursive Primitive Functions
- Computable Functions

Other approaches to Computability
- Turing Computability (Turing Machines (TM) /Universal TM)
- Other Model's Computability (e.g., Lambda Calculus)
- Church-Turing's Thesis

Computational Complexity Theory
- Time Complexity: the classes P and NP; NP-Completeness
- Space complexity: Savitch's theorem, PSPACE and PSPACE-completeness
- Intractability: hierarchy theorems; exponential space completeness

Teaching modality: The course is organized as a series of 12 lectures.
Language: Anglais
Mandatory: Oui
Evaluation: 100% final exam

Communication Theory

Module: Module 1.2 (Semester 1)
ECTS: 3
Objective: Provide mathematical fundamentals of the physical layer like stochastic signals and systems, ML and MAP principle, modulation, and channel models.
Course learning outcomes:
* Describe fundamental parameters of signals, systems, and channels
* Take optimal stochastic decisions based on observations
Description:
* Signals and Systems
* Convolution
* Sampling
* Stochastic Signals and Noise
* Modulation and Demodulation
* The Maximum Likelihood Principle
* Sources and Channels
Teaching modality:
* Review of Stochastic Signals and Systems
* Digital Transmission and Modulation
* Demodulation
* Channel Models
Networking

Module: Module 1.2 (Semester 1)
ECTS: 3
Objective: Introduce higher networking layers and mathematical descriptions of network concepts as Multiple Access Control (ALOHA, collision detection and resolution), Error detecting and correcting codes, ARQ, routing and flow control, Queueing and QoS.
Description: * Hierarchical Model of Network Functions (OSI Model, Service Access Points)
  * Point-to-Point Data Transmission (synchronous and asynchronous multiplexing, packets)
  * Error correcting and detecting codes, ARQ protocols
  * Multiple Access Control (ALOHA, Slotted ALOHA, collision resolution, detection and avoiding)
  * Routing and flow control
  * Introduction to Queueing Theory
  * Mobile Network Access Schemes
  * Quality of Service Parameters in TCP/IP
Teaching modality: The course consists of a series of lectures with dedicated time slots for exercises
Language: Anglais
Mandatory: Oui
Evaluation: There is a final exam counting 70%. Successful preparation, submission and participation in exercises is valued 30%
Professor: ENGEL Thomas

Distributed Systems

Module: Module 1.3 (Semester 1)
ECTS: 3
Objective: The objective of the course is to provide an overview of the area of distributed systems with a focus on distributed algorithms. After successful completion of this course the student should be able to:
  * Identify and explain the following concepts related to distributed systems: network topology, communication models, operation mode, failure model, scalability, complexity
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

- Explain the role and importance of distribution in a selection of related domains, e.g., mutual exclusion, fault tolerance Can list the relevant assumptions and requirements for each of these domains
- Understands the operation of the basic algorithms in each of these domains
- Can reason about the basic properties of these algorithms

Course learning outcomes:
* Identify and explain the following concepts related to distributed systems: network topology, communication models, operation mode, failure model, scalability, complexity
* Explain the role and importance of distribution in a selection of domains: mutual exclusion, self-stabilizing systems, distributed snapshot, termination detection, leader election, consensus, fault tolerance
* List the relevant assumptions and requirements for each of these domains
* Explain the operation of the basic algorithms in each of these domains
* Analyze the basic properties of these algorithms

Description: distributed systems. In details, it contains the following topics:
- Mutual exclusion
- Self-stabilizing systems
- Distributed snapshot
- Termination detection
- Leader election
- Consensus in distributed systems
- Fault tolerance
- Graph Algorithms

Teaching modality: The course consists of a series of lectures with dedicated time slots for exercises

Language: Anglais

Mandatory: Oui

Evaluation: We revise the numbers slightly: A final written exam will be counted for 70%, 30% being dedicated to the assignments.

Professor: MAUW Sjouke, PANG Jun

Intelligent Systems - Agents and Reasoning

Module: Module 1.4 (Semester 1)

ECTS: 3

Objective: Preparing the student for the emerging age of ubiquitous intelligent systems and robots, getting a solid background for studies in intelligent and/or adaptive systems, promoting the use of intelligent techniques in other areas of computer science, promoting the use of intelligent techniques in cross-disciplinary interaction.

Course learning outcomes:
- Explain the central role of the logical formalisms in knowledge based systems
- Give a solid formal background for studies in intelligent and/or adaptive systems
- Define and apply the basic concepts of classical propositional and first order logic, notably their semantics and proof calculi
- Explain the importance of logical reasoning for knowledge- and agent-based systems
- Explain how to use logical formalisms to model reasoning problems
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

- Model intelligent systems using classical logic
- Develop own solutions for some real modelling and reasoning problems

Description:
- Lesson 1: Syntax and Semantics of Propositional Logic;
- Lesson 2: Sequent Calculus for Propositional Logic;
- Lesson 3: Search Procedures and Completeness for Propositional Logic;
- Lesson 4: Practice with exercises;
- Lesson 5: Syntax and Semantics of First Order Logic;
- Lesson 6: Sequent Calculus for First Order Logic;
- Lesson 7: Practice with exercises;
- Lesson 8: From natural Language to First Order Logic;
- Lessons 9-10: IDP;
- Lessons 11-12: project meetings;
- Lesson 13: project presentation.

Teaching modality: Course will be based on 10 lectures, an intermediate exam, a final exam, and weekly homework exercises

Language: Anglais

Mandatory: Oui

Evaluation:

If the student takes the final test in a later session, the final mark will be fully determined by such a test, and previously submitted homework assignments and projects will not be considered.

Remark: Literature:

Handouts will be distributed during the course

Professor: VAN DER TORRE Leon, STEEN Alexander

Intelligent Systems - Information Retrieval and Learning

Module: Module 1.4 (Semester 1)

ECTS: 3

Objective: In general, the field of Information Retrieval denotes the process of delivering answers (for example: link recommendations) to a requested search as well as its presentation in an user-friendly way. However, this kind of information presentation is not that easy as it seems, since a variety of different components of different research fields come into play. In particular, the presented answers should be as precise as possible and should satisfy the enquirer's request. But what, if the enquirer's terms are misspelled or ambiguous? What, if the enquirer uses wildcards? How can we guarantee that the best answers appear on top of the answer set? How can the search engine learn from the input it receives?

Course learning outcomes: The student should be able to understand the basic concepts of Information Retrieval, which concern the structure of a search engine, the processing of queries, the ranking and the evaluation of search results, the role of Feedback, and others, as well as a number of learning mechanisms in view of further applications like the classification of texts as a bridge to Knowledge Discovery (Semester 2) and Machine Learning (Semester 3).
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

**Description:**
In the course, we discuss selected chapters of the book "Introduction to Information Retrieval" by Christopher Manning et al., which is also given at Stanford University. The candidate learns to understand the fundamental aspects regarding information retrieval as well as its elementary concepts, particularly the components of a search engine, the role of Natural Language Processing, and evaluation strategies, query extensions, and ranking. We also foster on Google's PageRank and discuss several text classification methods.

**Teaching modality:**
Lecture and Exercises

**Language:**
Anglais

**Mandatory:**
Oui

**Evaluation:**
100% Final Examination

**Remark:**
REFERENCES


**Professor:**
SCHOMMER Christoph

---

**Intelligent Systems - Problem Solving**

**Module:**
Module 1.4 (Semester 1)

**ECTS:**
3

**Objective:**
The objective of this lecture first consists in providing a structured approach to students in terms of optimization problem modeling. Next various solving techniques based on exact methods (A*, B&B, LP), approximated ones (heuristics, meta-heuristics, problem relaxation) and hybrids are described. Students are also taught how to validate the proposed solution by having a scientific approach.

**Description:**
Introduction to optimisation and decision problems
Linear programming, graphical interpretation and primal simplex
Branch&Bound, A*
The scheduling problem
List algorithms, greedies, heuristics
Meta-Heuristics and Evolutionary computation

**Teaching modality:**
The course is organized as a series of theoretical lectures intermixed with hands-on exercises. A computer lab is also organised at the end of the lecture

**Language:**
Anglais

**Mandatory:**
Oui

**Evaluation:**
Final Exam: 100%

**Professor:**
BOUVRY Pascal
Information Security Basics

Module: Module 1.5 (Semester 1)
ECTS: 3
Objective: The objective of this course it to provide an introduction to information security
Course learning outcomes:
* explain the role of security protocols in the design of secure systems;
* use the standard building blocks for security protocols: nonces, symmetric and asymmetric encryption, hash functions;
* compare symmetric vs public key cryptography
* classify attacks by attack scenarios and by attack goals
* describe weaknesses of historical ciphers: substitution, transposition, WWII ciphers
* describe basic modes of operation of block ciphers: ECB, CBC, Counter mode
* describe the RSA public-key encryption scheme and the RSA signature scheme

Description: The goal of this course is to provide basic background in Cryptography and IT security and to show what expertise in these areas is available in the laboratory of Algorithms, Cryptography and Security (LACS) and CSC. The topics cover symmetric and public key cryptography and security of protocols
Teaching modality: The course is based on lectures
Language: Anglais
Mandatory: Oui
Evaluation: 50%: project
50%: homework
Professor: RYAN Peter Y A, BIRYUKOV Alexei, CORON Jean-Sébastien

Principles of Software Development

Module: Module 1.6 (Semester 1)
ECTS: 3
Objective: The objective of the course is to introduce some principles of software development through the presentation of the Android ecosystem. After successful completion of this course the student should be able to:
- understand the main principles on which Java is based on;
- efficiently use software repositories or source code management systems and make the difference between git, svn or cvs
- understand the notion of design patterns
- implement an Android application and understand the concept of Inter-Component Communications.
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Course learning outcomes:
* design a reusable and evolvable software
* successfully conduct a software project
* leverage APIs

Description:
Sessions on Java
Session on Design Pattern
Session on Software development lifecycle
Session on source code management systems
Session on Introduction to Android
Sessions on Android Interface and Communication (Layout, Android Activity and introduction to the concept of Intent)
Sessions on Android Data and Services (Services, Broadcast receiver, and Content Providers)
Session for the preparation of the final project
Sessions of practical

Teaching modality: The lessons are combining (formal) presentations and practical exercises.
Each student will write code! Bring your laptop.

Language: Anglais
Mandatory: Oui
Evaluation: A final written exam will account for 50% of the grade.
The remaining 50% are based on a project evaluation focusing on the implementation of an Android Application.

Professor: KLEIN Jacques, BISSYANDE Tegawendé François d'Assise
Algorithmic Decision Theory

Module: Module 2.1 (Semester 2)
ECTS: 5
Objective: The objective of this course is to introduce students to ADT, a new interdisciplinary field at the intersection of decision theory, discrete mathematics, theoretical computer science and artificial intelligence. ADT proposes new ideas, approaches and tools for supporting decision making processes in presence of massive databases, combinatorial structures, partial and/or uncertain information, and distributed, possibly inter-operating, decision makers. Such problems arise in several real-world decision making problems such as humanitarian logistics, epidemiology, risk assessment and management, e-government, electronic commerce, and the implementation of recommender systems.

Course learning outcomes:
* Recognise and formulate problems that relate to Algorithmic Decision Theory (ADT)
* Identify the operational complexity issues arising in ADT
* Adapt some of the classical Operational Research and Decision Aid solving strategies to the ADT context
* Implement an ADT solver for selected case studies

Description: Varying with the main focus and the interest of the students, the content of the lectures may concern:
1) General introduction to algorithmic decision theory
2) Who wins the election ? An introduction to social choice theory
3) How to rank the candidates ? Main ranking rules
4) Building and aggregating performance measures
5) The American way: Multiple attributes value theory MAVT
6) The European way: Multiple criteria based decision aid MCDA
7) Best choice recommendation algorithms
8) Sorting and clustering algorithms
9) Inverse Decision Analysis

Teaching modality: The course is organized as a series of theoretical lectures with hands-on exercises intermixed
Language: Anglais
Mandatory: Non
Evaluation: The students may choose either to elaborate a project: 100% or to write a final exam: 100%
Remark: The following textbooks will be used in this course:
- Aiding decisions with multiple criteria, D. Bouyssou, E. Jacquet-Lagrèze et al., Kluwer 2002
- Evaluation and decision models with multiple criteria: Stepping stones for the analyst, D. Bouyssou, Th. Marchant et al. Springer 2006

Professor: BISDORFF Raymond Joseph
Algorithms for Numbers and Public-Key Cryptography

**Module:** Module 2.2 (Semester 2)  
**ECTS:** 5  
**Objective:** The objective of this course is to provide an introduction to algorithms for numbers and their use in public-key cryptography.  
**Course learning outcomes:**  
- describe the basic algorithms for numbers: gcd, CRT, modular exponentiation, primality tests.  
- list some basic properties of numbers: modular computation, Euler function, generators of multiplicative groups.  
- explain the RSA algorithm for public-key encryption and signature.  
- explain basic security proofs for public-key encryption and signature.  
**Description:**  
- basic basic algorithms for numbers: gcd, CRT, modular exponentiation, primality tests, etc.  
- the RSA algorithm for public-key encryption and signature.  
- main security notions for encryption and signature.  
- basic security proofs for public-key encryption and signature.  
**Teaching modality:** The course is based on lectures and practical sessions  
**Language:** Anglais  
**Mandatory:** Non  
**Evaluation:** The final grade is based on homework only.  
**Professor:** CORON Jean-Sébastien

Dependable Systems

**Module:** Module 2.3 (Semester 2)  
**ECTS:** 5  
**Objective:** The course will introduce, using a software engineering perspective, the fundamentals of dependable systems: standards for dependability, fundamental concepts of dependability, engineering activities and dependability attributes and means, fault prevention, tolerance, removal and forecasting. Some reusable dependability mechanisms will be studied conceptually and implementation means to support dependable systems development will be presented and used.  
A dependable system development project will be made and used for applying the concepts learned.  
**Course learning outcomes:**  
**Description:**  
1. Student/Teachers presentation  
Lecture general rules - Introduction to Dependability
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

2. Critical Systems
   Software Fault tolerance - big picture
3. Terminology
   Dependability evaluation and assessment - Fault-tree modelling and analysis
4. Continuation
5. FT Basic Concepts Redundancy techniques and Reliability evaluation
   Recovery Block
6. Measures: MTTF, failure rate, reliability, MTTR, MTBF, and availability
   Fault-tree analysis and reliability block diagram - Redundancy
7. Exception Handling - N-Version Programming
8. Exception handling in Java
   Backward and Forward recovery
9. Architectural issues in Software Fault-Tolerance
   Analysis of Software Fault tolerance
10. N-version programming
    Decision algorithms
    Fail fast component
11. Dependability Modeling
    Mini Project: dependability analysis (DRET)
12. Mini Project: modelling (Coordinated Atomic Actions)
13. Mini Project: implementation (CAA-DRIP framework)

Teaching modality: The course is organized as a series of theoretical lectures intermixed with hands-on exercises and practical work on computers

Language: Anglais

Mandatory: Non

Evaluation: continuous evaluation: 1/6
project deliverables: 2/3
project presentations: 1/6

Professor: NAVET Nicolas, HU Tingting

---

**Formal Methods**

Module: Module 2.4 (Semester 2)

ECTS: 5

Objective: In this course, we will study a collection of techniques that are essential in the construction of large and highly reliable systems

Course learning outcomes:
* Explain the informal meaning of the basic process algebra operators
* Relate this informal interpretation to its formalization through axioms and transition rules
* Specify the behaviour of simple concurrent systems using process algebra
* Explain the meaning of a temporal logic formula
* Specify system requirements using the studied logics
* Evaluate the logic formulas on labelled transition systems
* Explain the fundamental concepts in model-driven software development such as model, meta-model, modeling language, abstract syntax and semantics
* Define a simple modeling language by giving its syntax and semantics
* Evaluate and compare current model-driven technologies

**Description:**
The course will focus on the principles and applications of the following formal methods:

* **Alloy** - a declarative modeling language based on first-order logic, for expressing complex structural constraints and behavior. The Alloy Analyzer is a constraint solver that provides fully automatic simulation and checking. The motivation for the Alloy project was to bring to Z-style specifications (Z being another formal language) the kind of automation offered by model checkers.

* **Process Algebras** - Process Algebra is a formal description technique for complex computer systems, especially those involving communicating, concurrently executing components. It is a subject that touches many topic areas of computer science and discrete math.

* **Protocol Verification** - a specification language based on process algebra extended with abstract data types is used to describe communicating systems. Interesting properties about the systems are expressed as temporal logic formulas. Model checking is employed as a method to formally verify whether the model defined by the system satisfies the properties.

We have chosen these methods because they are based on complementary approaches for formal reasoning about systems. The course is tool-oriented in the sense that automatic tool support will be an essential part of the teaching units whenever possible. In this sense the lecture provides a more hands-on approach to theoretical concepts compared to courses in the first semester. Many examples and real-life case studies drawn from a variety of domains such as computer security, communicative systems and software engineering will be presented.

**Teaching modality:**
The course is organized as a series of theoretical lectures with hands-on exercises intermixed

**Language:**
Anglais

**Mandatory:**
Non

**Evaluation:**
Final exam: 70%
Assignments: 30%

**Professor:**
KELSEN Pierre, MAUW Sjouke, PANG Jun

---

**Big Data Analytics**

**Module:**
Module 2.5 (Semester 2)

**ECTS:**
5

**Objective:**
The lecture provides an entry point to large-scale data management and distributed computing principles in recent NoSQL architectures. We start with an overview of distributed file systems and MapReduce in Apache Hadoop and then move on to more advanced analytical tasks based on the machine-learning libraries in Apache Spark. The lecture serves as an ideal basis for further topics in this area (such as Master seminars, projects and theses).

**Course learning outcomes:**
- Students become familiar with the usage of recent Big Data platforms such as Apache Hadoop and Spark
- Student obtain an overview of both the theoretical foundations and practical applications of various Big Data and Machine Learning algorithms
- Students learn how to approach and solve different data-analysis tasks by a number of programming exercises with real-world datasets
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Description: The course consists of a combination of theory-oriented lectures and practical exercises, through which the students are guided by a series of real-world use cases and hands-on examples. Specifically, we focus on the following topics:
- Distributed File Systems (DFS) and MapReduce in Apache Hadoop
- Resilient Distributed Data (RDD) objects and DataFrames in Apache Spark
- Implementation of complex DataFlow programs in Spark using Scala
- Performing advanced analytical tasks in Spark's MLib:
  o Distributed clustering and classification of objects
  o Decision trees and random forests
  o Recommender systems via matrix factorization
  o Text analysis via latent semantic indexing
  o Geospatial data analysis
  o Social-network analysis

Teaching modality: The course offers both theory lectures and practical exercise sessions. The lectures serve as theoretical basis for the algorithmic concepts which we then apply during the practical sessions. The solutions to the exercises are developed and demonstrated interactively with the tutors.

Language: Anglais

Mandatory: Non

Evaluation: Practical exercises: 50%
Final written exam: 50%

Professor: THEOBALD Martin

---

Information Theory and Coding

Module: Module 2.6 (Semester 2)
ECTS: 5

Objective: The objective of this course is to provide an understanding of fundamental communication limits and means of approaching them

Course learning outcomes:
* Compute fundamental communication limits
* Compress simple information sources
* Describe the fundamental blocks of digital communication systems (physical layer)
* Encode binary information with a convolutional code

Description: The course contains:
- Shannon's concept of mathematically quantising information and uncertainty for a communication setup
- Explanations that both compression and error free transmission have an extremal rate which can be computed via entropy and mutual information
- Methods to compress sources
- Digital transmission techniques and their complexity for inter-symbol-interference channels
- Simple error correction codes, convolutional codes

Teaching modality: The course is organized as a series of theoretical lectures intermixed with exercises/homework.
Intelligent Agents I

Module: Module 2.7 (Semester 2)
ECTS: 5
Objective: The objective of this course is to introduce students to knowledge representation and reasoning methods for intelligent agent systems.

Course learning outcomes:
* Explain the nature and roles of different formal theories used for individual reasoning and autonomous agents, such as various modal logics, belief change formalisms, or methods for uncertainty management
* Define and apply the basic concepts of one or two non-classical logics (e.g. modal logic and default logics), notably their semantics and proof calculi
* Model intelligent systems using non-classical logics
* Explain the philosophical foundations of individual reasoning.

Description: The course has 4 parts:
1. Modal logics for agent reasoning
2. Conditional logic
3. Natural language semantics & non-monotonic logic
4. Formal argumentation

Language: Anglais
Mandatory: Non
Evaluation: Final exam 65%; Homework 35%
Professor: SORGER Ulrich

Knowledge Discovery and Data Mining

Module: Module 2.8 (Semester 2)
ECTS: 5
Objective: We understand Data Mining (Knowledge Discovery) as a life-cylce process from data to information and insights. In times of Big data, Data Mining has become a central interest both for industry and academia. In this course, we discuss several data-related aspects like preprocessing or pricacy as well as selected aspects of Machine Learning. An expansive definition of Data Mining, which is the derivation of insights from masses of data by studying
and understanding the structure of the constituent data, and selected applications complete the course.

**Course learning outcomes:**

* Explain the fundamental concepts of data mining and knowledge discovery
* List the properties of data relevant for deriving interesting and useful information/observation from that.
* Explain machine learning algorithms and strategies to deploy the discovered results
* Argue the importance of domain knowledge during the data analysis with its scope and limitations

**Description:**

* Definition and Process.
* Data Mining, Data Science, and the Big Data Hype.
* Data Quality and Preprocessing
* Data Privacy and Security.
* Data and Information Visualization.
* Machine Learning for Clustering, Classification, Association Discovery, Sequential Pattern Analysis, and/or Time Series Analysis.

**Teaching modality:**

The course is organised as a lecture with integrated exercises. It follows the "Information Retrieval" course and will itself be continued in Semester 3 by a more intensive discussion about "Machine Learning". Each participant must be inscribed via Moodle. Course material will be uploaded regularly.

**Language:**

Anglais

**Mandatory:**

Non

**Evaluation:**

60% oral or written examination; 40% midterm tests

**Remark:**

Selected references:


**Professor:**

SCHOMMER Christoph

---

**Networked Feedback Systems**

**Module:**

Module 2.10 (Semester 2)

**ECTS:**

5

**Objective:**

The objective of this course is to introduce students to networked feedback structures in interconnected information and communication technology in technical environments

**Course learning outcomes:**

* Identify feedback structures, decompose them and formulate continuous and sequential dynamics
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

* Determine reliable discrete or continuous enclosures for structure-variations and uncertainties
* Design controls with guaranteed dynamic tolerances
* Design reliable automata in technical context

Description:
Introduction
- networked feedback and feedforward
- sampling, scheduling and communication
- continuous system representations
- dynamics and approximations
- systems over the binary field
- binary transfer function and stability
- combined systems and decompositions
- feedback design in multiloop structures

Teaching modality: The course is organized as a series of lectures with practical simulation exercises

Language: Anglais

Mandatory: Non

Evaluation:
Final Exam: 70%
Project: 30%

Optimisation for Computer Science

Module: Module 2.11 (Semester 2)

ECTS: 5

Objective: Different problems have different nature. In terms of complexity some problems are called intractable and cannot be solved by classical computers. But there are also many other aspects of the nature of optimisation problems such as linearity, convexity, continuity, dynamicity, randomness that may lead the choice of different optimisation techniques

Course learning outcomes:
* Characterize problems
* Identify the key concepts related to optimisation techniques
* Use optimization frameworks
* Implement optimization algorithms
* Validate optimization algorithms and results
* Validate approaches for solving optimization problems

Description: This lecture confronts the students to real instances of such problems. They are first asked to model the problem and next proposed solutions include exact methods, relaxations, approximations, heuristics and meta-heuristics. And these practical study cases are supported by the theoretical lectures on Problem Solving (1st semester)

Teaching modality: The students are directly involved into research teams helping them to solve real problems illustrating the various approaches.
A mid term review is organized during which students present the problem model and at the end of the course, students will present the final results

Language: Anglais
Principles of Security Engineering

Module: Module 2.12 (Semester 2)
ECTS: 5
Objective: This course aims at teaching students the fundamental principles of engineering secure systems. More specifically, the goals are:
- to give a broad understanding of how secure systems are designed and evaluated.
- to explain key security concepts and mechanisms as well as pitfalls.
- to describe how systems are attacked and defended: typical threats, vulnerabilities and countermeasures.
- to take a "system-based" approach, i.e. to take account of the whole system rather than just the technical, e.g. crypto algorithms and protocols.

Course learning outcomes:
* Evaluate security systems and identify their vulnerabilities
* Propose countermeasures to vulnerabilities and attacks
* Evaluate security requirements

Description:
Intro (concepts, principles)
Policies and models (access control etc)
Information flow (enforceable policies)
Socio-technical aspects.
Physical security, locks, tamper resistance/evidence
Copyright, DRM, watermarking.
Privacy (Jonker)
Network security (malware, phishing, botnets…)
Security evaluation and testing
Advanced Protocols (PAKE, QKD, ZK, OT, …)
Misc topics (API, TCM, attack trees, game theory, ECC, MDD,…)
Secure voting systems

Teaching modality: lectures plus the mid-term exam

Language: Anglais
Mandatory: Non
Evaluation: The course will be evaluated based on a report (50%) and presentation (50%) at the end of the course
Professor: RYAN Peter Y A
Quality of Service in Computer Networks

Module: Module 2.13 (Semester 2)
ECTS: 5
Objective: The objective of this course is to introduce quantitative measures for network performance (like throughput, error correction, delays, routing) for different network topologies to be applied to security protocols. It also sensibilises for differences between static and dynamic networks as well as centralised and de-centralised topologies concerning reliability and trust issues.

Course learning outcomes:
* Describe performance metrics and list parameters of dedicated networks and protocols.
* Name and reproduce definitions of relevant parameters that theoretically characterise the communication traffic incl. queues, routing and error probabilities.
* Analyze existing solutions according to their capabilities for throughput, error rate and security.
* Construct and adapt real world communication architectures and protocols with given Quality of Service requirements on the basis of the theoretical concepts.

Description:
1. Intro
2. Recap: Random Processes
3. Recap: Homogeneous Markov Chains
4. Commutation Systems: Components and modules
5. Communication Traffic as Random Process
6. Routing and Flow Control
7. Introduction to Queueing Theory
8. QoS in TCP/IP

Teaching modality: The course is organised as a series of lectures with dedicated time slots for exercises.

Language: Anglais

Mandatory: Non

Evaluation: 70% Final Exam
30% Successful preparation, submission and active participation in exercise sessions.

Professor: ENGEL Thomas

Symmetric Key Cryptography and Security of Communications

Module: Module 2.14 (Semester 2)
ECTS: 5
Objective: Introduction to symmetric cryptography and applied cryptography: the students will learn design and analysis principles for symmetric crypto primitives (ciphers, hash functions, MACs). They will be also introduced to the aspects of practical application of cryptography.
Course learning outcomes:
* apply differential power analysis (DPA) to smartcard implementations of ciphers
* explain hardware and software implementation issues for cryptographic primitives and security protocols
* evaluate affects of cryptanalysis, side-channel attacks and traffic analysis on cryptographic primitives

Description:
The goal of the first half of this course is to introduce students to symmetric key cryptography, showing how ciphers and hash functions are designed and cryptanalyzed. We will start with some historical examples (M-209, Enigma) and then follow to the present day standards (DES, AES, KASUMI, SHA, GSM-A5/1, RC4, Bluetooth-E0, SNOW3G). In the second half of the course we will discuss broader applied cryptography and network security topics such as side-channel attacks, crypto-hacking, anonymity networks and digital currencies.

Teaching modality:
The course is organized as a series of theoretical lectures accompanied with practical exercise sessions

Language:
Anglais

Mandatory:
Non

Evaluation:
The grade for this class will be an average of the homework assignments given every week. Assignments are to be solved individually. There will be no final exam

Professor:
BIRYUKOV Alexei

Introduction to Static Program Analysis

Module: Module 2.15 (Semester 2)

ECTS: 5

Objective: Through this course the student will learn the fundamental theoretical concepts and techniques of static analysis.
The student will be able to use this knowledge to implement static analyses to solve concrete problems.

Course learning outcomes:
* The student should be able to critically read publications related to static analysis (research paper, etc.)
* The student should be able to select an adapted approach to solve a specific static analysis problem
* The student should be able to implement static analysis techniques

Description: Static analyses are used in various situations, from compiler code optimization to security analysis of Android applications.
This course provides the concepts and techniques underlying static program analysis.
Topics include forward/backward data-flow analysis, inter-procedural analysis, pointer analysis and call graph construction. A particular focus will be given to recent and advanced techniques such as Android bytecode static analysis for security.
The course will mix theory and practice. Students will implement simple analyses and complete a course project.

Teaching modality:
The course is organized as theoretical lectures intermixed with labs and also includes a final project.
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Language: Anglais
Mandatory: Non
Evaluation:
- 50% assignments and project
- 20% readings and exercises
- 30% exam(s)
Remark: Maybe needed textbooks
Professor: LE TRAON Yves, BARTEL Alexandre

Software Vulnerabilities: Exploitation and Mitigation

Module: Module 2.4 (Semester 2)
ECTS: 5
Objective: Through this course students will understand software vulnerabilities from memory corruptions to command injections. Both the defensive and offensive aspects will be studied: students will learn how to mitigate, find and exploit software vulnerabilities.

Course learning outcomes:
* Students should be able to critically read publications related to software vulnerabilities (research paper, etc.)
* Students should be able to identify vulnerable code and write robust code preventing vulnerabilities from being introduced in the code.
* Students should be able to exploit simple known vulnerabilities.

Description: Our lives and our societies rely on computer programs (software). Every day, we use devices running software written in millions of lines of code because it makes our lives easier. However, the complexity and the size of existing software, added to the fact that humans write most of the software, introduce bugs. Some of these bugs, called vulnerabilities, can be exploited by an attacker to compromise a device or leak information.
Have you ever wondered how programmers make their code more robust to avoid introducing vulnerabilities?
Have you ever wondered how attackers can find vulnerabilities and exploit them to take control of a remote device on the Internet or of your smartphone?
Have you ever wondered how attacker can dump an entire database containing personal information about millions of users?
In this course, you will learn both how to defend against vulnerabilities and how to exploit vulnerabilities. This course covers memory corruption vulnerabilities such as buffer or heap overflow, type confusion, or use after free. It also covers more high level vulnerabilities such as SQL injection or confused deputy.

The course will mix theory and practice.
On the offensive side, you will implement simple programs to exploit vulnerabilities. On the defensive side, you will correct vulnerable programs to prevent exploitation but also learn how to use techniques such as fuzzing to find new vulnerabilities.

**Teaching modality:** The course is organized as theoretical lectures intermixed with labs and also includes a final project.

**Language:** Anglais

**Mandatory:** Non

**Evaluation:** Rating: 50% assignments and project
20% readings and exercises
30% final exam

**Professor:** BARTEL Alexandre, LE TRAON Yves, KLEIN Jacques
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Semester 3

### Intellectual Property

**Module:** Module 3.1 (Semester 3)

**ECTS:** 3

**Objective:** The objective of the course is to promote greater awareness of intellectual property as one of the most important assets of a company in the global market environment.

**Course learning outcomes:**
- Explain in what sense intellectual property is an asset for a company
- Differentiate between patents, trademarks and copyright
- Compare different types of licences

**Description:**
I. Intellectual property an asset of your company
   ii. Trade secrets
   iii. What is an invention?
   iv. What is a patent of invention?
   V. Invention report and record form
   Vi. From the invention to a patent
   Vii. Patent grant procedures
   Vi. Patent rights & patent infringement
   Ix. Patents as a source of information
   X. Trademarks
   XI. Registered designs & design patents
   XII. Copyright
   XIII. Licensing of ip-rights
   XIV. The intellectual property audit

**Teaching modality:** The course is organized as a series of theoretical lectures with hands-on exercises intermixed.

**Language:** Anglais

**Mandatory:** Oui

**Evaluation:** Final Exam: 70%
          Project: 30%

**Professor:** KIHN Pierre

### Management of Information Security

**Module:** Module 3.3 (Semester 3)

**ECTS:** 4
Objective: The objective of this course is to prepare students to start a practical experience as Information Security Officer and to assume such responsibility in his future professional activities. The student will understand that her/his work consists mainly in the planning, coaching, designing of solutions and convincing both senior management and information users on the needs and effectiveness of security

Course learning outcomes:
* List typical tasks of a information security manager.
* Explain the standard approach to managing security (ISO 27001).
* Explain basics of information risk management (ISO 27005), like asset valuation, impact and occurrence assessment, risk treatment plans, risk acceptance.
* Apply a fast risk assessment method (by ENISA) to a small virtual organisation.
* Compare different approaches for risk assessment.
* Assess metrics to measure security (ISO 27004) and tactics to implement security (ISO 27003).
* Recognise practical aspects of an Information security policy, including incident management, business continuity management, asset classification, awareness rising.
* Propose an information security policy for a small organisation.

Description: This course presents the core requirements and guidelines of ISO standards on security management, risk assessment and risk management, security policies, and awareness raising:
• Practical aspects of information security
• Standard approach to managing security (ISO 27000, 27001, 27002, 27003, 27004, ...)
• Basics of information risk management (ISO 27005), like asset valuation, impact and occurrence assessment, risk treatment plans, risk acceptance, ...
• Applying a fast risk assessment method (e.g. ENISA) on a small virtual organisation SME and learning more detailed risk management method (e.g. TRICK-light).
• Practical aspects of an Information security policy, including incident management, business continuity management, asset classification, awareness rising

Teaching modality: The course is organized as a series of theoretical lectures with hands-on exercises and discussion of the solution of group and individual exercises intermixed

Language: Français
Mandatory: Non
Evaluation: Final Exam: 50%
Exercises: 50%
Professor: HARPES Carlo
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

* Define scope and construct the work breakdown structure of a project
* Select and employ project management standards and methods

Description:
Introduction to Project Management (review)
- Project Scope Management
- Define project scope
- Project Work Breakdown Structures (WBS)
- Project Planning
- Develop project WBS and planning
- Monitoring and Controlling Projects
- Project rescoping and replanting
- Project Closure

Teaching modality:
The course is a lecture including a project case study where students will research and development on selected subjects in advance project management methodologies and standards. We use the chalkboard, presentation slides, e-Learning support and real projects to illustrate the need of project management.

Language: Anglais

Mandatory: Oui

Evaluation:
The course examination is oral. Students will defend in small groups of 2 to 4 their content research results on the assigned Project Management process, methodology or standard. The examination takes place in winter and summer semester (re-exam).

Remark:

Professor: CORONADO Sergio

Computational Statistics

Module: Module 3.4 (Semester 3)

ECTS: 4

Objective:
The objective of this course is to introduce the students to the R language and environment for statistical computing and graphics (a GNU project). In particular, the course proposes effective data handling and storage solutions as well as useful operators for calculations on arrays, in particular matrices. A selected collection of intermediate tools for data analysis, graphical facilities for data analysis and display either on-screen or on hardcopy will be illustrated from examples of statistical analyses. Finally, the course will by the way familiarise the students with a well-developed, simple and effective programming language which includes conditionals, loops, user-defined recursive functions and input and output facilities.

Course learning outcomes:
* Recognise data analysis and graphics problems that may be efficiently solved with the help of the R resource.
* Handle efficiently large statistical data sets
* Compute statistical results for data analysis case studies
* Construct print-ready and interactive graphics for data exploration case studies
* Design and implement data analysis algorithms and graphics in the R language.
Description: Varying with the main focus and the interest of the students, the content of the lectures may concern:
1) Introduction to statistical computation tools in Python, C++ and R
2) Traps and watch, as well as best practice for generating uniform random numbers
3) Implementing various random variable generators
4) Monte Carlo simulation and integration techniques
5) Incremental quantiles estimation
6) Random graphs generators
7) Statistical diagram drawing tools

Teaching modality: The course is organized as a series of theoretical lectures with hands-on exercises intermixed

Language: Anglais

Mandatory: Non

Evaluation: The students may choose either to elaborate a project: 100% or to write a final exam: 100%

Remark: The following textbooks will be used in this course:

Professor: BISDORFF Raymond Joseph

Cryptocurrencies and the Cryptographic Blockchain

Module: Module 3.5 (Semester 3)

ECTS: 4

Objective: The blockchain is a paradigm shift for cryptographic protocols. The objective of the course is to understand the mechanism of the blockchain and its applications to cryptocurrencies such as Bitcoin and to smart contracts as in Ethereum.

Course learning outcomes:
* Hands on knowledge of some cryptographic primitives used in the blockchains
* In-depth study of blockchain technologies

Description:
* The blockchain and Byzantine fault tolerance.
* Proofs of work: design and implementation
* Cryptocurrencies: the Bitcoin protocol.
* Blockchain-based smart contracts: Ethereum
* Zero-Knowledge Succinct Non-interactive Argument of Knowledge (zk-SNARK) and the Zcash cryptocurrency protocol.

Teaching modality: The course is based on projects that can be done in pairs and the final presentation.

Language: Anglais

Mandatory: Non

Evaluation: the final grade is based on the grade for the project and technical quality of the final presentation.
### Advanced Database Topics

<table>
<thead>
<tr>
<th>Module:</th>
<th>Module 3.7 (Semester 3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECTS:</td>
<td>4</td>
</tr>
<tr>
<td>Objective:</td>
<td>The seminar provides an overview of current research topics and trends in the broad area of databases. We cover a variety of advanced database aspects such as temporal and probabilistic database models, distributed databases, graph databases, as well as respective applications in the context of information extraction, natural-language processing (NLP) and machine learning (ML). The seminar serves as an ideal basis for further topics in this area (such as Master projects and theses).</td>
</tr>
</tbody>
</table>
| Course learning outcomes: | - Students become familiar with recent trends and research topics in the broad area of databases and applications  
- Students learn which approaches and tools are best suited for a given problem setting  
- Students learn how to prepare both an oral presentation and a written report on a given research subject  
- The course serves as an ideal preparation for further Master projects and theses in this area |
| Description:           | The seminar is organized in the form of weekly student presentations based on the recent literature (such as research papers, journal articles and/or book chapters). Specifically, we focus on the following topics and applications:  
- Probabilistic and temporal databases  
- Distributed databases  
- Graph databases  
- Applications:  
  - Indexing and query processing  
  - Sampling and approximation  
  - Entity resolution and deduplication  
  - Data integration and fusion  
  - Provenance management  
  - Information extraction (with applications to NLP and ML) |
| Teaching modality:     | The course offers both theory lectures and tutored exercise sessions. |
| Language:              | Anglais                 |
| Mandatory:             | Non                     |
| Evaluation:            | Seminar Presentation: 40%, Final Written Report: 40%, Active Participation: 20% |
| Professor:             | THEOBALD Martin         |

### Autonomous Robot Software

| Module:                  | Module 3.8 (Semester 3) |
Master in Information and Computer Sciences  
- Master in Information and Computer Sciences

ECTS: 4

Objective: After the course, the students will have:
1. an overview of software and programming issues in robotics
2. An overview of knowledge representation and reasoning challenges for robotics
3. hands-on experience with ROS software and Nao robots in the CSC robolab

Description: This course is a collaboration between computer science (Leon van der Torre) and engineering (Holger Voos)
The course consists of fifty percent lectures and fifty percent practicals. In the lectures, Holger Voos gives an introduction to robotics, and Leon van der Torre discusses knowledge representation and reasoning for robotics. For the practicals, the students learn how to use ROS for building software for the Nao robots. ROS software is the de facto standard robot framework, widely used for component based robotic software engineering. The CSC robotlab is a collaborative research between CsC and SnT departments of the University of Luxembourg. We use NAO robots for research and education. For details checkout the project and publication sections on our website.

http://robolab.gforge.uni.lu/

Teaching modality: - 7 lectures for theory
- 7 lectures for practicals
- 1 week presentations
Each week, there will be a fixed afternoon where the CSC robolab is reserved for the students

Language: Anglais
Mandatory: Non
Evaluation: Project based, the students have to write some software and give a presentation on it
Remark: Teaching material:
- theory: Background material in textbooks
- K. Berns, E. von Puttkamer, Autonomous Land Vehicles: Steps towards Service Robots
- practicals: ROS wiki

Professor: VOOS Holger

Intelligent Agents II

Module: Module 3.9 (Semester 3)
ECTS: 4

Objective: The objective of this course is to introduce students to formal methods for normative reasoning for agents (deontic logic)

Course learning outcomes: * Capture the intuition behind rules and regulations, and how they relate to real life scenarios. Explain the nature and roles of the formal theories used for collective reasoning and
multiagent systems, such as game theory, social choice theory, normative reasoning, dialogue and argumentation

* Define the basic concepts of one or two logics for collective reasoning (e.g. deontic logic, action logics), notably its semantics and its proof calculi
* Model intelligent distributed systems using normative multiagent systems and deontic logic

* Explain the philosophical foundations of collective reasoning

Description:
Lectures: - rules and regulations - standard deontic logic - input/output logic - project

Teaching modality:
The course is organized as a series of theoretical lectures intermixed with practicals

Language:
Anglais

Mandatory:
Non

Evaluation:
Part I - Rules and Regulations:
Homework: 10%
Project: 40%

Part II - Deontic Logic:
Homework: 10%
Final exam: 40%

Remark:
Literature:
Handouts will be distributed during the course. In addition, the handbook of deontic logic provides background material

Professor:
VAN DER TORRE Leon

Machine Learning

Module: Module 3.10 (Semester 3)
ECTS: 4
Objective: The course "Machine Learning" is part of a course trilogy that is preceded by a course on 'Information Retrieval' (Semester 1) as well as by a course on 'Knowledge Discovery and Data Mining' (Semester 2), respectively. Originally having been a subfield of Artificial Intelligence, Machine Learning has nowadays established as an own research direction, whose main concern is to model and to simulate intelligence and to apply it for prediction, classification, and clustering - just to name a few. In this context, Deep Learning and Convolutional Networks - as part of the Neural Networks Theory - has become on everyone's lips for several years, promising a new way of solving complex problems under the disguise of a missing explainability.
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Course learning outcomes:
* List the fundamental concepts of Machine learning
* Explain the importance of an optimized modeling by Artificial Neural Networks for cognate application areas
* Implement fundamental concepts in an individual programming project
* Explain applications of Artificial Neural Network in academia and in industry

Description:
The course is a mixture of lecture and practical studies. While the first part of the course will continue to deepen the theoretical foundations - particularly in the sub-symbolic field-, the second part is devoted to TensorFlow (Google Machine Intelligence, see www.tensorflow.org) or an equivalent software library for a practical work on Deep Learning / Convolutional Network. Results of the project will be presented at the end of the lecture period at a workshop, which will be open to public.

Teaching modality: Lecture + Oberseminar

Language: Anglais

Mandatory: Non

Evaluation:
• 50% : Presentation and documentation of the project results.
• 50% : Written Test

Remark: REFERENCES
• TensorFlow: see www.tensorflow.org

Professor: SCHOMMER Christoph

Management of Information Security

Module: Module 3.11 (Semester 3)

ECTS: 4

Objective: The objective of this course is to prepare students to start a practical experience as Information Security Officer and to assume such responsibility in his future professional activities. The student will understand that her/his work consists mainly in the planning, coaching, designing of solutions and convincing both senior management and information users on the needs and effectiveness of security

Course learning outcomes:
* List typical tasks of a information security manager.
* Explain the standard approach to managing security (ISO 27001).
Master in Information and Computer Sciences - Master in Information and Computer Sciences

* Explain basics of information risk management (ISO 27005), like asset valuation, impact and occurrence assessment, risk treatment plans, risk acceptance.
* Apply a fast risk assessment method (by ENISA) to a small virtual organisation.
* Compare different approaches for risk assessment.
* Assess metrics to measure security (ISO 27004) and tactics to implement security (ISO 27003).
* Recognise practical aspects of an Information security policy, including incident management, business continuity management, asset classification, awareness rising.
* Propose an information security policy for a small organisation.

Description:
This course presents the core requirements and guidelines of ISO standards on security management, risk assessment and risk management, security policies, and awareness raising:
• Practical aspects of information security
• Standard approach to managing security (ISO 27000, 27001, 27002, 27003, 27004, ...)
• Basics of information risk management (ISO 27005), like asset valuation, impact and occurrence assessment, risk treatment plans, risk acceptance, ...
• Applying a fast risk assessment method (e.g. ENISA) on a small virtual organisation SME and learning more detailed risk management method (e.g. TRICK-light).
• Practical aspects of an Information security policy, including incident management, business continuity management, asset classification, awareness rising.

Teaching modality: The course is organized as a series of theoretical lectures with hands-on exercises and discussion of the solution of group and individual exercises intermixed.

Language: Anglais

Mandatory: Non

Evaluation:
Final Exam: 50%
Exercises: 50%

Professor: HARPES Carlo

Model-Driven Software Development

Module: Module 3.12 (Semester 3)

ECTS: 4

Objective:
This course has two main objectives: first the student will be taught a firm theoretical foundation for the fundamental concepts in model-driven software development. Second the student will be confronted with current frameworks, tools and techniques in this field.

Course learning outcomes:
* Explain the fundamental concepts of model-driven software development
* Formally define a new modeling language via its syntax and semantics
* Evaluate current techniques and frameworks in model-driven software development
* Read and present research papers in the area

Description:
Introduction, Review of Alloy

Fundamental Concepts: Models, Metamodels, Modelling Hierarchies; Syntax and Semantics of Modelling Languages, Denotational Semantics, Operational Semantics: Graph Transformations versus Alloy;
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Frameworks and tools: the Eclipse Modelling Framework (EMF); Graphical Modelling (Sirius); the Lightning Tool; the Unified Modelling Language (UML).

Teaching modality: The course is organized as a series of theoretical lectures intermixed with hands-on exercises

Language: Anglais

Mandatory: Non

Evaluation:
- Final Exam: 60%
- Project: 40%

Remark: The following textbooks will be used in this course:

Professor: KELSEN Pierre

Selected Topics in Network and System Security

Module: Module 3.13 (Semester 3)
ECTS: 4

Objective: The objective of this course is an elaborated, active understanding of distributed vs. centralized communication security and privacy concepts and their application to cooperative environments. The course creates links between the fundamental concepts and applied scenarios with references to ongoing research activities within the SECAN-Lab research group.

Course learning outcomes:
* Describe existing concepts to set-up centralized and distributed secure communication systems and protocols.
* Name and reproduce definitions for quantitative parameters linked to anonymity, privacy and reputation systems
* Analyze and compare existing privacy-enabled systems for cooperative and non-cooperative environments
* Construct, adapt and assess real world communication architectures and protocols for a distributed cooperative communication problem

Description: This course will provide an overview of the topic of security and privacy in computer networks. Concrete topics and application scenarios can vary depending on the particular focus of the methods discussed. Typically, the methods of SECAN-Lab involve data analytics for network management, software defined networking, machine learning, algebraic graph transformation, symbolic execution, device and network fingerprinting, deterministic and probabilistic medium access control, routing strategies applied to scenarios around vehicular communications, network management and forensics, cybersecurity, anonymity and privacy, financial technologies and others.

Teaching modality: The course consists of two to three parts:

In the first part, a set of lectures gives an introduction to the topics including some necessary background. If the number of students is not too large, it is followed by a second part, a seminar-style preparation of the state-of-the-art, where students give a presentation on the topics
described above. The third part consists of a hands-on implementation to apply the developed knowledge in a concrete, research-oriented, scenario.

**Language:** Anglais  
**Mandatory:** Non  
**Evaluation:** 70% Final exam  
30% Assignments (presentation, written summary and practical results)  
**Professor:** ENGEL Thomas

---

**Open Network Security**

**Module:** Module 3.14 (Semester 3)  
**ECTS:** 4  
**Objective:** The goal of this course is to introduce challenges in securing computer systems and networks. The course covers the foundations of system and network security. It introduces basic security and privacy concepts as building blocks for later specialization.

**Course learning outcomes:**  
* Master the epidemiological models for malware propagation and network-centric attacks  
* Assess the security of networks and software systems.  
* Explain the techniques for worm containment and detection.  
* List statistical and machine learning approaches for network traffic monitoring

**Description:** In the scope of this course, we explore the following topics:  
Introduction and general concepts.  
Definitions, security goals, attacker models.  
Symmetric encryption.  
Integrity protection.  
Asymmetric encryption.  
Certificates and public key infrastructure (PKI).  
Authentication and key agreement.  
Kerberos.  
Security protocols (IPSec, DNSSec, Kerberos, SSH, TLS, etc.).  
Spam, botnets, phishing.  
Anonymity and privacy.  
Machine learning and intrusion detection.  
Wireless security.

**Teaching modality:** Lectures, exercises, practical design and implementation of lab experiments, presentation of own contributions

**Language:** Anglais  
**Mandatory:** Non  
**Evaluation:** Final exam, active participation in exercises, practical assignments and presentation.  
**Professor:** ENGEL Thomas
Parallel and Grid Computing

Module: Module 3.15 (Semester 3)  
ECTS: 4  
Objective:  
Today, parallel computing is omnipresent across a large spectrum of computing platforms. At the microscopic level, processor cores have used multiple functional units in concurrent and pipelined fashions for years, and multiple-core chips are now commonplace with a trend toward rapidly increasing numbers of cores per chip. At this level, GPU also start to enter the area. At a more macroscopic level, one can now build clusters of hundreds to thousands of individual (multi-core) computers. Such distributed-memory systems have become mainstream and affordable in the form of commodity clusters. Furthermore, advances in network technology and infrastructures have made it possible to aggregate parallel computing platforms across wide-area networks in so-called grids. An efficient exploitation of parallel and distributed platforms requires a deep understanding of both architecture, software and infrastructure mechanisms and of advanced algorithmic principles. The aim of this course is thus twofold. It aims at introducing the main trends and principles in the area of high performance computing infrastructures, illustrated by examples of the current state of the art.

It intends to provide a rigorous yet accessible treatment of parallel algorithms, including theoretical models of parallel computation, parallel algorithm design for homogeneous and heterogeneous platforms, complexity and performance analysis, and fundamental notions of scheduling and work-stealing.

Course learning outcomes:  
* Identify the key concepts related to parallel computing  
* Solve problems using a parallel decomposition  
* Use parallel programming platforms, models and frameworks  
* Implement algorithms using MPI

Description:  
Parallel computing architectures  
Introduction to parallel computing using MPI  
Programming labs based on MPI on UL clusters

Teaching modality:  
A couple of theoretical lectures are proposed to described parallel computing computing architectures, design and implementation methods. This lecture is intensive in terms of C/C++ programming coupled with parallel libraries such as MPI

Language: Anglais  
Mandatory: Non  
Evaluation: Project: 100%  
Professor: BOUVRY Pascal, VARRETTE Sébastien
Security Modelling

Module: Module 3.16 (Semester 3)  
ECTS: 4  
Objective: The objective of the course is to provide an overview of techniques used to model and analyze the security of systems. After successful completion of this module the student should be able to  
- understand the role of modelling in the design and analysis of secure systems  
- understand the role of modelling in defining security properties and attacker capabilities.  
- select a suitable modeling technique for a given problem  
- apply a given modelling technique to a specific domain or problem  
- assess the security of a given system through its modeling  
- compare modelling techniques with respect to expressive power and suitability for a given domain  
- identify the limitations of modelling techniques with respect to real-world security  

Course learning outcomes:  
* select a suitable modeling technique for a given problem  
* apply a given modeling technique to a specific domain or problem  
* assess the security of a given system through its modeling  
* compare modeling techniques with respect to expressive power and suitability for a given domain  
* describe new trends in security modeling  
* identify the limitations of modeling techniques with respect to real-world security  

Description: The following topics will be presented during the class  
- The characterization of secrecy properties as process equivalence or indistinguishability.  
- Commitment schemes and zero-knowledge proofs  
- Quantum cryptography, in particular quantum key distribution  
- Verifiable voting schemes  
- Provable security  

Teaching modality: The course is organized as a series of 9 theoretical lectures and 4 seminar sessions. The lectures are mostly based on recent research papers. A set of exercises is proposed as a homework to be solved by the students. A set of topics will be proposed near the beginning of the course. Each student is asked to pick one topic and present it by giving a 20 min presentation. A research report, taking into account the feedback obtained during the seminar, will be written by the student.  

Language: Anglais  
Mandatory: Non  
Evaluation: Homework: 20%  
Presentation + research report: 80%  
Professor: RYAN Peter Y A
Security Protocols

Module: Module 3.17 (Semester 3)
ECTS: 4
Objective: The objective of the course is to provide students with an in-depth knowledge of the methods and tools for the specification, design and analysis of security protocols in different domains

Description:
- Review of basic cryptography
- Protocol specification
- Protocol execution model
- Adversary Models
- Security properties
- Secrecy
- Authentication
- Privacy, Anonymity, Untraceability
- Automatic Verification of Security Protocols
- Non-Repudiation Protocols
- Fair Exchange Protocols

Time permitting, one or more of the following topics will be covered.
- Distance Bounding
- RFID Protocols
- Secure Multi-Party Computation Protocols

Teaching modality: The course is organized as a series of lectures with homework assignments

Language: Anglais
Mandatory: Non
Evaluation: Evaluation: Homework 30% + Final Exam 70%
Professor: MAUW Sjouke
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

Selected topics in Artificial Intelligence

Module: Module 3.18 (Semester 3)
ECTS: 4
Objective: The objective of this course is to prepare the student for individual research work (e.g. a master/PhD thesis) in Artificial Intelligence, Knowledge Representation and Logic by studying research literature on a specific topic and possibly realizing a small project. It offers a gentle introduction into current research issues linked to the modeling of intelligent agents, paying special attention to those addressed in the Interdisciplinary Lab for Intelligent and Adaptive Systems (ILIAS).

Course learning outcomes:
* Analyze a specific research problem in Artificial Intelligence (AI) and existing ways to tackle it
* Explain and apply corresponding advanced techniques in AI
* Read, analyse, and discuss research papers in AI
* Present and explain research work in AI

Description: The topics can vary. In the past years we have discussed in particular causal reasoning, defeasible reasoning, real-world argumentation, story modeling, generalized reasoning under uncertainty, epistemic decisions, reasoning about actions, defeasible argumentation, general probabilistic reasoning, as well as knowledge and belief dynamics. The seminar theme for WS 2019-2020 will be

“Explanation and Artificial Intelligence”.

Recent years have seen tremendous progress in some areas of artificial intelligence, especially in vision and natural language processing. They profit from the availability of huge data sets, powerful hardware, and new machine learning algorithms, including but not limited to so-called deep learning techniques. Similarly, we see the emergence of (semi-)autonomous bots and robots interacting with humans in many areas of life. More adventurous application contexts are foreseen in a not too far away future, with some people even starting to fantasize about superintelligent machines in the long run. These highly complex dynamic systems entering our daily world are however often black boxes which may produce seemingly useful results, but in a highly opaque way, without much indications about their actual limitations in a real-world environment. Similarly, the real world itself, with all its complexity, also constitutes - in most dimensions - a black box for artificial agents.

A broadscale introduction of and reliance on (semi-)intelligent machines for possibly critical decisions will clearly require more transparency and the ability of the systems to explain why they are doing what they are doing. This accessibility is necessary to induce sufficient trust, to meet application requirements and legal demands of different types (see new EU rules w.r.t. explainability), to cooperate with humans, and to allow the improvement of the systems in their application contexts. On the other hand, intelligent behaviour itself presupposes that the AI agents themselves are able to explain and interpret their observations when pursuing their goals and build their world models and belief states.

In this seminar we are therefore going to read, discuss, and criticize a number of introductory and research-level texts dealing with explanation in AI. While this area has been around for several decades, it saw an important growth in recent years, for obvious reasons. The task for science is to fix the relevant concepts, partly imported from other domains, and to develop techniques
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

supporting explanation and understanding for and of intelligent systems, building the base for a coexistence of humans and artificial intelligences.

Some basic knowledge in logic, probability theory, and knowledge representation would be useful. If necessary, additional background material can be provided.

Teaching modality:
Language: Anglais
Mandatory: Non
Evaluation:
- 1-2 presentations (50%)
- Participation during lectures, including homework (15%)
- Course project or written tests (35%)

Professor: WEYDERT Emil, DAUPHIN Jérémie

Software Engineering Environments

Module: Module 3.19 (Semester 3)
ECTS: 4
Objective: At the end of this course, the student should be capable to understand the concepts, principles and practices related to DevOps along with their role in the software engineering life cycle, both from a theoretical and practical viewpoint.

Course learning outcomes:
- LO1: Set up a Deployment pipeline for a particular software development project
- LO2: Define, and perform a classification of tools based on quality attributes
- LO3: Use and integrate existing tools
- LO4: Produce a scientific report
- LO5: Plan, coordinate, and report activities in a multi-participant project

Description: DevOps (the combination of Development and Operations) is a new cultural shift that has emerged as response to the need of ensuring high quality for a particular software system, while reducing the time since a new change is introduced into such a software system until it makes it into production. In this manner, organizations aimed at applying this cultural shift need to make use of advanced software development methods. In this course, particular attention will be given to continuous integration, automated deployment and software delivery. These methods are chained one after the other, forming what is known as deployment pipeline.

Beside the software engineering aspects, DevOps also includes other two important dimensions: culture, and management. Culture corresponds to the required skills to be able to collaborate with other people in a professional manner, whereas management is related to the ability to find trade-offs and make decisions such that the expected outcomes are satisfy while respecting the
imposed constraints. The three dimensions are equally important, so they are the focus of the course.

**Teaching modality:** The course is centred around a project. The aim of the project is to let students incrementally develop a continuous delivery pipeline. Students are expected to deliver one or more components of such a pipeline by each milestone deadline.

In order to attain this objective, the course is organised as a combination of traditional lectures with in-class activities. The course starts by a series of lectures introducing the fundamental concepts and practices related to DevOps. Then practical sessions are organized such that teams report on their progress about the project. Thus, teams guided by the teaching staff, will not only design a DevOps pipeline, but also develop, extend or simply setup existing software engineering tools to achieve the implementation of such a pipeline and then demonstrate its functioning.

The number of students on each team will depend on the size of the class, but ideally, they will be made of up to 3 students.

**Language:** Anglais

**Mandatory:** Non

**Evaluation:**
- Project deliverables (40%)
- In-class assessment (20%)
- Project Checkpoints (20%)
- Project Peer review (20%)

Another method for repeat exams than the one indicated is to be used.

**Remark:**
- Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation (Addison-Wesley Signature Series (Fowler)) Addison-Wesley Professional; 1 edition (August 6, 2010)

**Professor:** GUELFI Nicolas, CAPOZUCCA Alfredo

### Testing and Validation

**Module:** Module 3.21 (Semester 3)

**ECTS:** 4

**Objective:** The goal of testing and validation is to assess the consistency/ conformity of a product with respect to its specification. These activities are thus crucial and costly activities for software companies, and eventually aim at providing a controlled level of trust.
in the final product before it is delivered to the client (and then during maintenance)

**Course learning outcomes:**
* participate in the software testing process of a company
* carry out research in the area of software testing, diagnosis, validation
* develop new models and solutions for applied software testing

**Description:** Testing is related to all the design stages of the development process and must deal with many application contexts (embedded systems, information systems ...) and various dimensions of complexity (programming-in-the-small, in-the-large and in-the-duration). This course reminds the principles of software testing, the main techniques and introduces new advanced techniques (similarity-based, security testing...)

**Teaching modality:** The course is composed of a set theoretical courses which are illustrated with practical exercises (lab and paper exercises). The content of the course vary from one year to another taking into account the progress of the domain (MBT, code-based approaches)

**Language:** Anglais

**Mandatory:** Non

**Evaluation:** The students will be evaluated through homeworks, typically review/report on an advanced testing technique, and a final presentation.

**Professor:** LE TRAON Yves

---

### Ubiquitous Computing

**Module:** Module 3.22 (Semester 3)

**ECTS:** 4

**Objective:** In this seminar, students will be prepared for scientific research. They will develop the ability to analyse a topic, extract knowledge from articles and books, finally communicating that knowledge in a clear and concise fashion both in terms of a presentation as well as a detailed report.

**Course learning outcomes:**
* Determine and examine useful information sources
* Analyse and assess the relevance of scientific articles
* Break down core ideas and contributions presented in those articles
* Integrate those core ideas into a coherent view
* Summarise the most important aspects in a concise report
* Prepare and conduct a presentation embracing those important aspects
* Justify the findings during a scientific discussion

**Description:** Each student individually explores some well-defined topic in the realm of mobile and ubiquitous computing. Coached by the supervisors, students need to identify proper information sources and select the most relevant publications. They must analyse and synthesise the core ideas. In a presentation, students will demonstrate the knowledge gathered together with their analysis, followed by a discussion. A final report will summarise the entire work.

**Teaching modality:** There will be some presence events:
- Introduction, topic presentation and selection
- "What's a Good Paper"-Workshop
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

- Pecha Kucha style short presentations
- Individual meetings to discuss the report structure and topic coverage
- Presentation Zen event
- Final presentations
Aside from those presence events, students work individually on their topic.

Language: Anglais
Mandatory: Non
Evaluation: Pecha Kucha style short presentation (20%), report (50%), final presentation (30%)
Participation to all presence events as well as completing all deliverables.
Note: Prerequisite to passing are the participation to all presence events as well as the completion of all deliverables.

Professor: ROTHKUGEL Steffen

Fault and Intrusion Tolerance

Module: Module 3.23 (Semester 3)
ECTS: 4
Objective: This advanced course exposes students to the problem of dependability and security of computer systems and communication networks --- with a slant toward distributed systems --- and to solutions under the tolerance paradigm, or how to prevent failures, especially security, under the allowed presence of accidental and malicious threats (faults, vulnerabilities, attacks and intrusions).

Course learning outcomes: A solid training in security and dependability of information systems leverages knowledge in: theory including fundamental paradigms and architectures; knowledge of techniques and tools for the design and development of secure and dependable systems and of their components; and, finally, practical knowledge and experience in the application of the paradigms and tools in diverse situations and execution environments, from networked embedded systems to large-scale systems in the web. The syllabus of this curricular unit contributes both to the objectives of the unit as well as to the program objectives.

Description:
1. Review of fundamental security and dependability concepts
2. Fault and Intrusion Tolerance (FIT) concepts and terminology
3. FIT frameworks and mechanisms
4. Fault and Intrusion Tolerance strategies
5. Modeling faults
6. Architecting fault-and-intrusion-tolerant systems
7. Tolerating Intrusions
8. Resisting Attacks
9. Testing Attacks

Teaching modality: Classes are organized in the following manner: Lecture and practice classes, including guided sessions.
- Lecture classes aim at exposing the course matters. They guide the flow of subjects, which the student has to pursue in the textbook and deepen through the reading of complementary text book excerpts, as well as papers, manuals and annotations.
Master in Information and Computer Sciences
- Master in Information and Computer Sciences

- Practice sessions aim at exposing some lecture material with more detail (e.g., algorithms or package interfaces or internals), or let the student get a deeper and practical insight on other lectured materials. Guided sessions aim at helping the student get started with assignments, experiments, and projects.
- Autonomous work concerns reading assignments and a project in this course. The reading assignments concern either an overview of a system, technology, or set thereof, or the study of a set of research papers. The format of the assignments will comprise combinations of paper reviews and written surveys, and/or oral presentations of papers. The project will consist of the paper design of an intrusion tolerant system. The design will be defended by the author(s) in a classroom presentation. Assignments will be distributed evenly throughout the semester.

Language: Anglais
Mandatory: Non
Evaluation:
Grading: Reading and Project Assignments (45%), Final Exam (50%), Class participation (5%).
Eliminating: Assignments, Final Exam.

Remark:
Recommended readings will form the thread of the course explanation, consolidated by complementary readings by the students to get deep insight on several aspects of the matters.
- Complemented by additional research and design papers, available from the course web.

Professor: ESTEVES-VERISSIMO Paulo
### Master thesis

<table>
<thead>
<tr>
<th><strong>Module:</strong></th>
<th>Module 4.1 (Semester 4)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ECTS:</strong></td>
<td>30</td>
</tr>
<tr>
<td><strong>Language:</strong></td>
<td>Anglais</td>
</tr>
<tr>
<td><strong>Mandatory:</strong></td>
<td>Oui</td>
</tr>
<tr>
<td><strong>Professor:</strong></td>
<td>SORGER Ulrich</td>
</tr>
</tbody>
</table>