COMPUTER SYSTEMS ENGINEERING – 2nd Cycle

Courses offered during Winter semester (academic year 2026-27)

Applied Machine Learning (1st year, 6 ECTS)

Syllabus
1. Unsupervised Machine Learning
1.1 Dimensionality Reduction
1.2 Clustering algorithms
1.3 Gaussian Mixtures
2. Ensemble Models for Machine Learning
2.1 Bagging, Boosting, Voting and others
2.2 Random Forest
2.3 Ada Boost and Gradient Boost
3. DNN and CNN Networks
3.1 Introduction to CNN and DNN
3.2 DNN Frameworks: Tensorflow/Keras and others
3.3 Training and customizations of MLP models
4. Processing Sequences using RNNs and CNNs
4.1 RNN Networks
4.2 Training RNNs
4.3 Forecasting Time Series
5. Machine Learning applied to text and NLP
5.1 ML applications in NLP and Text Mining
5.2 Word Embeddings
5.3 Deep Learning for NLP and Transformers
6. Reinforcement Learning
6.1 Markov Decision Processes (MDP)
6.2 Reinforcement Learning, policies, rewards
6.3 Deep Reinforcement Learning

Teaching methodology and assessment:
The methodology involves three main components:

- Theoretical-practical classes structured to present the main theoretical concepts of Machine Learning, combining theoretical explanations with practical demonstrations and the resolution of exercises. The concepts are presented interactively, using code examples in languages such as python. During the lessons, code snippets are analysed, allowing students to see in practice how the theories apply to software development adequate to ML. In addition, exercises are proposed for resolution in class, challenging students to apply the concepts discussed and solving problems that simulate situations found in real systems. This approach promotes active learning, while strengthening the understanding of concepts through practice.

- Practical laboratory classes dedicated to solving problems related to ML by means of practical worksheets that guide students in solving technical challenges related to programming applied to ML. Each worksheet addresses a set of topics in an applied way according to the syllabus and the learning outcomes. These classes provide an environment where students can experiment directly with concepts of ML and receive immediate feedback from teachers, which makes it easier to correct errors and understand the underlying concepts. Students work individually or in small groups to promote collaboration and the exchange of knowledge.
- Practical projects: programming in ML projects are a crucial part of the methodology, as they allow students to apply the knowledge they have acquired in practical, integrated projects. These projects are carried out in groups, encouraging collaboration and the development of teamwork skills. The projects are divided into phases, starting with the definition of requirements, moving on to the design and implementation of solutions, and ending with the evaluation of results. In addition, code review sessions and project presentations are organised to develop communication and critical evaluation skills.

Assessment elements: 2 tests, project, comprehensive exam

Bibliography

[1] Russell, Stuart ; Norvig, Peter, "Artificial Intelligence: A Modern Approach", Prentice Hall, 4th edition, 2020, online: http://aima.cs.berkeley.edu
[2] Costa, E.; Simões, A., "Inteligência Artificial - Fundamentos e Aplicações", Editora FCA, 2ª edição, 2008
[3] Witten, Frank, Hall, Pal, "Data Mining: Practical Machine Learning Tools and Techniques, 4rd Edition", Morgan Kaufmann, 2017
[4] Aurélien Géron, Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O'Reilly, 2019
[5] Zhang, Aston and Lipton, Zachary C. and Li, Mu and Smola, Alexander J., Dive into Deep Learning, 2022, online: https://d2l.ai

Intelligent Sensory Systems (1st year, 6 ECTS)

Syllabus

1. Introduction to Intelligent Sensory Systems (ISS)
1.1 Characteristics and objectives
1.2 Key capabilities and challenges
1.3 Examples of SSI - Ubicomp/IoT
2. Wireless Communications and Protocols
2.1 Coding and multiplexing
2.2 Wireless transmission technologies
2.3 Protocols in sensor networks
3. Context and Localisation in SSI
3.1 Context models
3.2 Location technologies
3.3 Architectural aspects
4. System requirements/aspects
4.1 Dynamic adaptation
4.2 Profiling and energy management
4.3 Security and privacy
5. Implementing SSI at Edge and Fog
5.1 Embedded platforms and micro-controllers
5.2 Integration of sensors and actuators

5.3 Wireless communications
5.4 Operating modes and energy management
5.5 Incorporating TinyML/Edge-AI
6. Implementing SSI in the Cloud
6.1 Cloud integration and management platforms
6.2 Storage, visualisation and automation
6.3 Processing and analysing sensory data
6.4 Incorporating Cloud-AI

Teaching methodology and assessment:

Teaching methodology:

- Theoretical-practical classes on the main concepts of ISS, such as the characteristics of intelligent sensory systems, wireless communication technologies and system requirements such as energy management and security, with the practical discussion of cases and examples.
- Practical laboratory classes are aimed at applying concepts through the practical development of solutions with Edge/Fog and Cloud platforms. Students carry out practical activities that include the integration of sensors and actuators, as well as the use of management platforms in the Cloud.
- Laboratory project: here students are challenged to develop a complete intelligent sensory system. The project promotes the application of technical skills in the collection and manipulation of environmental context data, in the handling and application of Edge/Fog and Cloud platforms, thus developing students' collaborative work skills and autonomy.

Assessment: 2 tests, project, comprehensive exam

Bibliography

[1] Chou, T., Precision: Principles, Practices and Solutions for the Internet of Things, 2nd Ed., Cloudbook Inc, 2020
[2] Krumm, J. (Ed.), Ubiquitous Computing Fundamentals, CRC Press, Taylors & Francis Group, 2010
[3] Moreira, R.S., Soares, C., Torres, J., Sobral, P., Intelligent Sensing and Ubiquitous Systems (ISUS) for Smarter and Safer Home Healthcare. Chap 1 In Intelligent Pervasive Computing Systems for Smarter Healthcare (Eds A.K. Sangaiah, S. Shantharajah and P. Theagarajan) (pp 1-36), John Wiley & Sons Inc., 2019
[4] Moreira, R.S., Soares, C., Torres, J., Sobral, P., Combining IoT architectures in next generation healthcare computing systems. In Intelligent IoT Systems in Personalized Health Care (Cognitive Data Science in Sustainable Computing series) (pp. 1-29), Elsevier, Academic Press, 2021
[5] Karvinen, T., Karvinen, K., Valtokari, V., Make: Sensors – Projects and Experiments to Measure the World with Arduino and Raspberry Pi, Make Community, 2014

Semantic Web and Web 3.0 (1st year, 6 ECTS)

Syllabus

1. Introduction to the Semantic Web: concepts, principles, and goals.
2. Ontology construction and management.
3. Web 3.0 and decentralization: blockchain and smart contracts.
4. Semantic interoperability and data integration.
5. Use cases: e-commerce, social networks, and healthcare.

Teaching methodology and assessment:

Teaching methodology:

- Theoretical-Practical Classes: introduction and discussion of fundamental concepts of the Semantic Web (CP1), construction and management of ontologies and principles of Web 3.0. Discussion of real cases and debates to stimulate critical thinking.
- Practical laboratory classes: using tools such to build ontologies. Experimenting with blockchain platforms to explore smart contracts. Practical activities on interoperability and data integration.
- Practical project: development of a practical use case that integrates Semantic Web and Web 3.0 concepts. Delivery of a practical solution with technical documentation and public presentation.

Assessment components: 1 theoretical-practical test: (40% of the final grade), practical project (50% of the final grade), presentation and defense (10% of the final grade)

Bibliography

[1] Antoniou, G., & Van Harmelen, F. (2021). A Semantic Web Primer. MIT Press.
[2] Hitzler, P. et al. (2020). Foundations of Semantic Web Technologies. Chapman and Hall/CRC.
[3] Mougayar, W. (2016). The Business Blockchain. Wiley.
[4] Documentation from W3C on RDF, OWL, and SPARQL.
[5] Protégé: Ontology Editor.

Data Science (1st year, 6 ECTS) – optional

Syllabus

1. Data preparation.
1.1 Data import
1.2 Data manipulation and data preparation for analysis
1.3 Observation unit.
1.4 Data aggregation.
1.5 Data merge.
1.6 "Feature engineering"
2. Data visualization.
2.1 Core concepts of visual perception.
2.2 Different data visualization techniques
3. Exploratory data analysis
3.1 Numerical and graphical description of data.
3.2 Outlier analysis - Missing analysis.
3.3 Exploration of data under different dimensions.
4. Prediction models with real-world data.
4.1 "Feature Selection"
4.2 Implementation and evaluation of forecasting models.
4.3 Time Series Analysis
5. Clustering Models with real-world data.
5.1 Cluster analysis
5.2 Analysis of latent classes
5.3 Model evaluation

Teaching methodology and assessment:

Teaching methodology:

- Interactive theoretical classes: These sessions are dedicated to explaining the fundamental concepts of each topic, using real examples and interactive discussion to promote student understanding. The instructor explores theoretical concepts and demonstrates their application in practical cases, encouraging questions and debates to deepen understanding.
- Practical laboratories: In these sessions, students apply the knowledge acquired in theoretical classes to real data sets, using data analysis software.
- Group projects: Students work on group data analysis projects that simulate real-world business challenges.
- Case studies and critical analysis: Analysis of real case studies to understand the application and outcomes of analytical strategies in business contexts.
- Performance-based assessment: The assessment of students is continuous and based on performance in practical projects, laboratory exercises, class participation, and presentations.
- This pedagogical model promotes deep learning of Data Science concepts and their strategic application in business, preparing students to face analytical challenges in the contemporary work environment.

The assessment components are distributed as follows:

- Group Practical Project: 40% (development and presentation of a project that applies the concepts and techniques learned throughout the course to a real data set, including data preparation, visualization, exploratory analysis, implementation, and evaluation of predictive and clustering models)
- Comprehensive exam: 30% (Written assessment covering all syllabus content, focusing on theoretical understanding, practical application of concepts, and critical analysis skills.
- Practical Laboratory Exercises: 20%
- Participation and classroom activities: 10% (Assessment based on active participation in theoretical and practical classes, contributions to discussions, teamwork ability during group projects, and engagement in interactive learning activities.)

Bibliography

[1] Grolemund, G., & Wickham, H. (2017). R for Data Science. O'Reilly Media.
[2] Boehmke, Bradley C. (2016) Data Wrangling with R
http://link.springer.com/book/10.1007%2F978-3-319-45599-0
[3] Alberto Cairo. 2016. The Truthful Art: Data, Charts, and Maps for Communication (1st. ed.). New Riders Publishing, USA.
[4] Irizarry, R. A. (2019). Introduction to Data Science: Data Analysis and Prediction Algorithms with R. United States: CRC Press.
[5] Timbers, Tiffany, et al. Data Science: A First Introduction. United States, CRC Press, 2022.

Wireless Sensor Networks (1st year, 6 ECTS) – optional

Syllabus

1. Introduction to wireless communication technology
2. Data transmission in wireless networks
3. Basic concepts of antennas and signal propagation in wireless networks
4. Terminal multiplexing techniques in wireless networks
5. Wireless local area network technologies (WLANs)
6. Wireless network technologies for IoT (WPANs and LPWANs)
7. Cellular networks (WWANs)

Teaching methodology and assessment:

Teaching methodology:

- Theoretical-practical classes are structured to present the main theoretical concepts of wireless communication technologies, combining theoretical explanations with practical demonstrations and the resolution of exercises. The concepts are presented interactively, using examples on emulators to illustrate how the mechanisms discussed work. During the lessons, the relevant network protocols are analysed using examples presented on emulators and traffic captures taken and analysed in real time. In addition, exercises are proposed for resolution in class, challenging students to apply the concepts discussed and solving problems that simulate situations found in real systems.
- Practical laboratory classes are dedicated to implementing solutions to specific problems by means of practical worksheets that guide students in solving technical challenges related to the design, configuration and management of wireless networks. Each worksheet addresses a set of topics in an applied way. Students work individually or in small groups to promote collaboration and the exchange of knowledge.
- Practical projects are a crucial part of the methodology, as they allow students to apply the knowledge they have acquired in integrated projects. These projects are carried out in groups, encouraging collaboration and teamwork. The projects are divided into phases, starting with the definition of requirements, moving on to the design and implementation of solutions, and ending with the evaluation of results. In addition, implementation review sessions and project presentations are organised to develop communication and critical evaluation skills.

The assessment components are distributed as follows: 2 tests, project, comprehensive exam

Bibliography

[1] Beard, C.; Stallings, W. – "Wireless Communication Networks and Systems" - Pearson, 2016.
[2] Schiller, J. – Mobile Communications 2nd ed. – Addison Wesley 2003.
[3] Tanenbaum, A.; Feamster, N.; Wetherall, D. – "Computer Networks 6th Edition – Pearson 2020.
[4] IEEE Wireless Communications, https://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7742

✪✪✪

Courses offered during Spring semester (academic year 2026-27)

Intelligent Mobile Applications (1st year, 6 ECTS)

Syllabus

1. Introduction to Intelligent Mobile Applications
1.1 Android platform and ecosystem
1.2 Development environment and programming model
1.3 Context-aware intelligent apps
2. Accessing and Managing Sensor Data
2.1 Sensor framework in Android
2.2 Environment sensor API
2.3 Motion sensor API
2.4 Position-sensing API
3. Sensor-based applications
3.1 Real-time data acquisition and storage
3.2 Sensor data pre-processing
3.3 Sensor data management and fusion

3.4 Real context-aware applications
4. Multimedia Data Access and Management
4.1 Handling and storing media
4.2 Image capture and processing
4.3 Audio capture and processing
5. Frameworks of AI/ML for Mobile Applications
5.1 Comparison of on-device versus cloud-based approaches
5.2 Characterization of frameworks for mobile applications
5.3 Optimization and compression of models
5.4 Integration, acceleration and performance aspects
5.5 Implementation and lifecycle management
5.6 Case studies/application

Teaching methodology and assessment:
Teaching methodology:

- Theoretical-Practical Classes: sessions that combine theoretical exposition with practical discussion, aimed at introducing the fundamentals of developing intelligent and context-sensitive mobile applications based on sensory data.
- Practical Laboratory Classes: supervised practical activities, where students implement specific functionalities, such as acquisition, pre-processing, and fusion of sensory data. The use of multimedia data with the management and integration of AI/ML tools for the development of intelligent mobile applications also applied. These laboratories aim to develop technical skills and familiarize students with the integration and customization of AI/ML tools applied in the development of mobile applications.
- Laboratory Project: development of a practical project covering all stages of creating a smart mobile application.

Assessment:

- Theoretical-Practical Tests: includes 2 tests throughout the semester
- Project: integrative practical project carried out in groups, in which students are presented with real examples of mobile applications.
- Comprehensive exam: consisting of a comprehensive quiz, applied using an LMS. This exam will cover the identification, analysis and application of aspects related to tools for developing intelligent and context-sensitive mobile applications.

Bibliography
[1] Bill Phillips, Chris Stewart, Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, 5th Ed., Addison-Wesley Professional, 2022
[2] Vasco Correia Veloso, Hands-On Artificial Intelligence for Android, 1st Ed., BPB Publications, 2022
[3] John Horton, Android Programming for Beginners – 3rd Ed., Packt, 2021. URL: https://github.com/PacktPublishing/Android-Programming-for-Beginners-Third-Edition?tab=readme-ov-file (Last visited: Nov 2024)
[4] Android Developers, Sensors Overview - Sensors and location, URL: https://developer.android.com/develop/sensors-and-location/sensors/sensors_overview
[5] Google AI for Developers, Google AI Edge - MediaPipe Solutions Guide, URL: https://ai.google.dev/edge/mediapipe/solutions/guide (Last visited: Nov 2024)
[6] Google for Developers, ML Kit, URL: https://developers.google.com/ml-kit?hl=pt-br (Last visited: Nov 2024)
[7] Google AI for Developers, Google AI Edge - Introducing LiteRT, URL: https://ai.google.dev/edge/litert (Last visited: Nov 2024)

Advanced Distributed Systems (1st year, 6 ECTS)

Syllabus
1. Fundamentals SD
1.1. SD fallacies and assumptions
1.2. Complex architectures and constraints
1.3. Monoliths, SOA, and Microservices
2. Styles, components, and tools
2.1. Architectural styles
2.2. Infrastructure – virtualization, containers, and services
2.3. Microservices, Serverless, and FaaS
2.4. Event-driven architectures
3. Communication, coordination, and consistency
3.1. Synchronous vs. asynchronous systems
3.2. Orchestration vs. choreography
3.3. Distributed state, consistency, and Saga
4. Scalability, resilience, and observability
4.1. Scalability models
4.2. State management and scalability
4.3. Performance and efficiency
4.4. Observability and resilience
4.5. Security
5. Agent-based systems
5.1. Autonomy vs. automation
5.2. Multi-agents
5.3. Coordination and orchestration
5.4. Deployment and operation
6. Frameworks
6.1. Environments and execution abstractions - sidecar
6.2. Event-driven infrastructures
6.3. Development and orchestration
6.4. Standardized operations and observability

Teaching methodology and assessment:
Teaching methodology :
- Theoretical-Practical Classes: sessions combining conceptual presentation with guided discussion and analysis of architectural styles, communication models, scalability, resilience, observability, and agent-based systems, supporting the understanding and justification of architectural decisions.
- Practical Laboratory Classes: supervised activities focused on the design and implementation of distributed system components, communication mechanisms, and operational concerns using modern frameworks and execution platforms, fostering applied skills and critical analysis.
- Laboratory Project (PROJ): development of a distributed system project integrating architectural design, coordination mechanisms, scalability, observability, and, when applicable, agent-based approaches, promoting teamwork, autonomy, and the consolidation of acquired competencies.

Assessment: 2 tests, project, comprehensive exam

Bibliography
[1] Bass, L., Clements, P., & Kazman, R. (2021). Software architecture in practice (4th ed.). Addison-Wesley
[2] Coulouris, G., Dollimore, J., Kindberg, T., & Blair, G. (2012). Distributed systems: Concepts and design (5th ed.). Addison-Wesley
[3] Tanenbaum, A. S., & Van Steen, M. (2017). Distributed systems: Principles and paradigms (3rd ed.). Pearson
[4] Kleppmann, M. (2017). Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems. O'Reilly Media
[5] Newman, S. (2021). Building microservices: Designing fine-grained systems (2nd ed.). O'Reilly Media.
[6] Burns, B. (2018). Designing distributed systems: Patterns and paradigms for scalable, reliable services. O'Reilly Media
[7] Wooldridge, M. (2009). An introduction to multiagent systems (2nd ed.). John Wiley & Sons
[8] Russell, S., & Norvig, P. (2021). Artificial intelligence: A modern approach (4th ed.). Pearson

Applied Computer Vision (1st year, 6 ECTS)

Syllabus

1. Fundamentals of image
1.1. Visual perception
1.2. Image formation
1.3. Image representation
2. Image processing and analysis
2.1. Intensity processing
2.2. Histograms
2.3. Filtering in the spatial domain
2.4. Morphological image processing
2.5. Image Processing and filtering in the frequency domain
2.6. Color and multi spectral image processing
3. Image segmentation
3.1. Segmentation fundamentals
3.2. Corner detection
3.3. Region detection
3.4. Video Analysis and background modeling
4. Representation and description
4.1. Description of images and regions
4.2. Description of contours
4.3. Description of keypoints
5. Recognition and segmentation of objects in images and video with ML
5.1. Introduction to Object and Pattern Recognition
5.2. Image Classification
5.3. Object Detection and Segmentation
6. Cameras and Stereo Vision
6.1. Cameras
6.2. Calibration and Homography
6.3. Epipolar geometry and depth

Teaching methodology and assessment:

Teaching methodology:

- Theoretical-practical classes are structured to present the main theoretical concepts of computer vision and image analysis and processing, combining theoretical explanations with practical demonstrations and the resolution of exercises. The concepts are presented interactively, using code examples in languages such as python. During the lessons, code snippets are analysed, allowing students to see in practice how the theories apply to software development adequate to CV and image processing. In addition, exercises are proposed for resolution in class, challenging students to apply the concepts discussed and solving problems that simulate situations found in real systems.
- Practical laboratory classes are dedicated to solving problems related to CV by means of practical worksheets that guide students in solving technical challenges related to programming applied to CV. Each worksheet addresses a set of topics in an applied way according to the syllabus and the learning outcomes. These classes provide an environment where students can experiment directly with concepts of CV and receive immediate feedback from teachers, which makes it easier to correct errors and understand the underlying concepts. Students work individually or in small groups to promote collaboration and the exchange of knowledge.
- Practical projects: programming in CV projects is a crucial part of the methodology, as they allow students to apply the knowledge they have acquired in practical, integrated projects. These projects are carried out in groups, encouraging collaboration and the development of teamwork skills. The projects are divided into phases, starting with the definition of requirements, moving on to the design and implementation of solutions, and ending with the evaluation of results. In addition, code review sessions and project presentations are organised to develop communication and critical evaluation skills.

Assessment: 2 tests, project, comprehension exam

Bibliography

[1] Rafael Gonzalez, Richard Woods, "Digital Image Processing (4rd Edition)", Prentice Hall, 2018
[2] Richard Szeliski, "Computer Vision: Algorithms and Applications, 2nd edition", Springer, 2022 (http://szeliski.org/Book/)
[3] Kaehler, Bradski, "Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library", O'Reilly Media, 2017
[4] Joseph Howse, Joe Minichino, Learning OpenCV 4 Computer Vision with Python 3, 3rd edition, Packt Publishing, 2020

Cybersecurity (1st year, 6 ECTS) – optional prerequisites

Syllabus
1. Cybersecurity Fundamentals
1.1 Information security principles.
1.2 Threat classification and assessment.
1.3 Risk assessment and mitigation.
2. Monitoring and Control Tools
2.1 Firewalls, IDS, and IPS: configuration and analysis.
2.2 Network monitoring and response to suspicious events.
2.3 Penetration testing and vulnerability audits.
3. Cryptography and Data Protection
3.1 Symmetric and asymmetric cryptographic algorithms.
3.2 Authentication protocols and digital signatures.
3.3 Data protection in cloud environments.
4. Compliance and Cybersecurity Regulations

4.1 Relevant regulations: GDPR, NIST, and ISO 27001.
4.2 Compliance implementation methodologies.
4.3 Compliance audits and reports.
5. Incident Response and Defense Measures
5.1 Forensic analysis of cyber incidents.
5.2 Response strategies for distributed attacks.
5.3 Use of automated response tools.

Teaching methodology and assessment:
Teaching methodology:

- Lectures: presentation of fundamental concepts and case studies. Discussion of real attack scenarios.
- Practical Laboratory Classes: configuration and use of tools such as firewalls and IDS. Simulation of attacks and response with reports.
- Practical Projects: development of defense scenarios and incident analysis. Integration of compliance measures into simulated audits.

Assessment components:

- Theory Tests: 40% (20% each) - assess understanding of fundamental concepts.
- Practical Work: 30% - includes attack and response simulations.
- Final Project: 30% - development of an integrated response and audit scenario.

Bibliography
[1] Anderson, R. (2020). Security Engineering: A Guide to Building Dependable Distributed Systems. Wiley.
[2] Schneier, B. (2015). Applied Cryptography: Protocols, Algorithms, and Source Code in C. Wiley.
[3] OWASP Foundation (2024). OWASP Top 10 Documentation. Disponível em: OWASP.
[4] Stallings, W., & Brown, L. (2018). Computer Security: Principles and Practice. Pearson.
[5] NIST. (2022). Cybersecurity Framework. Disponível em: NIST CSF.

Digital Transformation (1st year, 6 ECTS) – optional

Syllabus
1. Fundamentals of digital transformation
1.1 Strategic planning in digital transformation
1.2 Alignment between Digital Transformation and Organizational Objectives
1.3 Introduction to digital transformation maturity models
2. Emerging technologies in digital transformation
2.1 Artificial Intelligence, applications and impact
2.2 IoT and integration with information systems
2.3 Blockchain, smart contracts and secure supply chains
3. Evaluation and improvement of digital processes
3.1 Diagnosis of digital transformation processes
3.2 Identifying opportunities for improvement
3.3 Adapting processes to digital transformation
4. Strategic stakeholder management and digital change
4.1 Organizational culture and digital transformation
4.2 Leadership and team management in the digital context
4.3 Communication and team involvement
5. Case studies and practical projects in digital transformation
5.1 Analyzing real cases

5.2 Developing a practical project

Teaching methodology and assessment:
Teaching methodology:

- Theoretical-practical sessions: these sessions focus on the introduction and exploration of fundamental concepts, such as strategic planning, alignment between informational resources and organizational objectives, and diagnosis of informational processes. Audiovisual materials, interactive presentations, and real case analyses are used to promote understanding of the topics covered.
- Students are encouraged to participate in critical discussions, reflecting on how theoretical concepts can be applied in complex organizational scenarios. Case study analysis: during the sessions, students work in small groups to analyze real or fictional scenarios, identifying problems related to Digital Transformation and proposing solutions based on best practices.
- Applied and collaborative work: students develop a Digital Transformation plan that simulates a real problem of an organization, integrating the concepts of Digital Transformation. This project is carried out in teams, promoting leadership, communication, and time management skills.

Assessment: 2 tests, project, comprehensive exam

Bibliography
[1] Gouveia, L. (2023). Gestão da Informação para Transformação Digital. Paideia. Belo Horizonte: Editora Conhecimento
[2] Gouveia, L. (org.) (2022). Estudos sobre o digital e suas aplicações. Belo Horizonte: Conhecimento Editora
[3] Kane, G. C., Palmer, D., Phillips, A. N., Kiron, D., & Buckley, N. (2019). The Technology Fallacy: How People Are the Real Key to Digital Transformation. MIT Press
[4] Nascimento, J. C. (2022). Sistemas de Informação para Gestores em Tempo de Transformação Digital. Editora Sílabo.
[5] Ribeiro, R., & Veiga, P. (2022). Transformação Digital: Os Desafios, o Pensar e o Fazer. Editora Atual
[6] Rogers, D. L. (2016). The Digital Transformation Playbook: Rethink Your Business for the Digital Age. Columbia Business School Publishing
[7] Siebel, T. M. (2019). Digital Transformation: Survive and Thrive in an Era of Mass Extinction. RosettaBooks
[8] Westerman, G., Bonnet, D., & McAfee, A. (2014). Leading Digital: Turning Technology into Business Transformation. Harvard Business Review Press