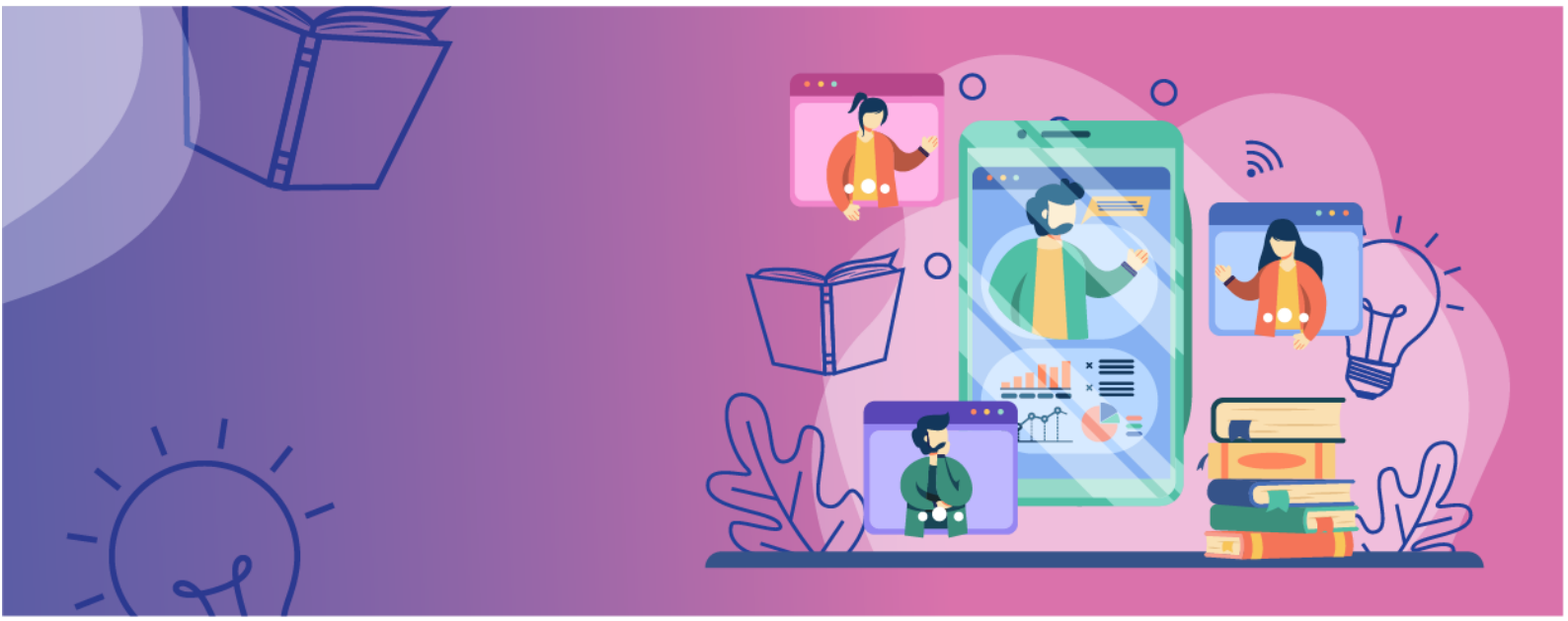
DIGITAL FIRST

Digital Tech as the First Language:

Informatics for Digital Natives

D4.4 Tertiary education programme for informatics teachers



Project 101132761

Co-funded by
the European Union

TABLE OF CONTENTS

Co-funded by the European Union

LIST OF TABLES

Co-funded by the European Union

LIST OF FIGURES

Co-funded by the European Union

1 Introduction

The Digital First project aims to strengthen informatics education across Europe by developing and promoting a functional approach to teaching digital technologies, aligned with the needs of today's digital natives. Within this broader objective, Work Package 4 (WP4), titled *The Picture of the Teacher*, focuses specifically on those who teach informatics in schools. It investigates how teachers are currently prepared for this role, examining the qualifications and training they receive, and evaluating the support structures in place to foster their professional growth and competence development.

Deliverable *D4.4 Tertiary education programme for informatics teachers* presents the results of Task 4.4, which focuses on structuring the formal tertiary teacher education of informatics teachers. The goal of this task is to create the **basic structure for tertiary education of informatics teachers** by describing the 3+2 study programme and its key components, including the course catalogue, core learning outcomes, and school mentoring scheme.

The development of this report builds on the insights and outputs of the previous WP4 deliverables. *D4.1 Informatics teacher education and training in the EU today – research* provided an overview of how informatics teachers are currently educated and trained across Europe, highlighting national approaches, programme structures, and existing gaps. *D4.2 Competencies map for informatics teachers in primary schools* and *D4.3 Competencies map for informatics teachers in secondary schools* then mapped the key competencies required for teaching informatics at primary and secondary levels, integrating subject-specific, pedagogical, and transversal competences. Together, these studies form the foundation for structuring the tertiary education programme.

This report also draws on contributions from the ten partner countries (Bulgaria, Croatia, Cyprus, Finland, Greece, Italy, Lithuania, Portugal, Slovenia, and Spain), offering a cross-country perspective and alignment of the proposed study programme for future informatics teachers. The insights from this deliverable are expected to support the development of high-quality, inclusive, and future-oriented teacher education models that respond to the evolving demands of informatics education in Europe.

Co-funded by
the European Union

2 Programme Design and Description

This chapter provides an overview of the key elements that define the proposed tertiary education programme for future informatics teachers. It outlines the justification for establishing the programme, the internal quality assurance processes guiding its development, the intended learning outcomes and competences, the teaching and learning processes, and the resources required for effective implementation. Together, these sections describe the programme's structure and ensure that it meets the academic, pedagogical, and organisational standards expected of modern teacher education.

2.1 Justification of the Programme

The introduction of a dedicated tertiary education programme for future informatics teachers is justified by a combination of **societal, economic, educational, and institutional factors**, which together highlight the urgent need to modernise and systematise the preparation of teachers in this discipline. This programme responds directly to the evolving role of informatics within European education systems and to the broader mission of higher education institutions to prepare qualified, future-ready professionals capable of supporting digital transformation in schools and society.

Societal and economic needs

Europe is undergoing rapid digital transformation, with digital technologies reshaping how people learn, communicate, work, and participate in society. Children today are immersed in digital environments from an early age. They often grow up as engaging with technology long before they develop literacy in their mother tongue. This creates a fundamental shift in learners' expectations, needs, and ways of interacting with information.

To prepare these students for active, responsible, and creative participation in the digital world, schools require teachers who possess not only strong technical expertise but also a deep understanding of how children in this era learn. The European Union's strategies, including the *Digital Education Action Plan 2021–2027*, and the *Digital Compass*, consistently emphasise the need for high-quality computing education and the development of advanced digital skills among all learners. Achieving these goals is impossible without a robust and well-structured system for educating informatics teachers.

Furthermore, many European countries face a persistent shortage of qualified informatics teachers, which limits the ability of schools to offer high-quality and comprehensive informatics education. This shortage is both structural and long-term, reflecting fragmented qualification pathways, inconsistent national requirements, and limited opportunities for specialised pedagogical preparation. Strengthening tertiary education for informatics teachers is therefore essential to ensure that schools are equipped with professionals who can engage and inspire learners, cultivate their interest in informatics from an early age, and support broader participation in the digital domain.

Pedagogical needs: moving from structural to functional informatics education

Traditional approaches to teaching informatics have tended to emphasise structures, such as programming syntax, hardware components, or isolated digital skills, without sufficiently addressing how these tools can be used meaningfully in everyday contexts. For today's generation of learners, who engage with digital technologies continuously and intuitively, this structural approach is no longer adequate.

The *Digital First* project introduces a functional approach to informatics education, in which teaching focuses on what digital technologies enable learners to do, rather than solely on how digital systems are built. This approach aligns informatics with the logic used in teaching languages (where function and meaning take precedence over structure)

and positions informatics also as a horizontal, cross-curricular competence that empowers students to solve problems, think critically, create, collaborate, and act responsibly in digital environments.

To effectively teach with this functional approach, future informatics teachers require preparation that goes beyond technical competence. They must be equipped with:

- strong pedagogical content knowledge in informatics,
- a deep understanding of learners,
- the ability to integrate informatics meaningfully across subjects,
- inclusive and innovative teaching strategies,
- skills to foster digital citizenship, critical thinking, and ethical awareness.

Such competences cannot be acquired through fragmented or ad-hoc training. They require a **coherent, modern, and research-informed tertiary education programme**.

Alignment with institutional mission and strategic goals

Higher education institutions have a central mission to prepare professionals who can contribute to societal progress, support innovation, and respond to emerging educational challenges. Establishing a dedicated tertiary programme for informatics teachers directly supports these strategic aims by:

- contributing to national and European priorities in digital education,
- enhancing the quality of teacher education through evidence-based curriculum design,
- strengthening links between higher education, schools, and the wider community,
- addressing regional and national shortages of qualified informatics teachers,
- positioning the institution as a leader in shaping future-ready education systems.

By implementing this programme, institutions also reinforce their commitment to inclusive, equitable, and high-quality education. Informatics teachers educated through this programme will be capable of supporting diverse learners and promoting equal opportunities.

2.2 Internal Quality Assurance

The quality assurance of the proposed tertiary education programme is ensured through a combination of institutional procedures and project-level validation. The programme has been developed in alignment with the general quality standards applied within higher education institutions, following established processes for curriculum design, approval, and revision. These procedures guarantee that the structure, course catalogue and learning outcomes of the programme meet national and European expectations for teacher education.

In addition to institutional mechanisms, the Digital First consortium provides an important layer of external peer validation. All partner organisations (representing ten European countries) will review the programme structure, course catalogue, learning outcomes, and pedagogical approach. Their feedback ensures that the programme reflects European best practices, supports the functional approach to teaching informatics, and is adaptable to different national contexts.

2.3 Learning Outcomes and Competences

The intended learning outcomes define the essential knowledge, skills, and competences that graduates must acquire to function as qualified, future-ready informatics teachers in European schools. They provide a coherent framework for curriculum planning, course design, student progression, and programme-level quality assurance.

The outcomes are expressed in a clear and measurable way, ensuring alignment with the European Qualifications Framework (EQF) and with national teacher education standards.

Deliverables *D4.2 Competencies map for informatics teachers in primary school* and *D4.3 Competencies map for informatics teachers in secondary school* identify three major competence domains essential for effective and future-ready informatics teaching:

- **Subject-specific**: competences related to programming, data and information, algorithms, digital systems, cybersecurity, and data ethics, including the application of computational thinking (e.g., decomposition, abstraction, pattern recognition, algorithmic reasoning) to analyse problems, design and refine solutions, and transfer these approaches across contexts.
- **Pedagogical**: competences connected to planning, implementing, and evaluating informatics instruction, assessment strategies, inclusive and differentiated teaching, classroom management, mentoring, and the pedagogical integration of technologies.
- **Transversal**: competences that enable teachers to collaborate, communicate, reflect, act ethically, solve problems creatively, and continuously develop their professional practice.

These domains form the backbone of the intended learning outcomes defined for this tertiary education programme. To ensure coherence and full alignment with D4.2 and D4.3, the learning outcomes are organised into three equally named categories (A-C). Each category contains a set of learning outcomes that describe what students should know, understand, and be able to do after completing the programme.

2.3.1 Bachelor level

A. Subject-specific Learning Outcomes

By the end of the programme, graduates of the Bachelor programme will be able to:

Table 1 Subject-specific learning outcomes of the Bachelor programme

Code	Learning outcomes
A1	Explain and apply core informatics concepts, including data, information, algorithms, programming, and digital systems, as well as fundamental computational thinking principles (e.g., decomposition, pattern recognition, abstraction, and algorithmic reasoning), at a level appropriate for lower secondary education.
A2	Design, implement, and analyse programs and algorithms using structured and object-oriented approaches, basic data structures, and fundamental algorithmic strategies.
A3	Model, represent, and process information using appropriate data representations, encoding methods, databases, and basic statistical techniques, and critically assess data quality, bias, and limitations.
A4	Explain the structure and functioning of computer systems, including hardware, operating systems, networks, and communication systems, using suitable abstractions and school-oriented examples.
A5	Apply principles of cybersecurity, privacy, and digital safety, including responsible data handling, secure system use, and ethical considerations relevant to school contexts.

Code	Learning outcomes
A6	Describe and critically discuss emerging informatics topics, including artificial intelligence, generative AI, modelling and simulation, and physical computing, and relate them to core informatics concepts and societal implications.

B. Pedagogical Learning Outcomes

Graduates of the Bachelor programme will be able to:

Table 2 Pedagogical learning outcomes of the Bachelor programme

Code	Learning outcomes
B1	Interpret informatics curricula and educational frameworks and translate learning outcomes into coherent lesson plans, learning sequences, and classroom activities.
B2	Design and implement age-appropriate informatics instruction, including programming, data, systems, and digital literacy topics, using inquiry-based, problem-based, and project-based approaches, including unplugged and plugged activities and learning designs situated in cross-curricular contexts.
B3	Apply assessment strategies in informatics education, including formative and summative assessment, criteria-based evaluation, and the use of rubrics for programming and project work.
B4	Integrate digital tools, platforms, and learning technologies into teaching in a pedagogically meaningful, safe, accessible, and ethically responsible way.
B5	Reflect on and adapt teaching practices to support inclusive, differentiated, and accessible informatics learning environments, addressing diverse learner needs and classroom contexts.

C. Transversal Learning Outcomes

Graduates of the Bachelor programme will be able to:

Table 3 Transversal learning outcomes of the Bachelor programme

Code	Learning outcomes
C1	Communicate informatics concepts, problem-solving processes, and learning goals clearly and appropriately to learners, colleagues, and other educational stakeholders.
C2	Collaborate effectively in professional teams, engage in reflective practice, and use feedback, observation, and basic educational research methods to improve teaching and learning.
C3	Act in a professionally responsible, ethical, and reflective manner, demonstrating awareness of societal impacts of digital technologies, academic integrity, equity, and continuous professional development.

Bachelor Programme Learning Outcomes Covered by Each Course

The following table presents the alignment between Bachelor programme-level learning outcomes and the individual courses offered within the 3-year study structure. This mapping illustrates constructive alignment across the curriculum by demonstrating how each course supports one or more intended learning outcomes from categories A-C. The table ensures transparency regarding the contribution of each course to the development of the required competences and provides evidence of a systematic progression of learning from foundational to advanced levels throughout the programme.

Table 4 Overview of the Bachelor level learning outcomes based on the courses

Course	A1	A2	A3	A4	A5	A6	B1	B2	B3	B4	B5	C1	C2	C3
Introduction to Informatics	x					x						x		
Introduction to Programming	x	x										x		
Digital Literacy and Learning Tools					x					x				x
Educational Psychology for Teachers											x		x	x
Mathematics for Informatics	x	x												
Digital Creativity and Visual Design						x		x		x		x		
Information Representation and Encoding in Digital Systems	x		x		x									x
Programming 1	x	x										x		
Computer System Architecture	x			x										
Educational Foundations							x				x			x
Human-Computer Interaction						x		x		x		x		
Privacy, Digital Safety and Cybersecurity in Schools					x					x				x
Programming 2	x	x						x				x		
Probability and Statistics	x		x											
Web Technologies		x		x				x		x		x		
Didactics of Informatics I							x	x	x		x		x	
Operating Systems	x			x										
Networks and Communication Systems	x			x	x									

Course	A1	A2	A3	A4	A5	A6	B1	B2	B3	B4	B5	C1	C2	C3
Graph Theory and Automata	x	x												
Robotics and Physical Computing		x		x				x				x		
Pedagogical Methodology									x				x	x
School Practicum I (Observation)											x		x	x
Data Structures and Algorithms	x	x										x		
Introduction to Software Engineering		x				x							x	
Artificial Intelligence and Machine Learning			x			x								x
Didactics of Informatics II							x	x	x	x	x			x
Databases and Information Systems	x		x		x									
Modelling and Simulation			x			x		x				x		
Generative Artificial Intelligence in Education						x				x				x
Inclusive and Accessible Pedagogies								x			x			x
School Practicum II (Teaching Practice)							x	x	x	x	x		x	x

The Bachelor programme learning outcomes are fully aligned with the subject-specific, pedagogical, and transversal competencies for informatics teachers in lower secondary education defined in deliverable *D4.3 Competencies map for informatics teachers in secondary school*.

2.3.2 Master level

A. Subject-specific Learning Outcomes

By the end of the programme, graduates of the Master programme will be able to:

Table 5 Subject-specific learning outcomes of the Master programme

Code	Learning outcomes
A1	Critically analyse and explain advanced informatics concepts, including algorithms, data and data analysis, computing systems, networks, artificial intelligence, cybersecurity, and emerging technologies, using appropriate levels of abstraction for upper secondary education.
A2	Evaluate and select informatics concepts, models, and representations based on their conceptual complexity, epistemological foundations, and suitability for upper secondary learners.

Code	Learning outcomes
A3	Integrate advanced informatics topics with societal, ethical, legal, and sustainability considerations, including issues related to data use, artificial intelligence, security, and digital infrastructures.
A4	Apply informatics knowledge to design, supervise, and evaluate complex student projects, research-oriented tasks, and extended learning activities at upper secondary level.

B. Pedagogical Learning Outcomes

Graduates of the Master programme will be able to:

Table 6 Pedagogical learning outcomes of the Master programme

Code	Learning outcomes
B1	Design, implement, and evaluate coherent sequences of upper secondary informatics lessons that address advanced subject content, curriculum requirements, and long-term learning goals.
B2	Apply advanced didactic strategies, including project-based, inquiry-based, and interdisciplinary approaches, to support deep conceptual understanding, problem solving, and student autonomy.
B3	Design and critically apply assessment and evaluation strategies for complex informatics learning outcomes, including project work, software artefacts, and research-oriented student tasks.
B4	Supervise, mentor, and support diverse learners in extended informatics learning processes, using reflective, inclusive, and ethically grounded pedagogical practices.

C. Transversal Learning Outcomes

Graduates of the Master programme will be able to:

Table 7 Transversal learning outcomes of the Master programme

Code	Learning outcomes
C1	Plan, conduct, and communicate educational research or development work in informatics education, using appropriate methodologies and academically rigorous standards.
C2	Demonstrate professional leadership and collaboration in informatics education, contributing to curriculum development, school innovation, and interdisciplinary cooperation.
C3	Act as reflective, ethical, and adaptive informatics educators, engaging in evidence-based professional development and responding critically to technological, educational, and societal change.

Master Programme Learning Outcomes Covered by Each Course

The following table presents the alignment between Master programme-level learning outcomes and the individual courses offered within the 2-year study structure. This mapping illustrates constructive alignment across the

curriculum by demonstrating how each course supports one or more intended learning outcomes from categories A-C. The table ensures transparency regarding the contribution of each course to the development of the required competences and provides evidence of a systematic progression of learning from foundational to advanced levels throughout the programme.

Table 8 Overview of the Master level learning outcomes based on the courses

Course	A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3
Didactics of Informatics for Upper Secondary		x	x	x	x	x	x	x			x
Cybersecurity	x		x		x	x					x
Advanced Topics in School Informatics	x	x	x			x			x		
Curriculum Design and Educational Policy in Informatics			x		x	x				x	x
Research Methods in Informatics Education		x					x		x		
Cloud Computing	x	x	x		x	x			x		x
School Practicum for Upper Secondary Teaching				x	x	x	x	x	x		x
Advanced Data Analysis for Educational Research	x	x					x		x		
Assessment and Evaluation in Informatics Education							x	x			x
Educational Technology and Digital Pedagogies			x		x	x	x		x		x
Professional Identity and Leadership for Informatics Teachers									x	x	x
Master Seminar		x					x		x	x	
Master Thesis	x	x	x	x			x	x	x	x	x

The Master programme learning outcomes are fully aligned with the subject-specific, pedagogical, and transversal competencies for informatics teachers in upper secondary education defined in deliverable *D4.3 Competencies map for informatics teachers in secondary school*.

2.4 Teaching Process and Student Support

The teaching process within the proposed tertiary education programme for future informatics teachers is designed to ensure high academic quality, promote student-centred learning, and develop the full range of competences required for effective, inclusive, and functional informatics teaching in European schools. This section outlines the admission requirements, teaching and learning methods, student support mechanisms, and how the achievement of learning outcomes is ensured.

2.4.1 Admission Requirements and Criteria

Admission to the bachelor's and master's level programmes follows clear, transparent, and fair criteria that ensure candidates possess the necessary prior knowledge, motivation, and potential to become competent informatics teachers.

Bachelor's Level (3-year programme)

Admission to the bachelor's programme is based primarily on successful **completion of upper secondary education** (ISCED 3) or an equivalent qualification that grants access to higher education in the relevant national system.

Where the number of applicants exceeds available study places, selection will follow transparent, published criteria determined by the university/faculty. Typical selection mechanisms may include one or more of the following:

- grades in upper secondary diploma / final high-school examination (e.g. national leaving exam scores),
- ranked academic score or point system based on secondary-level GPA and subject relevance,
- an institutionally administered entrance exam or test focused on general academic skills or subject-relevant knowledge (where national rules allow),
- recognition of prior learning or portfolio assessment, in cases where applicants present relevant non-formal learning or experience.

The programme will publish detailed admission rules (deadlines, weightings, appeal procedures) well in advance. Where deemed necessary by the faculty and in accordance with national/regional regulations, additional screening (diagnostic tests, short interviews, or submission of a motivation statement) may be used as *optional* instruments to help select candidates, but these are not mandatory prerequisites for admission.

This approach ensures fairness and transparency while giving the hosting institution the flexibility to apply locally appropriate selection measures when demand requires it.

Master's Level (2-year programme)

Admission to the master's programme is open to candidates who have successfully completed a first-cycle degree (bachelor's level) in accordance with the Bologna framework. Because candidates may come from diverse academic backgrounds, the programme adopts a transparent and flexible system of bridging requirements, ensuring that all students enter the master's programme with sufficient informatics and pedagogical foundations.

(a) Graduates of informatics education programmes

Candidates who have completed a bachelor's degree in informatics education (or an equivalent teacher-preparation programme combining informatics and pedagogy) may be admitted directly, without additional requirements. Their prior education is considered fully aligned with the expected entry profile.

(b) Graduates of informatics / computer science programmes

Candidates holding a bachelor's degree in informatics, computer science, software engineering, or related fields may be admitted, provided they demonstrate adequate pedagogical foundations. To ensure balanced preparation for the teaching profession, the faculty may require:

- completion of selected pedagogical and didactic courses (typically 10–30 ECTS, depending on national regulations), depending on the candidate's prior study programme,
- consideration of previously completed certified pedagogical training or relevant courses, which may be recognised through recognition of prior learning procedures.

The exact scope of required bridging courses is determined individually, based on the applicant's documented competencies and national regulations governing teacher qualification pathways.

(c) Graduates of education/pedagogy programmes

Candidates with a bachelor's degree in education, pedagogy, or educational sciences may also be admitted, provided they demonstrate sufficient informatics foundations. To ensure readiness for master's-level study in informatics education, the faculty may require:

- completion of introductory or intermediate informatics courses (typically 30–60 ECTS, depending on national regulations), depending on the extent of previous subject-specific coursework,
- recognition of any informatics-related modules already completed during the bachelor's degree or through non-formal learning, evaluated through recognition of prior learning.

Bridging requirements ensure that candidates acquire the necessary informatics subject knowledge before entering advanced master's-level coursework.

General provisions for all applicants

Across all categories, the programme applies the following principles:

- All admission and bridging requirements are clearly published, transparent, and aligned with national legislation regulating teacher education.
- The faculty determines the individual study plan for each candidate, specifying recognised prior learning and any additional coursework needed.
- Bridging modules (in informatics or pedagogy) must normally be completed before formal enrolment in master's-level courses or within the first academic year, according to institutional rules.

This flexible model ensures that students from different academic backgrounds, whether informatics-oriented or education-oriented, can access the programme while maintaining a consistent and high-quality entry profile for all candidates preparing to become informatics teachers.

2.4.2 Teaching and Learning Methods

The teaching process is based on the principles of **student-centred learning**, active participation, and the integration of theory with authentic classroom practice. It is aligned with the functional approach to informatics promoted by the Digital First project and supports the development of all intended learning outcomes, including subject-specific, pedagogical, and transversal competences.

Student-centred and active learning

The programme emphasises methods that place students at the centre of the learning process and foster autonomy, reflection, collaboration, and creativity:

- Project-based learning and design-based learning related to real classroom tasks.
- Problem-solving sessions focused on the functional application of informatics concepts.
- Microteaching and simulated classroom activities, enabling students to practice teaching scenarios in a safe environment.
- Learning-by-doing, including hands-on work with educational technologies, coding environments, and classroom-ready digital tools.
- Collaborative learning through group projects, peer review, and co-creation of lesson plans.

- Reflective practice, encouraging students to analyse their learning processes and teaching behaviour.
- Inquiry-based learning, supporting deeper conceptual understanding and pedagogical reasoning.

Integration of functional approach with informatics pedagogy

Teaching methods mirror the functional approach by:

- emphasising what learners can do with digital technologies,
- connecting informatics with real-life scenarios, cross-curricular applications, and societal challenges,
- fostering creativity, digital citizenship, ethics, critical thinking, and problem-solving.

School-based learning and mentoring

A crucial part of the teaching process is early and continuous exposure to authentic school environments. Students benefit from:

- progressive teaching practice, expanding from initial observation to full teaching responsibility,
- structured school mentoring, with trained mentors providing guidance and feedback,
- joint reflection sessions between mentors, university staff, and student teachers.

This model ensures that theoretical knowledge is consistently applied and tested in real-life teaching contexts.

2.4.3 Student Support Mechanisms

The programme integrates a comprehensive support system that assists students throughout their studies, both academically and personally. Support structures include:

- Academic Support
  - Tutoring and consultation hours with academic staff.
  - Peer mentoring schemes, especially for first-year students.
  - Supplementary workshops on difficult content areas.
  - Access to digital learning platforms, providing materials, self-assessment tools, and collaborative spaces.
- Pedagogical and Practical Support
  - Supervised teaching practice, supported by qualified school mentors.
  - Feedback cycles based on observation protocols and reflective journals.
  - Workshops on classroom management, inclusion, differentiation, and assessment.
  - Career development advice, including preparation for national teacher certification procedures.
- Wellbeing and Administrative Support
  - Counselling services for psychological, social, and study-related issues.
  - Advisory support for academic planning, recognition of prior learning, and mobility opportunities.
  - Support for students with special educational needs, ensuring inclusive participation.

2.4.4 Ensuring Achievement of Learning Outcomes

The programme ensures the achievement of learning outcomes through a coherent combination of structured coursework, active learning methods, and continuous engagement with authentic school environments. Students progressively develop their competences through carefully sequenced learning activities that integrate theoretical foundations with practical application.

A variety of assessment methods, such as continuous assessment, competence-based tasks, portfolio development, reflective journals, microteaching evaluations, and performance during teaching practice, allow students to demonstrate the full range of expected subject-specific, pedagogical, and transversal competences. These assessment approaches ensure a clear alignment between learning activities, intended outcomes, and the overall pedagogical philosophy of the programme.

Through this holistic and practice-oriented model, graduates become not only technically competent in informatics, but also reflective, adaptable, and innovative educators. They are prepared to design meaningful learning experiences, respond to diverse student needs, and successfully implement the functional approach to informatics education in real classroom contexts.

2.5 Teaching Resources and Infrastructure

The successful implementation of the proposed tertiary education programme for future informatics teachers requires a robust and well-coordinated system of teaching resources, facilities, and institutional support. The higher education institution must ensure that qualified staff, suitable learning environments, digital infrastructure, and financial resources are available to deliver the programme effectively and to support the achievement of the intended learning outcomes.

2.5.1 Academic Staff and Teaching Capacities

The higher education institution guarantees adequate teaching capacities for all components of the bachelor's and master's programmes. This includes:

- Sufficient number of academic staff with expertise in informatics, informatics education, pedagogy, and educational sciences.
- Balanced staff workload allocation, ensuring that teaching, mentoring, and student support responsibilities are manageable and aligned with institutional quality standards.
- Continuous professional development opportunities that enable staff to stay updated with advances in informatics education, digital pedagogy, and the functional approach.

Staff involved in practical training, such as teaching practice supervision, are selected based on their pedagogical experience and their ability to mentor student teachers in authentic school settings.

2.5.2 External Associates and Mentors

The programme may involve external associates such as school mentors, visiting lecturers, and specialists from the fields of informatics, educational technology, and teacher training. Their qualifications and professional experience must be relevant to the programme's objectives.

To ensure quality and coherence:

- External collaborators are selected based on documented expertise, teaching experience, or professional practice in the relevant domain.
- School mentors participating in teaching practice are required to have formal teaching qualifications, experience in mentoring, and competence in student-centred and innovative pedagogical approaches.
- The institution provides guidelines and training for mentors and associates to ensure consistency in mentoring practices and alignment with the programme's learning outcomes.

This ensures that all external contributors support the programme's pedagogical framework and contribute meaningfully to student learning.

2.5.3 Learning Spaces and Physical Infrastructure

The physical infrastructure of the institution supports modern, interactive, and practice-oriented instruction. Students have access to:

- Well-equipped classrooms supporting collaborative and active learning methods.
- Dedicated computer laboratories with up-to-date hardware and software required for teaching informatics.
- Specialised spaces for robotics, educational technologies, media production, and experimental teaching.
- Flexible learning environments, enabling coding workshops, project-based activities, and small-group work.
- Quiet study areas and shared student spaces supporting peer learning and academic collaboration.

All facilities meet safety, accessibility, and inclusion standards, ensuring equal participation for all students, including those with special educational needs.

2.5.4 Digital Infrastructure and Learning Technologies

Given the programme's focus on informatics education, access to relevant digital resources is essential. Students and staff will be supported by:

- Learning management systems for course materials, assessments, collaboration, and communication.
- Software tools, educational platforms, and programming environments aligned with the functional approach to informatics (e.g. problem-solving environments, didactic tools for teaching coding and computation, online simulation tools).
- Institutional access to digital libraries, online databases, and research portals supporting academic work and preparation for teaching practice.
- Hybrid and online teaching capabilities, including virtual classrooms, remote access to labs, and digital submission systems to support flexible learning pathways.

This infrastructure ensures that students develop strong digital competences and become confident users of innovative teaching technologies.

2.5.5 Library Resources and Academic Support Services

The institution provides library resources and services that ensure full access to the literature needed for the programme. This includes:

- Physical and digital collections containing core publications in informatics, computer science education, pedagogy, educational psychology, and the development of teaching and learning materials.
- Subscription-based access to international journals, conference proceedings, and databases relevant to the field.
- Information literacy training, offered by librarians to help students develop research and academic skills.
- Access to additional student services, such as writing centres, tutoring services, academic counselling, and support for scientific work.

Library operations must ensure adequate opening hours, study spaces, and digital access to literature for both on-campus and remote students.

2.5.6 Financial Resources and Programme Sustainability

The higher education institution ensures the availability of the necessary financial resources to organise and sustain the study programme. Financial planning includes:

- Staffing costs for academic and support personnel.
- Maintenance and development of physical and digital infrastructure, including laboratories, learning spaces, and teaching technologies.
- Licensing of specialised software required for informatics education.
- Support for practical training, including coordination with partner schools and mentor training.
- Investment in continuous improvement, ensuring the programme remains aligned with research and with European developments in teacher education.

Sufficient and predictable funding ensures long-term quality, continuity, and the ability of the programme to adapt to new developments in informatics education.

Co-funded by
the European Union

3 Course Catalogue

The course catalogue presents the detailed structure and content of the proposed tertiary education programme for future informatics teachers. It translates the previously defined learning outcomes and competence framework into a coherent set of courses distributed across a 3+2 years study model (Bachelor's and Master's levels).

The catalogue is designed to ensure constructive alignment between programme-level learning outcomes and individual course objectives. Each course contributes explicitly to the development of subject-specific, pedagogical, and transversal learning outcomes and competences described in subsection 2.3. The progression from foundational to advanced topics follows a logical academic and professional trajectory, gradually preparing students for increasing levels of autonomy, responsibility, and pedagogical expertise.

The overall structure of the Bachelor's and Master's study programme, including the distribution of mandatory courses across the clusters Informatics, Pedagogy and Didactics, Informatics Didactics, and Other (mathematics, electives, thesis), is presented in Annex 2, while a corresponding distribution of elective courses is illustrated in Annex 3.

The following subsections first present the structure and detailed course descriptions of the Bachelor's programme, followed by those of the Master's programme.

3.1 Bachelor's Degree (3 years)

The Bachelor's programme in Informatics Education provides the foundational academic and pedagogical preparation required for teaching informatics at primary and lower secondary level (depending on national systems). It integrates core informatics knowledge, educational sciences, didactics of informatics, and structured school practice into a coherent three-year study pathway comprising 180 ECTS credits.

The programme is designed according to a gradual competence-development model. During the first year, students build fundamental subject knowledge and pedagogical foundations. The second year strengthens disciplinary depth while introducing systematic didactic preparation and initial school observation. The third year focuses on advanced informatics topics, independent lesson planning, assessment competence, inclusion strategies, and extended teaching practice. The structure of the Bachelor's study programme, including course names, semesters and ECTS points, is presented in Table 9.

Table 9 Structure of the Bachelor's study programme

Course	Semester	ECTS
Introduction to Informatics	1	5
Introduction to Programming	1	6
Digital Literacy and Learning Tools	1	4
Educational Psychology for Teachers	1	5
Mathematics for Informatics	1	6
Digital Creativity and Visual Design	1	4
Total	*1*	*30*

Information Representation and Encoding in Digital Systems	2	5
Programming 1	2	6
Computer System Architecture	2	5

Educational Foundations	2	5
Human-Computer Interaction	2	4
Privacy, Digital Safety and Cybersecurity in Schools	2	5
Total	*2*	*30*

Programming 2	3	6
Probability and Statistics	3	4
Web Technologies	3	5
Didactics of Informatics I	3	6
Operating Systems	3	5
Elective subject I	3	4
Total	*3*	*30*

Networks and Communication Systems	4	5
Graph Theory and Automata	4	5
Robotics and Physical Computing	4	5
Pedagogical Methodology	4	5
School Practicum I (Observation)	4	6
Elective subject II	4	4
Total	*4*	*30*

Data Structures and Algorithms	5	6
Introduction to Software Engineering	5	5
Artificial Intelligence and Machine Learning	5	5
Didactics of Informatics II	5	5
Elective subject III	5	4
Total	*5*	*25*[1]

Databases and Information Systems	6	5
Modelling and Simulation	6	5
Generative Artificial Intelligence in Education	6	4
Inclusive and Accessible Pedagogies	6	5
School Practicum II (Teaching Practice)	5+6	12[2]
Elective subject IV	6	4
Total	*6*	*35*[3]

[1] + 5 ECTS for *School Practicum II (Teaching Practice).*

[2] 5 ECTS in semester 5 and 7 ECTS in semester 6, together 12 ECTS in year 3.

[3] - 5 ECTS for *School Practicum II (Teaching Practice).*

All	180

The structure presented above ensures a balanced distribution of subject-specific, pedagogical, and practical components throughout the three years of study.

The first year establishes strong conceptual foundations in informatics (programming, systems, data representation), mathematics, and educational sciences. This phase ensures that students develop both disciplinary understanding and awareness of learning processes before entering more specialised didactic training.

In the second year, the programme deepens technical knowledge while introducing explicit didactic preparation through the courses *Didactics of Informatics I* and *Pedagogical Methodology*. Importantly, *School Practicum I (Observation)* allows students to connect theoretical knowledge with authentic classroom practice in a guided and reflective manner.

The third year marks the transition toward professional autonomy. Students engage with advanced informatics topics such as data structures, artificial intelligence, databases, and modelling, while simultaneously developing assessment competence, inclusive teaching strategies, and classroom management skills. *School Practicum II (Teaching Practice)*, implemented as a year-long course, ensures sustained engagement in real teaching contexts and supports the development of professional identity.

Elective courses distributed across semesters allow students to select topics aligned with their professional interests, while maintaining coherence with the overall competence framework defined within the Digital First project.

3.1.1 Year 1

Semester 1

In the first semester of the first year of the undergraduate study programme, students will attend the following mandatory courses: *Introduction to Informatics*, *Introduction to Programming*, *Digital Literacy and Learning Tools*, *Educational Psychology for Teachers*, *Mathematics for Informatics*, and *Digital Creativity and Visual Design*. More detailed descriptions of the courses are provided in the following tables.

Table 10 Description of the course Introduction to Informatics

Course title	Introduction to Informatics
Year of study	1
ECTS	5
Course aims	Introduce students to fundamental concepts of informatics, computational thinking, and problem-solving.
Enrolment requirements	There is none.
Learning outcomes	• Explain core concepts of informatics, including data, information, algorithms, computation, and systems. • Describe basic principles of algorithmic problem solving and computational thinking. • Apply logical reasoning using Boolean values and simple decision structures. • Distinguish between informatics as a discipline and the use of digital technologies. • Interpret fundamental informatics concepts in everyday contexts.

Course content	○ Basic notions of data, information, systems, and computation ○ Informatics as a scientific discipline (informatics vs. ICT use) ○ Information and data: basic notions, representation, and meaning ○ Algorithms and problem solving: sequence, decision, repetition ○ Logical foundations of computing: Boolean values and basic logical operators ○ Programs and computing systems: programs as instructions; input–process–output model

Table 11 Description of the course Introduction to Programming

Course title	Introduction to Programming
Year of study	1
ECTS	6
Course aims	Develop foundational programming knowledge and skills using a high-level programming language (e.g. Python), with emphasis on algorithmic thinking.
Enrolment requirements	There is none.
Learning outcomes	• Use basic control structures (sequence, selection, repetition). • Implement simple algorithms. • Decompose problems into smaller parts and implement them using subprograms. • Debug simple programs using structured reasoning. • Apply algorithmic thinking to everyday problems.
Course content	○ Programs: syntax and semantics ○ Values and variables, basic data types ○ Control structures: sequence, selection, repetition ○ Basic input and output ○ Simple data structures: lists / arrays ○ Typical programming errors and error handling ○ Introductory debugging through examples ○ Simple algorithms and decomposition into subprograms

Table 12 Description of the course Digital Literacy and Learning Tools

Course title	Digital Literacy and Learning Tools
Year of study	1
ECTS	4
Course aims	Develop understanding of digital technologies in society and competence in the safe, responsible, and effective use of digital tools and resources.
Enrolment requirements	There is none.
Learning outcomes	• Describe key digital technologies and their societal impact. • Use basic digital tools for inquiry, communication and creation. • Use digital tools responsibly. • Evaluate ICT tools for instruction. • Understand digital safety.

	• Apply ethical and responsible use of digital technologies.
Course content	○ Digital technologies in society (communication, automation, creativity) ○ Digital identity ○ Productivity tools ○ Learning management systems ○ Online collaboration tools ○ Media literacy ○ Digital citizenship ○ Privacy and data protection ○ Evaluating digital resources

Table 13 Description of the course Educational Psychology for Teachers

Course title	Educational Psychology for Teachers
Year of study	1
ECTS	5
Course aims	Introduce key theories and concepts of educational psychology relevant to teaching and learning.
Enrolment requirements	There is none.
Learning outcomes	• Explain fundamental theories of learning and motivation. • Apply basic motivational strategies in educational contexts. • Understand key aspects of learners' cognitive development and individual differences, including the role of interaction, activity, and context, and individual differences.
Course content	○ Cognitive development ○ Learning theories (behaviourism, constructivism) ○ Motivation theories ○ Classroom behaviour and climate ○ Individual differences ○ Cognitive load theory ○ Memory and learning ○ Growth mindset ○ Teacher-student communication ○ Contemporary perspectives on learning: situated and embodied cognition

Table 14 Description of the course Mathematics for Informatics

Course title	Mathematics for Informatics
Year of study	1
ECTS	6
Course aims	Provide foundational mathematics for informatics education.
Enrolment requirements	There is none.
Learning outcomes	• Apply logic, sets, relations, functions to problems.

	• Use discrete structures in algorithm development. • Explain combinatorics and reasoning patterns relevant for computing.
Course content	○ Set theory ○ Logic and proofs ○ Functions and relations ○ Fundamentals of combinatorics ○ Fundamentals of graph theory ○ Mathematical reasoning and abstraction ○ Discrete structures as foundations of algorithms

Table 15 Description of the course Digital Creativity and Visual Design

Course title	Digital Creativity and Visual Design
Year of study	1
ECTS	4
Course aims	Develop students' creative and visual expression using digital tools and to enable them to design visually meaningful digital artefacts.
Enrolment requirements	There is none.
Learning outcomes	• Apply basic principles of visual design (contrast, alignment, hierarchy, colour, typography). • Create simple digital artefacts that combine text, image, audio, and animation. • Use digital creativity tools that are appropriate for school and classroom environments. • Explain the role of creativity and visual literacy within informatics education.
Course content	○ Digital drawing and image editing ○ Audio creation and simple animation ○ Principles of visual and interaction design ○ Visual storytelling and information design ○ Creative process and basic digital prototyping (from idea to digital artefact) ○ Creative computing tools for school environments ○ Copyright, licensing, and reuse of digital content

Semester 2

In the second semester of the first year of the undergraduate study programme, students will attend the following mandatory courses: *Information Representation and Encoding in Digital Systems*, *Programming 1*, *Computer System Architecture*, *Educational Foundations*, *Human-Computer Interaction*, and *Privacy, Digital Safety and Cybersecurity in Schools*. More detailed descriptions of the courses are provided in the following tables.

Table 16 Description of the course Information Representation and Encoding in Digital Systems

Course title	Information Representation and Encoding in Digital Systems
Year of study	1
ECTS	5

Course aims	Understanding of how information is represented, encoded and processed in digital systems, including ethical and social aspects.
Enrolment requirements	There is none.
Learning outcomes	• Explain number systems and encoding of text, images, sound. • Describe data storage, compression and transmission principles. • Evaluate accuracy, bias and ethical implications of information. • Apply basic forms of information representation to support classroom-based learning activities.
Course content	○ Number systems (binary, hexadecimal) ○ Representation and encoding of text, images, and sound ○ Fundamentals of data storage and data compression ○ Limits of representation and information loss ○ Information accuracy, bias, and reliability ○ Ethical aspects of information representation and use of digital information ○ Information reliability and misinformation in digital media

Table 17 Description of the course Programming 1

Course title	**Programming 1**
Year of study	1
ECTS	6
Course aims	Develop foundational programming skills with emphasis on structured program design, use of basic data structures, and code readability and clarity.
Enrolment requirements	There is none.
Learning outcomes	• Design structured programs using functions and procedures. • Implement algorithms in a high-level programming language. • Use strings, lists, sets, dictionaries, and other data structures. • Apply systematic debugging techniques.
Course content	○ Functions and procedures ○ Lists, strings, sets, dictionary, and other data structures ○ Introduction to object-oriented programming ○ Testing and debugging strategies ○ Problem decomposition and stepwise refinement ○ Code readability and basic documentation

Table 18 Description of the course Computer System Architecture

Course title	**Computer System Architecture**
Year of study	1
ECTS	5
Course aims	Understand how digital systems are structured, how they function, and how information flows.
Enrolment requirements	There is none.

Learning outcomes	• Understand the basic architecture of computer systems at different levels of abstraction. • Describe how information flows and is processed within a computer system. • Recognise the role of abstraction and architectural models (e.g. Von Neumann) in understanding computer systems. • Explain the cooperation between the CPU, memory, and input/output subsystems. • Identify the key components and basic functioning of simple microcontroller systems.
Course content	○ Computer systems and their basic organization ○ Components of computer systems ○ Information flow and processing ○ Von Neumann model ○ CPU, memory, and input/output cooperation ○ Abstraction in computer systems ○ Introduction to microcontrollers

Table 19 Description of the course Educational Foundations

Course title	Educational Foundations
Year of study	1
ECTS	5
Course aims	Introduce fundamental educational concepts for understanding teaching and learning, with emphasis on learning environments and activities relevant to informatics education.
Enrolment requirements	There is none.
Learning outcomes	• Describe key educational and pedagogical fundamentals of formal schooling. • Explain the role of curriculum, learning objectives, and educational frameworks. • Analyse how learning environments and technology shape educational practices. • Explain the roles and responsibilities of key educational stakeholders (including parents). • Recognize basic principles of inclusive education and support for learners with special educational needs.
Course content	○ Education as a system: aims of schooling and the role of teachers ○ Basic concepts and pedagogical foundations of education ○ Curriculum frameworks, learning objectives, and learning outcomes ○ Learning environments: physical, digital, and hybrid ○ Technology-enhanced educational environments ○ Education in relation to the individual and society ○ Contemporary educational goals: computational thinking and emerging digital literacies ○ Educational stakeholders (students, teachers, parents, school leadership, support staff, policymakers), their roles, and collaboration in schooling

	○ Inclusive education and basic principles of supporting learners with special educational needs

Table 20 Description of the course Human-Computer Interaction

Course title	Human-Computer Interaction
Year of study	1
ECTS	4
Course aims	Introduce principles of human-computer interaction (HCI), usability, and accessibility with a focus on designing and evaluating digital tools for learning and teaching.
Enrolment requirements	There is none.
Learning outcomes	• Apply basic usability and accessibility principles. • Design simple, user-friendly digital interfaces. • Evaluate educational digital tools with regard to usability, accessibility, and learner experience • Explain how interface design influences learning, engagement, and inclusion.
Course content	○ Human-centred design principles ○ Usability heuristics ○ Accessibility basics and inclusive interface design ○ User interface components ○ Interaction patterns ○ Prototyping simple digital interfaces ○ Evaluation of educational software and digital learning tools ○ Examples of HCI in learning technologies

Table 21 Description of the course Privacy, Digital Safety and Cybersecurity in Schools

Course title	Privacy, Digital Safety and Cybersecurity in Schools
Year of study	1
ECTS	5
Course aims	Equip future teachers with knowledge and skills to ensure digital safety, privacy, and responsible use of digital technologies in school environments.
Enrolment requirements	There is none.
Learning outcomes	• Identify common digital risks affecting students, teachers, and school stakeholders, and explain their potential impact on learning. • Apply basic cybersecurity and safe-use principles in educational contexts. • Explain data protection, privacy, and ethical principles. • Respond appropriately to common online safety and digital misconduct incidents.
Course content	○ Online risks and threats ○ Cybersecurity fundamentals ○ Data protection, privacy, and GDPR principles ○ Passwords, authentication, and safe digital practices

	○ Social media, digital identity, and responsible online behaviour ○ Risk assessment in schools ○ Preventive and responsive strategies

3.1.2 Year 2

Semester 1

In the first semester of the second year of the undergraduate study programme, students will attend the following mandatory courses: *Programming 2*, *Probability and Statistics*, *Web Technologies*, *Didactics of Informatics I*, and *Operating Systems*. Students will also choose one elective subject. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 22 Description of the course Programming 2

Course title	Programming 2
Year of study	2
ECTS	6
Course aims	Strengthen programming competence through more complex problem-solving tasks, data processing, and structured program design.
Enrolment requirements	Completed courses *Introduction to Programming* and *Programming 1*.
Learning outcomes	• Design medium-sized programs using structured decomposition and clear code organization. • Work confidently with compound data structures. • Use recursion appropriately. • Apply programming to simple data processing and simulations. • Read, understand, and modify existing code. • Explain programming solutions in pedagogically meaningful ways.
Course content	○ Advanced use of functions and modular program design ○ Recursion (programming perspective) ○ Nested and compound data structures ○ File input and output ○ Data processing, analysis and simulations ○ Introduction to object-oriented design ○ Code readability, refactoring, and reuse ○ Debugging strategies for medium-sized programs ○ Pedagogical discussion: common student misconceptions

Table 23 Description of the course Probability and Statistics

Course title	Probability and Statistics
Year of study	2
ECTS	4
Course aims	Provide foundational knowledge of probability and statistics necessary for informatics, data reasoning, and teaching data-related topics in school curricula.
Enrolment requirements	Completed course *Mathematics for Computer Science*.

Learning outcomes	• Apply basic probability concepts to simple computational and real-world problems. • Interpret and analyse data using elementary statistical measures. • Explain randomness, uncertainty, and variability in an accessible way. • Use digital tools to collect, visualise, and analyse data in educational and exploratory contexts.
Course content	○ Probability basics (events, outcomes, probability rules) ○ Discrete probability and simulations ○ Random variables and simple distributions ○ Descriptive statistics (mean, median, variance, standard deviation) ○ Data collection and sampling ○ Data visualisation (charts, plots) ○ Interpretation and misuse of statistics

Table 24 Description of the course Web Technologies

Course title	Web Technologies
Year of study	2
ECTS	5
Course aims	Develop knowledge and skills for designing and implementing basic web applications, with attention to accessibility, usability, and responsible web practices.
Enrolment requirements	Completed course *Human-Computer Interaction*.
Learning outcomes	• Design and create simple, functional websites following basic web standards. • Explain the basics of client-server interactions and web communication. • Use basic web technologies to develop simple interactive web content. • Apply basic principles of responsive design and web accessibility. • Explain core web concepts at a level appropriate for school teaching.
Course content	○ HTML and CSS fundamentals ○ JavaScript basics ○ Document Object Model (DOM) and simple interactivity ○ Client–server model and HTTP protocol ○ Web hosting and deployment basics ○ Responsive web design principles ○ Web accessibility ○ Introduction to back-end concepts ○ Basics of web APIs ○ Web security and safe development practices

Table 25 Description of the course Didactics of Informatics I

Course title	Didactics of Informatics I
Year of study	2
ECTS	6

Course aims	Prepare students to plan, design, and implement informatics teaching by introducing core didactic approaches, curriculum interpretation, and age-appropriate learning designs.
Enrolment requirements	Completed courses *Educational Psychology for Teachers* and *Educational Foundations*.
Learning outcomes	• Distinguish between structural and functional perspectives of teaching informatics. • Interpret informatics curricula and learning outcomes • Design lesson plans addressing programming, data, and computing systems concepts. • Apply inquiry-based, problem-based, and project-based approaches. • Design and use unplugged and plugged activities to support understanding of informatics concepts. • Select age-appropriate tools and resources for teaching informatics concepts. • Apply knowledge taxonomies when formulating learning objectives in informatics lessons.
Course content	○ Structural and functional approaches to teaching informatics ○ Informatics curriculum analysis and interpretation ○ Learning objectives and knowledge taxonomies in informatics education ○ Designing learning activities for core informatics topics: programming, data, and systems ○ Designing unplugged and plugged activities for teaching informatics concepts ○ Inquiry-based and problem-based learning in informatics ○ Project-based learning as an introductory approach ○ Classroom organization and management in informatics ○ Overview of educational technologies and tools commonly used in informatics education

Table 26 Description of the course Operating Systems

Course title	**Operating Systems**
Year of study	2
ECTS	5
Course aims	Understand how operating systems manage hardware resources and support applications and relate these concepts to school-level informatics education.
Enrolment requirements	Completed course *Computer System Architecture*.
Learning outcomes	• Explain core operating system functions, including process, memory, file, and device management. • Use basic system tools to explore and understand system behaviour in learning tasks. • Illustrate operating system concepts using examples appropriate for school-level informatics education.
Course content	○ Operating system architecture and core components

	○ Process management and basic scheduling concepts ○ Memory management fundamentals ○ File systems and storage organization ○ Command-line basics and system interaction ○ Operating system–level security concepts ○ School-oriented examples and platforms (e.g. Micro:bit, Raspberry Pi)

Semester 2

In the second semester of the second year of the undergraduate study programme, students will attend the following mandatory courses: *Networks and Communication Systems*, *Graph Theory and Automata*, *Robotics and Physical Computing*, *Pedagogical Methodology*, and *School Practicum I (Observation)*. Students will also choose one elective subject. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 27 Description of the course Networks and Communication Systems

Course title	Networks and Communication Systems
Year of study	2
ECTS	5
Course aims	Introduce principals of computer networks relevant for school curricula.
Enrolment requirements	Completed course *Computer System Architecture*.
Learning outcomes	• Explain the basic structure of the internet and principles of digital communication. • Demonstrate and interpret simple networking concepts and configurations. • Address network security, privacy, and safe online behaviour.
Course content	○ Network concepts and types (LAN, WAN, internet) ○ Basic communication models and data flow in networks ○ Layered communication models (ISO/OSI as a conceptual reference) ○ Internet architecture and core services ○ Client-server model and the role of servers in networked systems ○ Network protocols and their roles within the TCP/IP model (e.g. DNS, HTTPS, FTP) ○ Packet-based communication and encapsulation concepts ○ IP addressing and basic routing concepts ○ Wired and wireless networks ○ Network services and everyday applications ○ Basic principles of network security and data protection ○ Conceptual troubleshooting of network-related problems

Table 28 Description of the course Graph Theory and Automata

Course title	Graph Theory and Automata
Year of study	2
ECTS	5
Course aims	Deepen students' understanding of the mathematical foundations of algorithms, with emphasis on graph-based models, path-finding algorithms, optimisation problems, and formal models of computation (automata theory).
Enrolment requirements	Completed course *Mathematics for Computer Science*.

Learning outcomes	• Apply graph-theoretical concepts to model and analyse algorithmic problems. • Explain and analyse shortest-path and related graph algorithms. • Use mathematical reasoning to compare algorithmic approaches. • Understand optimisation problems from a structural and conceptual perspective. • Explain the basic ideas of automata theory as formal models of computation.
Course content	○ Review of graph theory (graphs, trees, paths, cycles, connectivity) ○ Weighted graphs and path-related problems ○ Shortest-path algorithms (Floyd-Warshall, Dijkstra – conceptual and applied) ○ Graph-based optimisation problems ○ Matrix-based representations of graphs ○ Introduction to finite automata ○ Deterministic finite automata (DFA) and state transitions ○ Automata as formal models of computation and system behaviour

Table 29 Description of the course Robotics and Physical Computing

Course title	Robotics and Physical Computing
Year of study	2
ECTS	5
Course aims	Develop competence in robotics and physical computing as tools for teaching informatics concepts through hands-on, inquiry-based learning.
Enrolment requirements	Completed course *Computer System Architecture*.
Learning outcomes	• Use sensors and actuators in simple physical computing and robotics projects. • Program microcontrollers using age-appropriate block-based environments. • Design classroom activities involving robotics and physical computing. • Explain core concepts such as input/output, control, events, and feedback. • Evaluate the pedagogical value and limitations of robotics and physical computing.
Course content	○ Sensors and actuators ○ Microcontrollers and embedded systems ○ Block-based robotics and physical computing environments ○ Introduction to microcontroller-based platforms (e.g. Arduino, Micro:bit) ○ Control structures and event-driven programming ○ Classroom robotics kits ○ Project-based learning with physical computing

Table 30 Description of the course Pedagogical Methodology

Course title	Pedagogical Methodology
Year of study	2
ECTS	5
Course aims	Develop the ability to plan, conduct, and critically interpret educational research.
Enrolment requirements	There is none.
Learning outcomes	• Formulate clear research problems, aims, and questions.

	• Design small-scale educational studies using appropriate quantitative, qualitative, or mixed research approaches. • Select and apply suitable data collection techniques and basic data analysis procedures. • Evaluate validity, reliability, and ethical aspects of educational research. • Use digital tools and platforms to support data collection, analysis, and self-evaluation research. • Interpret and communicate research findings in a clear and academically appropriate manner.
Course content	○ Foundations of educational research: purposes, types, and research approaches (quantitative, qualitative, mixed). ○ Educational research designs: descriptive, causal-experimental, and causal non-experimental. ○ Research process: problem definition, research questions, hypotheses, literature review, data collection, and interpretation. ○ Data collection methods: surveys, scales, interviews, observation, sociometric techniques, and knowledge tests. ○ Basics of data analysis: data organisation, graphical presentation, descriptive statistics, correlation, and interpretation of statistical significance. ○ Digital tools for educational research: computer-based data analysis, online platforms for data collection and self-evaluation. ○ Academic reporting, research ethics, and systematic literature review using online databases

Table 31 Description of the course School Practicum I (Observation)

Course title	**School Practicum I (Observation)**
Year of study	2
ECTS	6
Course aims	Introduce students to real classroom environments through guided observation, with a focus on teaching practices, learner engagement, and the use of digital technologies in school contexts.
Enrolment requirements	Completed course *Didactics of Informatics I.*
Learning outcomes	• Observe and critically reflect on teaching and learning processes in school settings. • Identify effective teaching strategies, classroom management approaches, and forms of student engagement. • Recognise the role of digital technologies in classroom practice. • Analyse lesson structure, communication patterns, and assessment practices. • Develop reflective skills through structured observation and guided mentoring discussions.
Course content	○ Classroom observation ○ Roles and responsibilities of the teacher ○ Student behaviour, interaction, and engagement

	○ Lesson structure and organisation ○ Assessment practices ○ Teacher-student communication ○ Inclusivity and support for diverse learners in practice ○ Use of digital technologies in classrooms ○ Mentorship discussions and guided reflection ○ Reflective diary writing

3.1.3 Year 3

Semester 1

In the first semester of the third year of the undergraduate study programme, students will attend the following mandatory courses: *Data Structures and Algorithms*, *Introduction to Software Engineering*, *Artificial Intelligence and Machine Learning*, and *Didactics of Informatics II*. There is also one whole year mandatory course *School Practicum II (Teaching Practice)* that will be described among the second semester courses. Students will also choose one elective subject. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 32 Description of the course Data Structures and Algorithms

Course title	**Data Structures and Algorithms**
Year of study	3
ECTS	6
Course aims	Develop a solid understanding of fundamental data structures and algorithms, with emphasis on problem solving and algorithmic efficiency.
Enrolment requirements	Completed course *Programming 2*.
Learning outcomes	• Analyse algorithm efficiency using basic time and space complexity measures. • Implement and compare core data structures and algorithms. • Select appropriate data structures and algorithms for given computational problems. • Explain algorithmic solutions, design choices, and trade-offs at an appropriate level of abstraction.
Course content	○ Algorithmic complexity (time and space) ○ Recursion (algorithmic perspective) ○ Searching and sorting algorithms ○ Linear data structures: linked lists, stacks, queues ○ Trees and tree traversals ○ Graph representations and basic graph algorithms ○ Hash tables ○ Algorithm design strategies ○ Algorithmic problem solving

Table 33 Description of the course Introduction to Software Engineering

Course title	**Introduction to Software Engineering**
Year of study	3

ECTS	5
Course aims	Develop foundational understanding of software development processes, collaboration, and code quality for educational and small-scale projects.
Enrolment requirements	Completed course *Programming 2*.
Learning outcomes	• Apply basic software engineering principles in small-scale projects. • Use version control systems to support collaborative development. • Describe key phases of the software development life cycle. • Apply fundamental testing and refactoring practices to improve code quality. • Explain software development concepts in forms suitable for school-level teaching.
Course content	○ Software development life cycle and development models ○ Requirements analysis (introductory, practical focus) ○ Unified Modeling Language (UML) basics and simple modelling ○ Version control systems (Git): basic workflows ○ Testing fundamentals (conceptual and practical) ○ Collaborative development workflows ○ Agile principles (introductory level) ○ Documentation techniques ○ Refactoring basics ○ Small-scale team project implementation

Table 34 Description of the course Artificial Intelligence and Machine Learning

Course title	**Artificial Intelligence and Machine Learning**
Year of study	3
ECTS	5
Course aims	Provide foundational knowledge of machine learning and artificial intelligence concepts.
Enrolment requirements	There is none.
Learning outcomes	• Explain core concepts of artificial intelligence and machine learning, including basic problem-solving and search approaches. • Distinguish between supervised, unsupervised, and reinforcement learning. • Describe the role of data preparation and data quality in machine learning. • Explain basic machine learning models and their applications. • Interpret model behaviour using basic evaluation concepts. • Recognise overfitting, underfitting, bias, and fairness issues. • Discuss ethical and societal aspects of artificial intelligence. • Design simple, educationally appropriate machine learning examples.
Course content	○ Artificial intelligence and machine learning: core concepts and scope ○ Problem solving and search in artificial intelligence ○ Problem formulation and state space ○ Uninformed and informed search (conceptual overview) ○ Data preparation and data quality

	○ Learning paradigms: supervised, unsupervised, and reinforcement learning (conceptual overview) ○ Classification and regression ○ Clustering ○ Decision trees ○ Neural networks (conceptual level) ○ Model evaluation and limitations ○ Overfitting and underfitting ○ Bias, fairness, and ethical artificial intelligence ○ Simple machine learning projects and educational examples

Table 35 Description of the course Didactics of Informatics II

Course title	Didactics of Informatics II
Year of study	3
ECTS	5
Course aims	Prepare students for independent teaching of informatics at lower secondary education level (ISCED 24), with emphasis on coherent lesson sequencing, assessment, and differentiation.
Enrolment requirements	Completed courses *Didactics of Informatics I* and *School Practicum I (Observation)*.
Learning outcomes	• Plan and teach informatics lessons independently. • Design and apply formative and summative assessment strategies in informatics. • Design coherent sequences of informatics lessons and learning units. • Adapt instruction and learning activities for diverse learners. • Use criteria and rubrics to evaluate student work and projects. • Apply digital tools responsibly and ethically in informatics teaching. • Reflect on ethical, legal, and societal issues related to teaching informatics.
Course content	○ Curriculum mapping and vertical alignment of informatics topics ○ Designing sequences of lessons and learning units ○ Teaching debugging and supporting problem-solving processes ○ Managing group work and collaborative learning ○ Responsible and ethical integration of digital tools ○ Assessment in informatics: formative and summative approaches ○ Criteria-based assessment and rubric design ○ Evaluating student projects and artefacts ○ Differentiation and inclusive strategies in informatics teaching ○ Ethical, legal, and societal issues in informatics education

Semester 2

In the second semester of the third year of the undergraduate study programme, students will attend the following mandatory courses: *Databases and Information Systems*, *Modelling and Simulation*, *Generative Artificial Intelligence in Education*, *Inclusive and Accessible Pedagogies*, and *School Practicum II (Teaching Practice)*. Students will also

choose one elective subject. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 36 Description of the course Databases and Information Systems

Course title	Databases and Information Systems
Year of study	3
ECTS	5
Course aims	Introduce relational databases and fundamental principles of data organisation and management.
Enrolment requirements	There is none.
Learning outcomes	• Explain fundamental database concepts and principles. • Design basic relational databases. • Use SQL to query and manage data. • Apply database concepts in educational and real-world information systems. • Recognise basic principles of data modelling, data integrity, and responsible data use.
Course content	○ Data modelling ○ Tables, keys, relationships ○ Relational model ○ Entity relationship (ER) diagrams ○ SQL basics (SELECT, JOIN, filtering) ○ Database design ○ Backup and recovery ○ Normalization principles ○ Indexes and query performance (conceptual) ○ Transactions and data consistency (conceptual) ○ Basic concepts of information systems (data, processes, users) ○ Data visualisation ○ Data security ○ Ethics of data handling

Table 37 Description of the course Modelling and Simulation

Course title	Modelling and Simulation
Year of study	3
ECTS	5
Course aims	Enable students to use conceptual computational models and simulations to support understanding of complex systems, algorithmic processes, and data-driven representations in informatics and interdisciplinary school contexts.
Enrolment requirements	There is none.
Learning outcomes	• Explain the role of modelling and simulation in informatics. • Use simple modelling tools to represent key abstractions of systems and processes.

	• Interpret results of simulations critically including model assumptions, limitations, and data patterns. • Design classroom activities using simulations to support algorithmic reasoning and data literacy. • Connect modelling approaches to school curricula.
Course content	○ Introduction to computational modelling ○ System dynamics ○ Agent-based models ○ Simulations in informatics and interdisciplinary education ○ Using existing simulation tools ○ Designing educational modelling activities ○ Limitations and assumptions of models

Table 38 Description of the course Generative Artificial Intelligence in Education

Course title	Generative Artificial Intelligence in Education
Year of study	3
ECTS	4
Course aims	Introduce core concepts of generative artificial intelligence and prepare future informatics teachers to understand, critically use, and teach generative AI in educational contexts in a responsible, ethical, and pedagogically grounded manner.
Enrolment requirements	There is none.
Learning outcomes	• Explain the basic principles behind generative artificial intelligence systems. • Use generative AI systems in controlled educational contexts to explore their behaviour and limitations. • Relate generative AI use to core informatics concepts, including data and algorithmic literacy. • Design an educational conversational agent using given data and defined constraints. • Identify ethical risks, limitations, and academic integrity issues related to generative AI use. • Promote critical, responsible, and transparent use of generative AI in informatics education.
Course content	○ Foundations of generative artificial intelligence ○ Data-driven nature and limitations of generative AI systems ○ Generative AI as part of data and algorithmic literacy ○ Conceptual overview of generative AI models ○ Generative models beyond text (overview) ○ Human-AI interaction and prompt design ○ Pedagogical implications of prompt-based interaction ○ Design and modelling of an educational chat agent using given data ○ Risks, limitations, and academic integrity ○ Ethical, legal, and professional responsibility of informatics teachers

Table 39 Description of the course Inclusive and Accessible Pedagogies

Course title	Inclusive and Accessible Pedagogies
Year of study	3
ECTS	5
Course aims	Prepare future teachers to design inclusive, accessible, and equitable informatics learning environments supported by digital technologies.
Enrolment requirements	There is none.
Learning outcomes	• Apply principles of inclusive and accessible design. • Adapt informatics instruction for diverse learners. • Select and use assistive and adaptive technologies in educational contexts. • Design learning activities following Universal Design for Learning (UDL). • Reflect critically on equity, accessibility, and participation in informatics education.
Course content	○ Universal Design for Learning (UDL) ○ Accessibility standards and guidelines ○ Differentiation strategies in informatics ○ Assistive and adaptive technologies ○ Inclusive classroom practices ○ Gender, socio-economic, and cultural equity in informatics ○ Case studies and examples from school practice

Table 40 Description of the course School Practicum II (Teaching Practice)

Course title	School Practicum II (Teaching Practice)
Year of study	3
ECTS	12 (whole year course)
Course aims	Develop practical teaching competence through independent classroom teaching.
Enrolment requirements	Completed courses *Didactics of Informatics I* and *School Practicum I (Observation)*.
Learning outcomes	• Design, implement, and evaluate lesson structure, communication patterns, and assessment practices in own teaching. • Implement effective teaching strategies, classroom management approaches, and forms of student engagement in own teaching practice. • Use digital technologies in classroom practice. • Develop reflective skills through structured reflection and guided mentoring discussions.
Course content	○ Lesson planning and evaluation ○ Micro-teaching ○ Classroom teaching (min. 6–8 weeks) ○ Feedback cycles ○ Assessment implementation ○ Managing real classrooms ○ Use of digital tools

	○ Collaboration with mentor ○ Portfolio creation ○ Final teaching evaluation

3.1.4 Elective courses

Elective courses allow Bachelor students to explore topics beyond the core curriculum, deepening their informatics knowledge or pedagogical skills according to personal interests. They provide flexibility to strengthen specific competencies, engage with emerging technologies, or gain additional practical experience, while remaining aligned with the overall learning outcomes of the programme. Through these choices, students can tailor their professional profile and prepare for diverse teaching contexts at primary and lower secondary levels.

Table 41 Description of the elective course 3D Modelling and Digital Fabrication

Course title	3D Modelling and Digital Fabrication
ECTS	4
Course aims	Develop foundational knowledge and practical skills in 3D modelling and digital fabrication, enabling students to create and prepare digital objects for visualisation and basic production while recognising their applications in educational contexts.
Learning outcomes	• Explain core concepts of 3D geometry, meshes, and spatial modelling. • Create and manipulate 3D models using industry-standard software. • Apply principles of scale, proportion, and geometric transformation in modelling. • Prepare digital models for rendering or basic digital fabrication (e.g. 3D printing).
Course content	○ Introduction to 3D coordinate systems ○ Meshes, vertices, edges, and faces ○ Geometric transformations (translation, rotation, scaling) ○ Basic modelling techniques (extrusion, subdivision, Boolean operations) ○ Texturing and materials (introductory level) ○ Rendering fundamentals ○ Export formats and model optimisation ○ Introduction to 3D printing workflow ○ Mini project: design and prototype a 3D object

Table 42 Description of the elective course History of Computing and Informatics

Course title	History of Computing and Informatics
ECTS	4
Course aims	Provide an understanding of the historical development of computing and informatics, enabling students to relate key technological milestones to contemporary digital systems and their role in society and education.
Learning outcomes	• Describe key historical milestones in computing and digital technologies. • Explain how social, economic, and political contexts shaped computing development. • Analyse the evolution of programming paradigms and system architectures. • Connect historical developments to contemporary informatics topics.
Course content	○ Mechanical computation and early calculating machines

	○ Turing, von Neumann, and theoretical foundations ○ Development of programming languages ○ Personal computing revolution ○ The rise of the internet ○ Open-source movement ○ Mobile computing and platform ecosystems ○ Milestones in AI development ○ Computing history in education and society

Table 43 Description of the elective course Learning Analytics and Data-Informed Teaching

Course title	Learning Analytics and Data-Informed Teaching
ECTS	4
Course aims	Introduce the principles of learning analytics and develop students' ability to interpret and use educational data to support informed teaching decisions, while critically addressing ethical and privacy considerations.
Learning outcomes	• Explain the principles of learning analytics and educational data use in school contexts. • Interpret basic learning data (engagement, progress, assessment results) to support pedagogical decision-making. • Use digital tools to collect, visualise, and analyse classroom data. • Design learning activities that generate meaningful evidence of student understanding. • Critically evaluate ethical, privacy, and data protection issues in school-based data use.
Course content	○ Introduction to learning analytics in K-12 education ○ Types of educational data: behavioural, performance, process, and interaction data ○ Data literacy for teachers: interpreting dashboards and visualisations ○ Using learning management system analytics and digital platform reports ○ Formative assessment and data-informed feedback ○ Designing data-rich learning tasks ○ Basics of data visualisation for educators ○ Personalisation and differentiation using learning data ○ Ethical considerations: privacy, GDPR, data minimisation, algorithmic bias ○ Case studies of data-informed teaching practices ○ Mini project: analyse anonymised classroom data and design a data-informed instructional intervention

Table 44 Description of the elective course Career Development and Teamwork

Course title	Career Development and Teamwork
ECTS	4

Course aims	Develop students' teamwork, communication, and reflective skills to support effective collaboration and professional growth in technical and educational contexts.
Learning outcomes	• Demonstrate effective teamwork and collaboration in technical and educational projects. • Apply communication and conflict-resolution strategies in group work. • Reflect on personal strengths, roles, and professional development pathways. • Organise and manage collaborative student projects in educational contexts.
Course content	○ Team roles and team dynamics ○ Communication in collaborative environments ○ Conflict resolution and constructive feedback ○ Leadership and responsibility in teams ○ Project collaboration in digital environments ○ Career planning and professional identity development ○ Reflection on teamwork experiences ○ Mini project: collaborative team project and reflection

Table 45 Description of the elective course Digital Communication and Professional Skills

Course title	**Digital Communication and Professional Skills**
ECTS	4
Course aims	Develop effective digital communication and presentation skills, enabling students to clearly explain informatics concepts and engage diverse audiences in educational and professional settings.
Learning outcomes	• Communicate technical and digital concepts clearly to diverse audiences. • Develop presentation and communication skills for educational contexts. • Apply digital communication tools in professional and teaching environments. • Design clear explanations of informatics concepts for learners.
Course content	○ Communication of technical concepts ○ Presentation and storytelling techniques ○ Visual explanation of complex ideas ○ Digital communication tools ○ Public speaking and classroom communication ○ Communicating technology to non-experts ○ Feedback and constructive dialogue ○ Mini project: present and explain an informatics concept to a non-technical audience

Table 46 Description of the elective course Communication and Digital Pedagogy

Course title	**Communication and Digital Pedagogy**
ECTS	4

Course aims	Develop students' ability to communicate effectively in digital learning environments and to facilitate interactive, inclusive, and engaging teaching practices using appropriate pedagogical strategies and tools.
Learning outcomes	• Apply effective communication strategies in teaching digital subjects. • Facilitate interactive and collaborative learning environments. • Use digital tools to support engagement and participation. • Reflect on communication styles in digital classrooms.
Course content	○ Communication in digital learning environments ○ Teacher-student interaction strategies ○ Facilitation of group work and discussion ○ Digital storytelling and presentation tools ○ Feedback and formative assessment ○ Inclusive communication in diverse classrooms

Table 47 Description of the elective course Game Design and Development

Course title	**Game Design and Development**
ECTS	4
Course aims	Introduce fundamental concepts of game design and development, enabling students to create simple game prototypes and explore their pedagogical potential in informatics education.
Learning outcomes	• Apply core principles of game design: mechanics, dynamics, aesthetics, and player experience. • Implement a simple 2D game prototype using an appropriate game engine or framework. • Use game loops, collision detection, and sprite management as programming concepts. • Evaluate the pedagogical role of game design and development in school informatics education. • Reflect critically on ethical dimensions of game design, including representation and engagement patterns.
Course content	○ Foundations of game design: mechanics, rules, goals, and feedback ○ Game genres and their structural characteristics ○ Introduction to a 2D game engine or framework (e.g. Pygame, Godot, or Unity basics) ○ Game loops, frame rate, and event handling in game programming ○ Sprites, animations, and asset management ○ Collision detection and basic physics simulation ○ Level design and procedural content generation (introductory) ○ Sound integration and game atmosphere ○ Game testing: playtesting methods and iterative design ○ Pedagogical perspectives: game design as a project-based learning context in secondary school

Table 48 Description of the elective course Functional and Logic Programming

Course title	Functional and Logic Programming
ECTS	4
Course aims	Provide foundational knowledge of functional and logic programming paradigms and develop students' ability to apply alternative computational models while reflecting on their educational value.
Learning outcomes	• Distinguish between imperative, functional, and logic programming paradigms at a conceptual and practical level. • Write basic programs using a functional language (e.g. Haskell or Racket) employing higher-order functions, immutability, and recursion. • Explain and apply core functional concepts: pure functions, closures, map/filter/reduce, and lazy evaluation. • Write simple Prolog programs using facts, rules, and unification to solve constraint and search problems. • Reflect on the pedagogical value of paradigm diversity for teaching computational thinking in school contexts.
Course content	○ Overview of programming paradigms and their conceptual foundations ○ Functional programming: pure functions, immutability, and referential transparency ○ Higher-order functions, closures, and lambda expressions ○ Recursion from a functional perspective; tail recursion ○ List processing with map, filter, and fold/reduce ○ Introduction to a functional language (e.g. Haskell or Racket): syntax and evaluation model ○ Introduction to logic programming: facts, rules, and queries in Prolog ○ Unification and backtracking as computational mechanisms ○ Comparative analysis of paradigm suitability for different problem classes ○ Pedagogical perspectives: teaching paradigm diversity at secondary level

Table 49 Description of the elective course Mobile Application Development

Course title	Mobile Application Development
ECTS	4
Course aims	Develop knowledge and skills for designing and implementing simple mobile applications, with attention to user interaction, basic functionality, and their potential use in school informatics projects.
Learning outcomes	• Describe the architecture and development environment of a contemporary mobile platform (e.g. Android or a cross-platform framework). • Design and implement simple mobile applications with a basic graphical user interface. • Apply event-driven and component-based programming concepts in a mobile context.

	• Use platform APIs for basic device features (e.g. camera, sensors, local storage). • Evaluate the pedagogical potential of mobile application development as a school-level project context.
Course content	○ Mobile platform landscape: Android, iOS, and cross-platform frameworks (e.g. Flutter, React Native) ○ Development environment setup and project structure ○ UI components, layouts, and navigation patterns in mobile applications ○ Event-driven programming and callback models on mobile platforms ○ Application lifecycle and state management ○ Accessing device APIs: camera, sensors, GPS, local storage ○ Basic data persistence: local databases and shared preferences ○ Introduction to mobile application testing and debugging ○ Publishing and distribution of mobile applications ○ Pedagogical use cases for mobile development in secondary school informatics

Table 50 Description of the elective course Data Science and Visualisation

Course title	Data Science and Visualisation
ECTS	4
Course aims	Introduce core concepts and practices of data science and visualisation, enabling students to analyse data, communicate insights effectively, and design data-driven learning activities.
Learning outcomes	• Apply a data science workflow (data acquisition, cleaning, exploration, analysis, and communication) to real-world datasets. • Use a data science library (e.g. pandas, NumPy) to manipulate and process structured data programmatically. • Create informative and accurate data visualisations using appropriate chart types and visualisation libraries. • Identify and address common data quality issues including missing values, outliers, and sampling bias. • Design school-level data science activities that promote data literacy and critical thinking about data.
Course content	○ Data science as a discipline: workflow, roles, and tools ○ Data acquisition: structured data formats (CSV, JSON, XML) and basic web data ○ Data cleaning and preprocessing: handling missing values, type conversion, and normalisation ○ Exploratory data analysis (EDA): summary statistics, distributions, and correlations ○ Data wrangling with a programming library (e.g. pandas) ○ Principles of effective data visualisation: chart selection, visual encoding, and Gestalt principles ○ Visualisation libraries (e.g. Matplotlib, Seaborn, or Plotly): practical use ○ Communicating data stories: dashboards, infographics, and narrative visualisation ○ Ethical dimensions: data bias, privacy in data science, and misleading visualisations

	○ Designing data literacy activities for secondary school curricula

Table 51 Description of the elective course Internet of Things and Smart Environments

Course title	Internet of Things and Smart Environments
ECTS	4
Course aims	Introduce the principles and technologies of the Internet of Things, enabling students to design simple connected systems and explore their applications in educational and everyday environments.
Learning outcomes	• Explain the architecture and core components of IoT systems, including sensors, actuators, edge devices, and cloud connectivity. • Design and implement simple IoT prototypes using microcontroller platforms and wireless communication. • Apply basic protocols used in IoT communication (e.g. MQTT, HTTP-based APIs). • Evaluate privacy, security, and ethical risks associated with IoT deployments in everyday and school contexts. • Design a classroom activity or project using IoT concepts appropriate for secondary school learners.
Course content	○ IoT architecture: sensing, connectivity, processing, and actuation layers ○ Wireless communication technologies: Wi-Fi, Bluetooth, Zigbee (conceptual overview) ○ IoT protocols: MQTT and RESTful HTTP APIs for device communication ○ Cloud platforms for IoT: data ingestion, storage, and dashboards (e.g. ThingSpeak, Adafruit IO) ○ Sensor types and data acquisition: environmental, motion, and proximity sensors ○ Edge computing concepts: local processing vs. cloud offloading ○ Security and privacy in IoT: attack surfaces, authentication, and data protection ○ Smart home, smart city, and smart school use cases ○ Hands-on IoT prototyping (e.g. ESP32 or Raspberry Pi with sensors) ○ Pedagogical integration: IoT as a cross-disciplinary project context in schools

Table 52 Description of the elective course Foundations of Computer Graphics

Course title	Foundations of Computer Graphics
ECTS	4
Course aims	Provide foundational knowledge of computer graphics, enabling students to understand how visual content is generated and manipulated in digital systems and applied in interactive applications.
Learning outcomes	• Explain how digital images are generated and represented in computer systems. • Describe basic rendering techniques and the graphics pipeline conceptually. • Apply simple geometric and colour transformations. • Analyse visual artefacts such as aliasing and resolution limits.
Course content	○ Raster vs vector graphics

	○ Pixels, resolution, colour models (RGB, HSV) ○ Coordinate systems and transformations ○ Line and polygon drawing algorithms (conceptual overview) ○ 2D and introductory 3D graphics pipeline ○ Shading and lighting basics (conceptual) ○ Animation fundamentals ○ Image processing basics ○ Graphics in games, simulations, and interfaces

Table 53 Description of the elective course Use of AI in Education

Course title	Use of AI in Education
ECTS	4
Course aims	Develop a critical understanding of the role of Artificial Intelligence in education, enabling students to evaluate, select, and apply AI tools in pedagogically meaningful ways while addressing ethical and data protection considerations.
Learning outcomes	• Explain fundamental concepts of Artificial Intelligence relevant to educational contexts. • Analyse how AI systems (e.g., adaptive learning systems, chatbots, automated assessment tools) function at a conceptual level. • Critically evaluate the pedagogical benefits and limitations of AI in teaching and learning. • Apply AI-based tools to design inclusive and engaging learning activities. • Identify ethical, legal, and data protection considerations related to AI use in schools.
Course content	○ Introduction to Artificial Intelligence: key concepts and terminology ○ Overview of AI applications in education (adaptive learning, intelligent tutoring systems, learning analytics, generative AI) ○ AI-supported assessment and feedback ○ Personalisation and differentiation using AI tools ○ Generative AI in teaching: lesson planning, content creation, and student support ○ Critical AI literacy for teachers and students ○ Ethics of AI in education: bias, fairness, transparency, privacy, academic integrity ○ Data protection and responsible AI use in school environments ○ Hands-on exploration of AI tools for education (e.g., AI content generators, Teachable Machine, visual ML platforms, AI-based quiz and feedback tools) ○ Designing AI-enhanced learning scenarios ○ Mini-project: develop and present an AI-integrated lesson plan or classroom intervention

Table 54 Description of the elective course Introduction to Electronics

Course title	Introduction to Electronics

ECTS	4
Course aims	Introduce basic concepts of electronics and circuit design, enabling students to build simple systems and understand the hardware foundations of computing technologies.
Learning outcomes	• Explain fundamental electrical concepts (voltage, current, resistance). • Interpret simple circuit diagrams. • Build and test basic analogue and digital circuits. • Relate electronics principles to microcontrollers and computing systems.
Course content	○ Basic electrical quantities and Ohm's law ○ Series and parallel circuits ○ Resistors, capacitors, diodes, LEDs ○ Transistors and switching principles ○ Analogue vs digital signals ○ Logic gates at hardware level ○ Sensors and actuators ○ Breadboarding and circuit prototyping ○ Introductory embedded systems examples

Table 55 Description of the elective course Physics for Computing

Course title	**Physics for Computing**
ECTS	4
Course aims	Provide an understanding of key physics concepts, enabling students to relate physical principles to hardware, signals, and computational models.
Learning outcomes	• Explain key physics principles. • Apply basic mechanics, electromagnetism, and wave concepts to digital systems. • Analyse physical constraints of hardware systems. • Connect physical models to simulations and computational representations.
Course content	○ Units, measurement, and modelling ○ Basic mechanics (force, motion, energy) ○ Electricity and magnetism ○ Waves and signal transmission ○ Semiconductor basics (conceptual) ○ Optics and displays ○ Energy efficiency and sustainability in computing ○ Physics-based simulation examples

Table 56 Description of the elective course Linear Algebra for Computing

Course title	**Linear Algebra for Computing**
ECTS	4
Course aims	Develop foundational knowledge of linear algebra, enabling students to model and solve problems involving vectors, matrices, and transformations.

Learning outcomes	• Apply vector and matrix operations in computational contexts. • Explain geometric interpretations of linear transformations. • Use matrices to represent systems, graphs, and transformations.
Course content	○ Vectors and vector spaces ○ Matrix operations ○ Systems of linear equations ○ Determinants and invertibility ○ Eigenvalues and eigenvectors (conceptual level) ○ Linear transformations and geometric interpretation ○ Matrix representations in graphs and networks

Table 57 Description of the elective course Mathematical Analysis for Modelling

Course title	Mathematical Analysis for Modelling
ECTS	4
Course aims	Provide a conceptual understanding of mathematical analysis and its role in modelling dynamic systems, enabling students to interpret and apply continuous mathematical processes in computational contexts.
Learning outcomes	• Explain limits, derivatives, and integrals conceptually. • Apply calculus to modelling dynamic systems. • Interpret rates of change and accumulation processes. • Connect continuous models to computational simulations.
Course content	○ Functions and limits ○ Differentiation and rates of change ○ Integration and accumulation ○ Differential equations (introductory) ○ Continuous vs discrete modelling ○ Optimisation problems ○ Applications in physics and simulation ○ Numerical approximation methods ○ Computational visualisation of functions

3.1.5 Concluding Remarks

The Bachelor's programme establishes a coherent and progressively structured pathway that integrates informatics knowledge, pedagogical competence, and sustained school-based experience. Through the balanced combination of subject-specific courses, didactic preparation, and increasingly autonomous teaching practice, students develop the foundational competences required for responsible and effective informatics teaching at primary and lower secondary level.

The integration of disciplinary depth, reflective pedagogy, inclusive strategies, and extended practicum reflects the functional approach promoted by the Digital First project. Graduates are prepared not only to teach core informatics concepts meaningfully and ethically, but also to support diverse learners and respond to evolving technological developments. The programme therefore provides a strong academic and professional basis either for entry into the

teaching profession (in accordance with national regulations) or for progression to the Master's level described in the following section.

3.2 Master's Degree (2 years)

The Master's programme in Informatics Education builds upon the foundational competences developed at Bachelor level and prepares graduates for teaching advanced informatics at upper secondary level (ISCED 34, depending on national systems), as well as for leadership and development roles within schools. The programme comprises 120 ECTS credits distributed across two academic years and is designed to deepen subject-specific expertise, strengthen advanced didactic competence, and foster research-informed professional practice.

While the Bachelor programme emphasises foundational disciplinary knowledge and initial professional formation, the Master's programme focuses on conceptual depth, curriculum development, advanced assessment strategies, and the supervision of complex student projects. Particular attention is given to the pedagogical transformation of advanced informatics topics, enabling future teachers to translate complex disciplinary knowledge into forms accessible and meaningful to upper secondary learners.

The structure of the Master's study programme is presented in Table 58.

Table 58 Structure of the Master's study programme

Course	Semester	ECTS
Didactics of Informatics for Upper Secondary	1	6
Cybersecurity	1	5
Advanced Topics in School Informatics	1	6
Curriculum Design and Educational Policy in Informatics	1	5
Elective course I	1	4
Elective course II	1	4
Total	*1*	*30*
Research Methods in Informatics Education	2	5
Cloud Computing	2	5
School Practicum for Upper Secondary Teaching	2	12
Elective course III	2	4
Elective course IV	2	4
Total	*2*	*30*
Advanced Data Analysis for Educational Research	3	6
Educational Technology and Digital Pedagogies	3	5
Professional Identity and Leadership for Informatics Teachers	3	5
Assessment and Evaluation in Informatics Education	3	6
Elective course V	3	4
Elective course VI	3	4
Total	*3*	*30*
Master Seminar	4	6

Master Thesis	4	20
Elective course VII	4	4
Total	*5*	*30*

All	**120**

The structure of the Master's programme is organised into two coherent phases that progressively develop advanced disciplinary, pedagogical, and research competences.

The first year focuses on deepening subject-specific expertise and strengthening advanced didactic competence for upper secondary education. Students engage with complex informatics topics, curriculum design, cybersecurity, and cloud technologies, while simultaneously developing the ability to pedagogically transform advanced disciplinary knowledge into accessible learning experiences. The inclusion of research methods and upper secondary teaching practice ensures that theoretical knowledge is consistently connected with reflective classroom application.

The second year emphasises professional consolidation, leadership development, and research-based practice. Students refine their assessment competence, explore educational technologies and digital pedagogies at an advanced level, and develop professional identity and leadership skills. The *Master Seminar* and *Master Thesis* represent the culmination of the programme, enabling students to conduct independent research or development work in informatics education and to demonstrate analytical, methodological, and academic maturity.

Elective courses distributed across semesters allow students to select topics aligned with their professional interests, while maintaining coherence with the overall competence framework defined within the Digital First project.

3.2.1 Year 1

Semester 1

In the first semester of the first year of the postgraduate study programme, students will attend the following mandatory courses: *Didactics of Informatics for Upper Secondary*, *Cybersecurity*, *Advanced Topics in School Informatics*, and *Curriculum Design and Educational Policy in Informatics*. Students will also choose two elective subjects. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 59 Description of the course Didactics of Informatics for Upper Secondary

Course title	Didactics of Informatics for Upper Secondary
Year of study	1 (M)
ECTS	6
Course aims	Train students to plan, implement, and evaluate advanced informatics instruction at upper secondary level (ISCED 34), with emphasis on complex concepts, long-term projects, assessment, and student supervision.
Enrolment requirements	There is none.
Learning outcomes	• Design coherent lesson sequences for advanced informatics topics. • Apply project-based and inquiry-based approaches in upper secondary informatics. • Develop assessment criteria for complex student work and projects.

	• Teach topics related to computing systems, networks and internet, data and analysis, algorithms and programming, impacts of computing, AI, and cybersecurity. • Supervise student projects and research-oriented tasks. • Critically reflect on ethical, legal, and societal aspects of informatics education.
Course content	○ Upper secondary informatics curricula and learning objectives ○ Didactic transposition of advanced informatics concepts ○ Project-based and inquiry-based learning in informatics ○ Teaching computing systems, networks and internet, data and analysis, algorithms and programming, impacts of computing, AI, and cybersecurity at upper secondary level ○ Assessment of complex tasks and student projects ○ Advanced debugging instruction and conceptual difficulties ○ Ethical, legal, and societal dimensions of informatics ○ Interdisciplinary approaches and authentic tasks ○ Supervision of student research and graduation projects

Table 60 Description of the course Cybersecurity

Course title	**Cybersecurity**
Year of study	1 (M)
ECTS	5
Course aims	Provide cybersecurity knowledge and promote secure and responsible use of digital technologies in educational settings.
Enrolment requirements	There is none.
Learning outcomes	• Explain cybersecurity concepts relevant to school contexts. • Identify common digital threats and protection measures. • Design age-appropriate learning activities for teaching cybersecurity. • Promote online safety, privacy, and responsible digital behaviour.
Course content	○ Threat modelling ○ Encryption ○ Authentication ○ Secure communication ○ Malware and attacks ○ Security policies ○ Online safety ○ Secure coding basics ○ Risk assessment ○ Teaching cybersecurity and practical examples

Table 61 Description of the course Advanced Topics in School Informatics

Course title	**Advanced Topics in School Informatics**

Year of study	1 (M)
ECTS	6
Course aims	Develop students' ability to analyse advanced informatics topics taught at upper secondary level from a conceptual, pedagogical, and didactic perspective, and to translate disciplinary knowledge into forms suitable for school teaching.
Enrolment requirements	There is none.
Learning outcomes	• Analyse advanced informatics topics in terms of core concepts, abstraction, and complexity. • Interpret key informatics areas (algorithms, data, modelling, artificial intelligence, cybersecurity) at a conceptual school level. • Identify typical student misconceptions and learning difficulties in advanced informatics topics. • Apply didactic reduction and pedagogical transformation to address misconceptions and conceptual difficulties. • Contextualise advanced informatics concepts within real-world and societal settings relevant to school education.
Course content	○ Core concepts and abstraction levels in advanced informatics ○ Algorithms, data, modelling, artificial intelligence, and cybersecurity at a conceptual school level ○ Complexity and formalisation in informatics topics ○ Typical student misconceptions and learning difficulties ○ Didactic reduction and pedagogical transformation of complex content ○ Pedagogical representations and explanations ○ Real-world and societal contextualisation of informatics topics ○ Progression of advanced informatics concepts in school education

Table 62 Description of the course Curriculum Design and Educational Policy in Informatics

Course title	**Curriculum Design and Educational Policy in Informatics**
Year of study	1 (M)
ECTS	5
Course aims	Develop competence in analysing, interpreting, and designing informatics curricula within national and international educational policy frameworks.
Enrolment requirements	There is none.
Learning outcomes	• Analyse informatics curricula at different educational levels. • Compare national and international curriculum frameworks. • Design coherent curriculum sequences for upper secondary informatics. • Understand the role of educational policy in shaping informatics education. • Critically reflect on curriculum goals, competencies, and standards.
Course content	○ Curriculum theory and models ○ National informatics curricula and standards

	○ International frameworks (e.g. I4All, grounded in ACM/CSTA K-12, EU digital competence frameworks) ○ Vertical and horizontal curriculum alignment ○ Competency-based curriculum design ○ Educational policy and governance in informatics education ○ Curriculum reform and innovation processes ○ Role of teachers in curriculum development

Semester 2

In the second semester of the first year of the postgraduate study programme, students will attend the following mandatory courses: *Research Methods in Informatics Education*, *Cloud Computing*, and *School Practicum for Upper Secondary Teaching*. Students will also choose two elective subjects. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 63 Description of the course Research Methods in Informatics Education

Course title	Research Methods in Informatics Education
Year of study	1 (M)
ECTS	5
Course aims	Develop the ability to plan, conduct, and critically interpret research in computing and informatics education, grounded in an understanding of computing as a scientific discipline, with emphasis on domain-specific research questions, methods, and evidence-based improvement of teaching practice.
Enrolment requirements	There is none.
Learning outcomes	• Formulate clear and researchable problems, aims, and questions in informatics education. • Design small-scale studies using appropriate quantitative, qualitative, or mixed research approaches in educational and computing-related contexts. • Select and apply suitable data collection techniques and basic data analysis procedures relevant to computing education research. • Evaluate validity, reliability, and ethical aspects of research in computing education. • Use digital tools and platforms to support data collection, analysis, and reflective research of teaching practice. • Interpret and communicate research findings in a clear and academically appropriate manner, aligned with conventions of computing education research. • Demonstrate an understanding of computing as a scientific discipline and reflect this understanding in the formulation and interpretation of research in computing education.
Course content	○ Foundations of informatics (computing) education research: purposes, scope, and research approaches (quantitative, qualitative, mixed). ○ Research designs in computing education: descriptive, design-based, causal-experimental, and non-experimental studies. ○ Research process in computing education: problem definition, research questions, literature review, data collection, analysis, and interpretation.

	○ Data collection methods in computing education research: surveys, concept inventories, tests, interviews, classroom observation, learning artefacts, and student work analysis. ○ Basics of data analysis for educational research: data organisation, graphical presentation, descriptive statistics, correlation, and interpretation of statistical significance. ○ Digital tools for computing education research: computer-based data analysis, online platforms for data collection, classroom analytics, and self-evaluation. ○ Academic reporting, research ethics, and systematic literature review in computing education using scholarly databases.

Table 64 Description of the course Cloud Computing

Course title	Cloud Computing
Year of study	1 (M)
ECTS	5
Course aims	Introduce future informatics teachers to cloud computing concepts, models, and educational implications, enabling them to understand, use, and critically evaluate cloud-based infrastructures and services in school, teaching, and learning contexts.
Enrolment requirements	There is none.
Learning outcomes	• Explain core cloud computing concepts and service models (IaaS, PaaS, SaaS) in an educational context. • Use selected cloud-based platforms and services for teaching, collaboration, and data-driven school activities. • Analyse benefits, limitations, and risks of cloud solutions in schools (privacy, security, vendor lock-in, sustainability). • Design simple educational scenarios that meaningfully integrate cloud services into informatics teaching. • Relate cloud computing to data, AI services, and digital school infrastructure, including ethical and legal considerations.
Course content	○ Introduction to cloud computing ○ Cloud service and deployment models ○ Cloud platforms in education ○ Cloud, data, and AI ○ Security, privacy, and ethics ○ Didactic perspectives

Table 65 Description of the course School Practicum for Upper Secondary Teaching

Course title	School Practicum for Upper Secondary Teaching
Year of study	1 (M)
ECTS	12

Course aims	Prepare students for full independent teaching of informatics at upper secondary level (ISCED 34).
Enrolment requirements	Completed course *Didactics of Informatics for Upper Secondary*.
Learning outcomes	• Independently plan and deliver advanced informatics lessons. • Apply didactic and pedagogical knowledge in real classroom contexts. • Assess student learning using appropriate methods. • Manage diverse classrooms effectively. • Reflect critically on own teaching practice.
Course content	○ Teaching advanced informatics topics ○ Planning and delivering lesson sequences ○ Classroom leadership and management ○ Project supervision and mentoring ○ Use of advanced digital tools ○ Peer, mentor, and self-evaluation ○ Student feedback and reflection ○ Teaching portfolio development ○ Final teaching evaluation

3.2.2 Year 2

Semester 1

In the first semester of the second year of the postgraduate study programme, students will attend the following mandatory courses: *Advanced Data Analysis for Educational Research*, *Educational Technology and Digital Pedagogies*, *Assessment and Evaluation in Informatics Education*, and *Professional Identity and Leadership for Informatics Teachers*. Students will also choose two elective subjects. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 66 Description of the course Advanced Data Analysis for Educational Research

Course title	**Advanced Data Analysis for Educational Research**
Year of study	2 (M)
ECTS	6
Course aims	Develop advanced competence in analysing and interpreting educational data relevant to computing education research, with emphasis on rigorous quantitative and qualitative analysis, critical interpretation of results, and evidence-based conclusions.
Enrolment requirements	Completed course *Research Methods in Informatics Education*.
Learning outcomes	• Apply advanced quantitative and qualitative data analysis techniques in computing education research contexts. • Interpret statistical and qualitative analysis results in relation to research questions in computing education. • Use digital tools and software for data analysis, visualisation, and reporting of research findings. • Critically evaluate data quality, assumptions, limitations, and methodological constraints of educational datasets.

	• Integrate results from quantitative and qualitative analyses in mixed-methods computing education studies. • Communicate research findings effectively using academically appropriate representations and conventions.
Course content	○ Apply advanced quantitative and qualitative data analysis techniques in computing education research contexts. ○ Interpret statistical and qualitative analysis results in relation to research questions in computing education. ○ Use digital tools and software for data analysis, visualisation, and reporting of research findings. ○ Critically evaluate data quality, assumptions, limitations, and methodological constraints of educational datasets. ○ Integrate results from quantitative and qualitative analyses in mixed-methods computing education studies. ○ Communicate research findings effectively using academically appropriate representations and conventions.

Table 67 Description of the course Educational Technology and Digital Pedagogies

Course title	Educational Technology and Digital Pedagogies
Year of study	2 (M)
ECTS	5
Course aims	Develop the ability to critically select, design, and evaluate the pedagogical use of digital technologies, including AI-based systems, in diverse educational contexts.
Enrolment requirements	There is none.
Learning outcomes	• Design learning activities that meaningfully integrate digital technologies and AI. • Critically evaluate educational technologies in relation to pedagogical goals, learners' needs, and ethical considerations. • Apply digital pedagogy frameworks to analyse teaching and learning scenarios. • Use digital assessment and learning analytics to support formative and inclusive teaching practices. • Reflect on ethical, legal, and social implications of educational technologies.
Course content	○ Digital pedagogy frameworks ○ AI-supported learning and teaching ○ Adaptive and personalised learning systems ○ Digital assessment and feedback ○ Learning analytics in education ○ Online and blended learning models ○ Immersive technologies (VR/AR) in education ○ Gamification and game-informed pedagogy ○ Open Educational Resources (OER) ○ Ethical, legal, and inclusive dimensions of educational technology

Table 68 Description of the course Assessment and Evaluation in Informatics Education

Course title	Assessment and Evaluation in Informatics Education
Year of study	2 (M)
ECTS	6
Course aims	Prepare students to design, implement, and critically reflect on assessment strategies for informatics education at upper secondary level, with emphasis on formative support of learning, assessment of project-based work and software artefacts, alignment with curricula, and ethical and inclusive assessment practices.
Enrolment requirements	Completed course *Didactics of Informatics for Upper Secondary*.
Learning outcomes	• Develop clear rubrics and criteria for assessing complex tasks, projects, and student-developed software artefacts. • Align assessment methods with learning objectives and upper secondary informatics curricula. • Use digital tools to support assessment, feedback, and monitoring of student progress. • Interpret and reflect on assessment data to support student learning and instructional decisions. • Apply peer assessment and peer testing approaches (e.g. peer programming, peer review) to support learning and reflection. • Combine multiple sources of evidence (e.g. project work, tests, peer feedback, teacher judgement) using triangulation to ensure fair and transparent assessment. • Assess student-developed software artefacts with respect to functionality, correctness, readability, documentation, and alignment with learning goals.
Course content	○ Principles of educational assessment in informatics ○ Formative and summative assessment strategies ○ Project-based and performance-based assessment ○ Assessment of programming and computational thinking concepts ○ Assessment of student-developed software artefacts ○ Peer assessment and peer testing (e.g. peer programming, peer review, test design) ○ Rubrics, criteria, and grading reliability ○ Digital assessment tools and platforms ○ Feedback strategies and introductory learning analytics ○ Triangulation of assessment evidence (projects, tests, peer input, teacher judgement) ○ Ethical, inclusive, and transparent assessment practices

Table 69 Description of the course Professional Identity and Leadership for Informatics Teachers

Course title	Professional Identity and Leadership for Informatics Teachers
Year of study	2 (M)
ECTS	5

Course aims	Support the development of professional identity, ethical responsibility, leadership, and collaborative competence among future informatics teachers, with emphasis on curriculum leadership, school innovation, and participation in professional communities.
Enrolment requirements	Completed course *School Practicum for Upper Secondary Teaching*.
Learning outcomes	• Critically reflect on their professional role and identity as informatics educators in contemporary school systems. • Demonstrate leadership in educational, curricular, and school-development contexts related to informatics and digital education. • Collaborate effectively with colleagues, school leadership, and external stakeholders in curriculum development and innovation processes. • Engage in reflective, evidence-based professional development, including the use of research, self-evaluation, and peer learning. • Contribute to innovation in informatics education, including participation in professional networks, projects, and communities of practice.
Course content	○ Teacher leadership and mentoring in school contexts ○ Collaboration and professional learning communities (PLCs) ○ Curriculum leadership and school-level innovation in informatics ○ Communication and collaboration with stakeholders (schools, parents, policy, community) ○ Ethics, responsibility, and professional standards of the teaching profession ○ Lifelong learning (LLL) and continuing professional development (CPD) ○ Reflective practice and teacher inquiry / practitioner research

Semester 2

In the second semester of the second year of the postgraduate study programme, students will attend the following mandatory courses: *Master Seminar* and *Master Thesis*. Students will also choose one elective subject. More detailed descriptions of the mandatory courses are provided in the following tables.

Table 70 Description of the course Master Seminar

Course title	**Master Seminar**
Year of study	2 (M)
ECTS	6
Course aims	Provide structured academic support for Master thesis development through peer discussion, feedback, and research communication.
Enrolment requirements	Enrolment in Year 2 of the master's programme.
Learning outcomes	• Present and discuss their research or development projects. • Provide and receive constructive academic feedback. • Reflect critically on research processes and challenges. • Communicate research ideas clearly and professionally.
Course content	○ Presentation of thesis proposals and progress ○ Peer feedback and academic discussion ○ Research design refinement ○ Ethical and methodological reflection

	○ Academic writing and argumentation ○ Preparation for thesis defence

Table 71 Description of the course Master Thesis

Course title	Master Thesis
Year of study	2 (M)
ECTS	20 ECTS
Course aims	Enable students to conduct independent research or develop an applied innovation in informatics education, demonstrating Master-level academic and professional competence.
Enrolment requirements	Successful completion of all previous coursework.
Learning outcomes	• Design and conduct a methodologically sound research or development project. • Apply appropriate research methods and analysis techniques. • Critically engage with relevant academic literature. • Present findings clearly in written and oral form. • Reflect on the implications of their work for informatics education practice.
Course content	○ Research design and planning ○ Literature review ○ Data collection and analysis ○ Academic writing and documentation ○ Ethical research principles ○ Thesis submission and defence ○ Reflection on contribution to research or practice

3.2.3 Elective courses

Master's electives offer students opportunities to specialise and deepen expertise in advanced informatics topics, innovative pedagogies, or leadership-oriented skills. They support research-informed practice, conceptual sophistication, and professional differentiation, enabling students to shape a profile aligned with their career goals. Electives complement the core curriculum while ensuring coherence with the programme's competence framework, preparing graduates for teaching, curriculum development, and educational innovation at upper secondary level.

Table 72 Description of the elective course Interdisciplinary Project Design in Informatics

Course title	Interdisciplinary Project Design in Informatics
ECTS	4
Course aims	Develop students' ability to design, implement, and evaluate interdisciplinary informatics projects that integrate multiple subject areas and support meaningful, project-based learning in school contexts.
Learning outcomes	• Design interdisciplinary informatics projects aligned with curriculum goals. • Integrate informatics with mathematics, science, or social sciences. • Plan long-term student project supervision. • Evaluate interdisciplinary learning outcomes.
Course content	○ Principles of interdisciplinary learning ○ Project-based learning design

	○ Authentic real-world problem contexts ○ Assessment of interdisciplinary projects ○ Collaboration with other subject teachers ○ Risk management in long-term projects ○ Case studies ○ Project proposal development

Table 73 Description of the elective course Game-Based Learning and Gamification

Course title	Game-Based Learning and Gamification
ECTS	4
Course aims	Introduce the principles of game-based learning and gamification, enabling students to design engaging learning experiences and critically evaluate their impact on motivation and learning outcomes.
Learning outcomes	• Understand principles of gamification and game-based learning. • Design learning activities that integrate game mechanics. • Evaluate educational games and digital learning environments. • Apply game design elements to motivate students in informatics learning.
Course content	○ Foundations of gamification ○ Game mechanics and motivation theory ○ Educational games in informatics teaching ○ Designing simple educational games ○ Gamified classroom activities ○ Evaluating engagement and learning outcomes

Table 74 Description of the elective course Community Engagement through Digital Technologies

Course title	Community Engagement through Digital Technologies
ECTS	4
Course aims	Develop students' ability to design and facilitate learning projects that connect informatics with community engagement and social impact through the meaningful use of digital technologies.
Learning outcomes	• Design learning projects connecting informatics with community challenges. • Apply digital tools for civic engagement and social impact. • Facilitate student participation in community-oriented digital projects. • Reflect on the societal role of informatics education.
Course content	○ Technology for social good ○ Digital participation and civic engagement ○ Community-based learning projects ○ Digital storytelling and advocacy tools ○ Collaboration with local organisations ○ Evaluation of social impact projects

Table 75 Description of the elective course Emerging Technologies in Society

Course title	Emerging Technologies in Society
ECTS	4
Course aims	Provide an understanding of emerging digital technologies and their societal implications, enabling students to critically analyse innovation trends and integrate relevant topics into informatics education.
Learning outcomes	• Identify emerging digital technologies and their societal implications. • Analyse how new technologies influence work, education, and communication. • Evaluate opportunities and risks of technological innovation. • Integrate emerging technology topics into informatics education.
Course content	○ Emerging technology landscape ○ Internet of Things and smart environments ○ Extended reality and immersive technologies ○ Automation and digital transformation ○ Technology and future labour markets ○ Ethical and social implications of technological innovation ○ Technology foresight and future scenarios ○ Mini project: analyse an emerging technology and its educational implications

Table 76 Description of the elective course Teaching Gifted Students in Informatics

Course title	Teaching Gifted Students in Informatics
ECTS	4
Course aims	Develop knowledge and pedagogical strategies for supporting gifted students in informatics, enabling students to design differentiated learning experiences and mentor advanced learners effectively.
Learning outcomes	• Identify characteristics and learning needs of gifted students in informatics education. • Design differentiated learning activities for students with advanced digital and computational abilities. • Apply instructional strategies that support creativity, problem solving, and advanced exploration in computing. • Mentor gifted students in extended projects, competitions, and research-oriented activities.
Course content	○ Characteristics of gifted students in STEM and informatics ○ Identification of giftedness in computing and digital problem solving ○ Differentiated instruction for advanced learners ○ Enrichment and acceleration strategies in informatics education ○ Mentoring student projects, hackathons, and competitions ○ Supporting creativity and innovation in computing ○ Social and emotional needs of gifted learners ○ Case studies of programmes for gifted students in informatics

	○ Mini project: design an enrichment activity or mentoring plan for a gifted informatics student

Table 77 Description of the elective course Philosophy of Computer Science and Epistemology of Informatics

Course title	Philosophy of Computer Science and Epistemology of Informatics
ECTS	4
Course aims	Provide a critical understanding of the philosophical foundations and epistemological nature of informatics, enabling students to reflect on its status as a discipline and its implications for curriculum design and teaching.
Learning outcomes	• Critically analyse competing accounts of the nature of informatics and computing as scientific, mathematical, and engineering disciplines. • Evaluate foundational philosophical questions concerning computation, information, and intelligence, drawing on primary philosophical and technical literature. • Apply epistemological concepts, such as models, abstraction, and representation, to critique the claims made within informatics subdisciplines. • Synthesise philosophical perspectives to construct a coherent rationale for informatics as a school subject, addressing its distinct epistemic character. • Design curriculum content that conveys the nature and epistemology of informatics to upper secondary learners in an intellectually honest and rigorous manner.
Course content	○ What is informatics? Competing definitions: science, engineering, mathematics, and design ○ The Church-Turing thesis and its philosophical implications for the concept of computation ○ Information theory and the philosophy of information: Shannon, Floridi, and beyond ○ Abstraction, models, and representation as epistemic tools in informatics ○ The nature of algorithms: mathematical objects, processes, or artefacts? ○ Artificial intelligence and the philosophy of mind: strong AI, weak AI, and the Chinese Room ○ Ethics of technology as applied philosophy: autonomy, responsibility, and algorithmic bias ○ Science, technology, and society (STS) perspectives on informatics ○ The nature of informatics knowledge and its implications for curriculum design ○ Teaching the "nature of informatics": strategies and challenges for upper secondary education

Table 78 Description of the elective course Learning Analytics and Educational Data Mining

Course title	Learning Analytics and Educational Data Mining
ECTS	4
Course aims	Develop advanced knowledge of learning analytics and educational data mining, enabling students to analyse educational data, design data-informed interventions, and critically evaluate their methodological and ethical implications.
Learning outcomes	• Critically evaluate the theoretical and methodological foundations of learning analytics and educational data mining as research fields. • Apply appropriate data mining and pattern recognition techniques to educational datasets, including log data, assessment data, and learner interaction traces. • Design learning analytics interventions (dashboards, early-warning systems, or personalised feedback) grounded in evidence and ethical principles. • Interpret and critique existing learning analytics research, identifying methodological limitations and ethical concerns. • Assess the implications of algorithmic decision-making in educational contexts for equity, autonomy, and institutional responsibility.
Course content	○ Learning analytics as a field: origins, scope, and relationship to educational data mining ○ Types of educational data: log files, assessment records, trace data, and physiological signals ○ Sequence analysis and temporal pattern mining in learner activity data ○ Clustering and classification applied to learner modelling and profiling ○ Student success prediction and early-warning systems: methods and evidence ○ Learning analytics dashboards: design principles, actionability, and user studies ○ Intelligent tutoring systems and adaptive learning from an analytics perspective ○ GDPR, consent, and ethical governance of educational data ○ Equity, algorithmic bias, and the risk of amplifying disadvantage in educational settings ○ Critical perspectives on datafication of learning: power, surveillance, and institutional use of data

Table 79 Description of the elective course Advanced Topics in AI

Course title	Advanced Topics in AI
ECTS	4
Course aims	Provide an in-depth understanding of advanced Artificial Intelligence concepts, enabling students to critically analyse contemporary AI systems and design meaningful learning activities addressing their technical, social, and ethical dimensions.
Learning outcomes	• Critically analyse the conceptual underpinnings of advanced AI subfields (natural language processing, computer vision, and reinforcement learning) at a level suitable for curriculum design and advanced teaching.

	• Evaluate the technical and social limitations of contemporary AI systems, including distributional shift, adversarial examples, and model interpretability. • Synthesise developments in large language models and foundation models, assessing their implications for education, labour, and society. • Design upper secondary informatics learning activities that engage critically and accurately with advanced AI concepts. • Construct a principled position on the governance and regulation of AI systems, drawing on technical, legal, and ethical frameworks.
Course content	○ Natural language processing: tokenisation, embeddings, transformers, and language model architectures (conceptual deep-dive) ○ Computer vision: convolutional neural networks, object detection, and image segmentation (conceptual level) ○ Reinforcement learning: agents, rewards, policies, and value functions; key algorithms conceptually (Q-learning, policy gradients) ○ Foundation models and large language models: capabilities, limitations, and emergent behaviour ○ Adversarial examples, distributional shift, and robustness of AI systems ○ Explainability and interpretability: LIME, SHAP, and model transparency ○ AI governance, regulation, and standards: EU AI Act and international frameworks ○ AI and the future of work: labour market implications and skills debates ○ AI in education beyond generative tools: adaptive systems, intelligent tutoring, and predictive analytics ○ Advanced curriculum design for upper secondary AI education

Table 80 Description of the elective course Comparative Informatics Education: International Perspectives

Course title	Comparative Informatics Education: International Perspectives
ECTS	4
Course aims	Develop the ability to analyse and compare international approaches to informatics education, enabling students to draw on global perspectives to inform curriculum design and educational policy decisions.
Learning outcomes	• Conduct systematic comparative analysis of informatics curricula across multiple national educational systems, applying established frameworks for curriculum comparison. • Critically evaluate the influence of international bodies (ACM, CSTA, UNESCO, EU) on national informatics curriculum policy. • Analyse cultural, institutional, and socio-economic factors that shape the implementation of informatics education across different contexts. • Construct evidence-based arguments for specific curriculum design decisions by reference to comparative international evidence. • Produce a scholarly comparative analysis of an informatics education policy question of educational significance.

Course content	○ Frameworks for comparative education research: tertia comparationis, borrowing and lending, and globalisation theories ○ Informatics curriculum models globally: computing science, digital literacy, and ICT traditions ○ The K-12 CS Framework (ACM/CSTA) and its international adaptations ○ Informatics education in Europe: national profiles, Bebras movement, and regional variation ○ Computing education in Asia: East Asian models (Japan, South Korea, China) ○ Computing education in the Global South: equity, access, and infrastructure challenges ○ Teacher education for informatics: comparative policy analysis ○ International assessments and benchmarks in digital and computational literacy ○ Gender, equity, and diversity in informatics education: international evidence ○ Producing a comparative policy analysis: methodology, sources, and academic argumentation

Table 81 Description of the elective course Human-AI Interaction and Intelligent Interfaces

Course title	**Human-AI Interaction and Intelligent Interfaces**
ECTS	4
Course aims	Develop an understanding of human-AI interaction principles, enabling students to design and critically evaluate intelligent systems with a focus on usability, trust, and ethical considerations in educational contexts.
Learning outcomes	• Critically evaluate the unique challenges of human-AI interaction relative to classical human-computer interaction, including issues of transparency, calibration, and trust. • Apply mixed-initiative and human-centred AI design principles to the design and critique of AI-powered educational tools. • Analyse empirical research on user perception, mental models, and behaviour in interactions with AI systems. • Evaluate conversational agents and intelligent tutoring systems using appropriate heuristics and user study methods. • Design and critically justify an AI-integrated learning tool concept for an upper secondary informatics context, addressing interaction design, ethical implications, and accessibility.
Course content	○ From classical HCI to human-AI interaction: new challenges of probabilistic and adaptive systems ○ Mental models and explainability: how users understand (and misunderstand) AI systems ○ Trust calibration in human-AI interaction: automation bias, over-trust, and under-trust

	○ Conversational interfaces: chatbot design, dialogue management, and failure modes ○ Mixed-initiative systems: when should AI act autonomously vs. defer to humans? ○ Fairness and transparency in AI-powered interfaces: technical and design dimensions ○ Evaluation methods for human-AI interaction: controlled experiments, think-aloud, and interaction logs ○ Intelligent tutoring systems: design principles, effectiveness evidence, and interaction patterns ○ AI nudging, persuasive design, and ethical boundaries in educational interfaces ○ Design studio: critiquing and redesigning AI-integrated educational tools

Table 82 Description of the elective course Digital Inequality, Equity, and Informatics Education

Course title	Digital Inequality, Equity, and Informatics Education
ECTS	4
Course aims	Develop a critical understanding of digital inequality and equity in informatics education, enabling students to design inclusive educational strategies and evaluate policies addressing participation and access.
Learning outcomes	• Critically analyse theoretical frameworks for understanding digital inequality (the digital divide, digital capital, and algorithmic discrimination) and evaluate their explanatory power. • Synthesise empirical evidence on the relationship between socio-economic, gender, racial, and geographic factors and participation in computing education and careers. • Evaluate educational policies and interventions designed to address underrepresentation in informatics and computing, using evidence-based criteria. • Design and justify curricular and pedagogical strategies that explicitly address structural barriers to equitable participation in upper secondary informatics. • Produce a critical policy analysis of a digital equity issue relevant to informatics education, drawing on sociological, educational, and technical literature.
Course content	○ Digital divide theories: access, skills, motivation, and outcomes inequalities ○ Digital capital and the sociology of technology: Bourdieu, intersectionality, and computing ○ Racial and gender disparities in computing education: evidence, explanations, and contested interpretations ○ Algorithmic discrimination: bias in recruitment, credit, criminal justice, and education systems ○ CS for All initiatives: policy design, implementation, and evaluation evidence ○ Culturally responsive pedagogy in informatics: principles, examples, and critiques ○ Rural and low-income school contexts: infrastructure, teacher supply, and curriculum access

	○ Global digital inequality: colonial legacies and technology transfer in the Global South ○ Datafication and surveillance in education: implications for marginalised communities ○ Policy analysis methodology: frameworks for evaluating equity in educational technology policy

Table 83 Description of the elective course Computational Social Science and Digital Research Methods

Course title	Computational Social Science and Digital Research Methods
ECTS	4
Course aims	Develop knowledge and skills in computational social science methods, enabling students to apply computational approaches to social and educational research while addressing methodological and ethical considerations.
Learning outcomes	• Apply computational methods (text analysis, social network analysis, and digital trace data analysis) to questions in social and educational research. • Critically evaluate the epistemological assumptions and methodological limitations of computational social science approaches. • Collect, process, and ethically manage digital trace data (e.g. social media data, web data) in compliance with GDPR and research ethics standards. • Integrate computational social science methods into educational research designs relevant to informatics education. • Communicate findings from computational analyses in academically rigorous and publicly accessible forms.
Course content	○ Computational social science: scope, methods, and epistemological debates ○ Digital trace data: types, sources, and collection methods (APIs, web scraping, platform data) ○ Text analysis: tokenisation, TF-IDF, topic modelling (LDA), and sentiment analysis ○ Social network analysis: graph metrics, centrality, community detection, and educational applications ○ Agent-based social simulation: modelling collective behaviour conceptually ○ Ethical challenges of computational social science: consent, re-identification, and platform terms of service ○ GDPR and digital research: legal frameworks for computational data use in research ○ Replication crisis and reproducibility in computational research ○ Applications in educational research: analysing online learning communities and discussion forums ○ Public communication of computational findings: data journalism and open science principles

Table 84 Description of the elective course Interpretable and Explainable AI in Education

Course title	Interpretable and Explainable AI in Education
ECTS	4
Course aims	Develop students' understanding of interpretable and explainable artificial intelligence, with emphasis on how AI systems make decisions, how such decisions can be critically examined, and how explainability can be addressed in educational contexts and school teaching.
Learning outcomes	• Explain the difference between artificial intelligence, machine learning, interpretable models, and explainable AI. • Analyse how model structure, data quality, and feature selection influence AI outputs and their interpretability. • Apply basic methods for interpreting and explaining model behaviour in accessible ways. • Critically examine bias, fairness, transparency, and accountability issues in educational uses of AI. • Design learning activities that help school students understand how AI systems produce predictions, classifications, or recommendations. • Evaluate the pedagogical value and limitations of explainable AI tools in school and teacher education contexts.
Course content	○ Foundations of interpretable and explainable AI ○ Interpretable vs. black-box models ○ Feature importance, local and global explanations ○ Model behaviour, uncertainty, and error analysis ○ Data quality, bias, and fairness in AI systems ○ Transparency, accountability, and ethical issues in educational AI ○ Explainable AI in learning analytics, tutoring systems, and assessment tools ○ Communicating AI explanations to non-expert audiences ○ Designing classroom activities on AI explainability ○ Critical evaluation of educational AI applications

Table 85 Description of the elective course Designing Accessible Informatics Learning Environments

Course title	Designing Accessible Informatics Learning Environments
ECTS	4
Course aims	Develop competence in designing accessible and inclusive learning environments for informatics education, with emphasis on removing barriers to participation, adapting digital resources, and supporting diverse learners through pedagogically sound and technically appropriate design choices.
Learning outcomes	• Explain key principles of accessibility, inclusion, and universal design in digital learning environments. • Identify barriers that may affect participation in informatics learning for diverse groups of learners.

	• Design accessible digital materials, interfaces, and classroom activities for informatics education. • Adapt programming, computing, and digital making tasks to support learners with different needs and profiles. • Evaluate digital tools and learning environments using accessibility and inclusion criteria. • Plan informatics learning activities that combine accessibility, pedagogical appropriateness, and learner participation.
Course content	○ Accessibility, inclusion, and universal design principles ○ Barriers to participation in informatics education ○ Accessible design of digital materials and online learning environments ○ Readability, multimodality, navigation, and interaction support ○ Accessibility in coding environments, simulations, and educational software ○ Adaptation of programming and physical computing activities for diverse learners ○ Assistive technologies and support tools in digital learning ○ Evaluation frameworks for accessible digital environments ○ Case studies in inclusive informatics teaching ○ Designing and reviewing accessible informatics lessons and resources

Table 86 Description of the elective course Mobile App Development for Education

Course title	Mobile App Development for Education
ECTS	4
Course aims	Develop knowledge and skills for designing and implementing simple mobile applications for educational purposes, with attention to usability, accessibility, learner needs, and pedagogically meaningful classroom use.
Learning outcomes	• Explain core concepts of mobile application development and mobile interaction design. • Design simple mobile applications that respond to educational goals and user needs. • Implement basic mobile app functionality using appropriate development tools and frameworks. • Apply usability, accessibility, and responsive design principles in mobile learning applications. • Integrate multimedia, local data, sensors, or simple APIs into educational mobile apps. • Evaluate the pedagogical affordances and limitations of mobile applications in school contexts.
Course content	○ Introduction to mobile computing and app ecosystems ○ Mobile interface design and user experience principles ○ Mobile app architecture and development workflows ○ Event-driven programming in mobile environments ○ User input, navigation, and screen design

	○ Multimedia integration in educational apps ○ Local data storage and simple cloud connectivity ○ Use of device features such as camera, microphone, location, or sensors ○ Accessibility and usability in mobile app design ○ Prototyping, testing, and evaluation of educational mobile applications

Table 87 Description of the elective course Internet of Things and Smart School Environments

Course title	Internet of Things and Smart School Environments
ECTS	4
Course aims	Introduce students to the Internet of Things (IoT) as a technological and educational domain, and develop their ability to design, prototype, and critically evaluate IoT-based solutions for smart and pedagogically meaningful school environments.
Learning outcomes	• Explain key concepts of the Internet of Things, including sensing, connectivity, data exchange, and automation. • Build simple IoT prototypes using sensors, microcontrollers, and networked communication. • Analyse how IoT systems collect, process, and transmit data in educational and school-related settings. • Evaluate security, privacy, reliability, and ethical issues in smart school environments. • Design classroom projects that use IoT to support inquiry, monitoring, automation, or problem solving. • Critically assess the educational value and practical constraints of IoT solutions in schools.
Course content	○ Foundations of the Internet of Things ○ Sensors, actuators, and embedded devices ○ Microcontrollers and IoT prototyping platforms ○ Network communication and data transmission in IoT systems ○ Data collection, dashboards, and simple automation ○ Smart classrooms and smart school environment applications ○ Environmental monitoring, energy use, and safety-related scenarios ○ Security, privacy, and data protection in IoT ○ Reliability, maintenance, and sustainability considerations ○ Designing educational IoT projects for school contexts

Table 88 Description of the elective course Computer Vision for Educational Applications

Course title	Computer Vision for Educational Applications
ECTS	4
Course aims	Provide foundational knowledge of computer vision and develop students' ability to design simple vision-based applications and learning activities relevant to

	informatics education, while critically addressing technical, pedagogical, and ethical aspects.
Learning outcomes	• Explain core concepts of computer vision, including image representation, feature extraction, and classification. • Use basic computer vision tools and libraries to process images and video. • Implement simple vision-based applications such as object detection, image classification, or tracking. • Analyse the role of data quality, variation, and bias in computer vision systems. • Design educational activities that introduce school students to computer vision concepts and applications. • Critically reflect on privacy, surveillance, fairness, and ethical concerns related to vision technologies.
Course content	○ Introduction to computer vision and image-based AI ○ Digital images, pixels, colour spaces, and image representation ○ Image preprocessing and transformation ○ Feature extraction and pattern recognition basics ○ Image classification and object detection ○ Video processing and tracking fundamentals ○ Use of computer vision libraries and pretrained models ○ Educational applications of computer vision ○ Data quality, bias, and limitations in vision systems ○ Ethical, legal, and societal issues in computer vision

Table 89 Description of the elective course Sociology of Education

Course title	Sociology of Education
ECTS	4
Course aims	Provide an understanding of sociological perspectives on education, enabling students to analyse how social structures and inequalities shape educational processes and outcomes.
Learning outcomes	• Explain key sociological perspectives on education and schooling. • Analyse how social background, inequality, and institutional structures influence educational outcomes. • Critically examine the role of digital technologies in shaping access and participation in education. • Reflect on implications for informatics teaching and school practice.
Course content	○ Education as a social institution ○ Social inequality and educational opportunity ○ Cultural capital and social reproduction (conceptual overview) ○ Digital divide and participation in computing education ○ Gender and diversity in STEM education ○ School systems, policy, and governance ○ Technology and social stratification

	○ Analysis of selected empirical studies

Table 90 Description of the elective course Philosophy of Education

Course title	Philosophy of Education
ECTS	4
Course aims	Provide a foundational understanding of philosophical perspectives on education, enabling students to critically reflect on the aims, values, and ethical dimensions of teaching and learning in contemporary contexts.
Learning outcomes	• Explain central philosophical questions about the aims and purposes of education. • Analyse different conceptions of knowledge and learning. • Examine ethical questions related to digital technologies in education. • Formulate reasoned arguments about the role of informatics in contemporary education.
Course content	○ What is the purpose of education? ○ Knowledge, truth, and justification (introductory epistemology) ○ Education and autonomy ○ Authority, responsibility, and professional ethics ○ Technology and human development ○ Ethical questions in AI and digital learning ○ Normative assumptions in curriculum design ○ Structured philosophical discussion and position paper

Table 91 Description of the elective course Formal Specification and Program Correctness

Course title	Formal Specification and Program Correctness
ECTS	4
Course aims	Build on students' existing knowledge of logic and proof from Mathematics for Informatics to develop a principled understanding of formal specification and program correctness, with emphasis on reasoning rigorously about algorithmic behaviour, loop invariants, and the correctness of programs. Prepare future informatics teachers to explain why programs work (not just how) and to design learning activities that foster this depth of understanding at upper secondary level.
Learning outcomes	• Apply pre- and post-conditions to specify the intended behaviour of functions and procedures. • Use loop invariants to reason formally about the correctness of iterative algorithms. • Prove the partial and total correctness of simple programs using Hoare-style reasoning. • Explain the distinction between testing and formal verification, and the role of each in software development. • Apply formal specification techniques to standard algorithms covered in school informatics curricula (searching, sorting, recursive algorithms).

	• Identify and correct formal specification errors in given programs. • Design learning activities that help school students develop a reasoned understanding of algorithm correctness and the difference between an algorithm that runs and one that is provably correct.
Course content	○ Review of relevant logic: predicates, quantifiers, and mathematical induction (brief, building on Mathematics for Informatics) ○ Pre- and post-conditions: specifying functional behaviour ○ Hoare triples: syntax, semantics, reasoning rules, weakest preconditions, and strongest postconditions ○ Loop invariants: design, verification, and termination arguments ○ Partial vs. total correctness ○ Formal reasoning about standard algorithms: binary search, sorting, recursion ○ Introduction to formal specification languages (e.g. JML, Alloy – conceptual level) ○ Limits of formal methods: undecidability, scalability, role of testing, and connections to static analysis tools ○ Pedagogical applications: teaching algorithmic correctness at upper secondary level, common student misconceptions about why programs are correct

Table 92 Description of the elective course Competitive Programming and Algorithmic Problem Solving

Course title	Competitive Programming and Algorithmic Problem Solving
ECTS	4
Course aims	Develop advanced algorithmic problem-solving skills through competitive programming practice, building on existing knowledge of data structures, algorithms, and graph theory to focus on efficient implementation under contest constraints, advanced problem-solving heuristics, and the pedagogical value of competitive tasks for informatics education.
Learning outcomes	• Apply known algorithmic concepts (sorting, searching, graph algorithms, dynamic programming) effectively under time and memory constraints. • Recognise algorithmic problem patterns and select appropriate strategies rapidly. • Implement correct and efficient solutions to contest-style problems with attention to edge cases and input handling. • Apply advanced techniques not covered in core courses: segment trees, binary indexed trees, advanced string algorithms, and combinatorial optimisation. • Explain the connections between competitive programming tasks and core informatics concepts taught at school level. • Design or adapt competitive programming tasks for school informatics education and Olympiad preparation, with attention to differentiation and gifted learners.
Course content	○ Contest problem-solving strategies: pattern recognition, problem decomposition, and rapid prototyping ○ Implementation discipline: edge cases, input/output handling, and time complexity awareness in contest settings

	○ Advanced data structures: segment trees, binary indexed trees (Fenwick trees), sparse tables ○ Advanced string algorithms: KMP, Z-algorithm, suffix arrays (conceptual and applied) ○ Advanced combinatorics and number theory: modular arithmetic, Euler's theorem, combinatorial counting ○ Network flows and matching: max-flow min-cut, bipartite matching (building on graph theory foundations) ○ Computational geometry basics: convex hull, point-in-polygon, sweep line ○ Competitive programming platforms and contest formats (USACO, IOI, Bebras, Codeforces) ○ Pedagogical value of competitive tasks: engagement, differentiation, and supporting gifted learners ○ Designing and adapting tasks for school informatics competitions and Olympiad training

Table 93 Description of the elective course Bioinformatics and Computational Biology

Course title	Bioinformatics and Computational Biology
ECTS	4
Course aims	Introduce students to computational methods used in biology and life sciences, with emphasis on understanding how informatics drives scientific discovery. Students apply their existing knowledge of algorithms and machine learning to biological domains and develop the ability to design interdisciplinary learning activities that connect informatics with natural sciences in school contexts.
Learning outcomes	• Explain core concepts of bioinformatics, including biological sequences, genomic data, and molecular information. • Apply computational methods for sequence alignment, pattern matching, and similarity analysis to biological datasets. • Use bioinformatics tools and public databases to explore genomic and proteomic data. • Apply and interpret (building on prior machine learning knowledge) classification and clustering methods in biological research contexts. • Identify connections between bioinformatics algorithms and core informatics concepts covered in school curricula. • Design interdisciplinary learning activities that connect informatics and biology for school-level teaching. • Reflect critically on the ethical and societal dimensions of genomic data use, personalised medicine, and data privacy.
Course content	○ Introduction to biological sequences: DNA, RNA, proteins, and their computational representation ○ Sequence alignment algorithms: pairwise alignment (Needleman-Wunsch, Smith-Waterman) and multiple alignment

	○ Pattern matching and regular expressions in biological sequence analysis ○ Genomic databases and public repositories: NCBI, UniProt, Ensembl ○ Phylogenetics and evolutionary trees as graph-based computational structures ○ Application of classification and clustering (building on AI and ML): gene expression analysis, protein function prediction ○ Introduction to structural bioinformatics: protein structure prediction and AlphaFold (conceptual overview) ○ Data visualisation in life sciences: genome browsers, heatmaps, phylogenetic visualisations ○ Ethical and societal dimensions: genomic privacy, data sharing, personalised medicine, and informed consent ○ Cross-curricular connections: designing informatics-biology learning activities for school, including STEM integration examples

Table 94 Description of the elective course Theory of Programming Languages

Course title	Theory of Programming Languages
ECTS	4
Course aims	Develop a deep theoretical understanding of programming languages, their formal semantics, and type systems, with emphasis on the conceptual foundations that underlie all programming paradigms, and on translating this understanding into pedagogically effective teaching at upper secondary level.
Learning outcomes	• Explain the formal syntax and semantics of programming languages using operational and denotational approaches. • Describe the role and design of type systems in ensuring program correctness and safety. • Apply the lambda calculus as a foundational model of computation and function abstraction. • Analyse trade-offs between expressiveness, safety, and complexity in programming language design. • Compare paradigms (functional, object-oriented, logic) from a unified theoretical perspective. • Design learning activities that help school students develop a principled understanding of why programming languages behave as they do.
Course content	○ Syntax and grammars: BNF, context-free grammars, parsing ○ Operational semantics: small-step and big-step evaluation ○ Lambda calculus: syntax, reduction, Church encodings ○ Simply typed lambda calculus and the Curry-Howard correspondence (overview) ○ Type systems: static vs. dynamic typing, type inference, Hindley-Milner (conceptual) ○ Polymorphism, generics, and parametric types ○ Functional programming concepts: higher-order functions, closures, immutability ○ Scope, binding, and environments in programming languages

	○ Memory management from a language perspective: ownership models, RAII, and garbage collection strategies across programming languages ○ Didactic perspectives: teaching programming language concepts at upper secondary level

Table 95 Description of the elective course Parallel and Distributed Systems

Course title	Parallel and Distributed Systems
ECTS	4
Course aims	Develop advanced understanding of the principles, architectures, and challenges of parallel and distributed computing, building on existing knowledge of operating systems and cloud computing. The course focuses on advanced concurrency models and distributed coordination mechanisms not covered in prerequisite courses and prepares informatics teachers to contextualise these topics at upper secondary level.
Learning outcomes	• Explain and compare advanced concurrency models: actor model, communicating sequential processes (CSP), and software transactional memory. • Analyse correctness challenges arising from advanced concurrency: consistency models, distributed consensus, and Byzantine fault tolerance. • Explain how distributed systems achieve agreement and handle failures through consensus protocols such as Paxos and Raft. • Describe parallel programming paradigms (data parallelism, task parallelism, SIMD) and their relationship to modern hardware including GPUs. • Relate parallel and distributed architectures to real-world AI infrastructure: training pipelines, parameter servers, and federated learning. • Design learning activities that introduce upper secondary students to parallelism and distributed system concepts in relevant, contemporary contexts.
Course content	○ Advanced concurrency models: actor model (Erlang/Akka), CSP, software transactional memory, beyond the thread/lock model covered in Operating Systems ○ Consistency models in distributed systems: sequential consistency, linearizability, eventual consistency ○ Distributed consensus: Paxos and Raft (conceptual and applied) ○ Byzantine fault tolerance and its relevance to modern systems (blockchain, overview) ○ Parallel programming paradigms: data parallelism, task parallelism, pipeline, SIMD ○ GPU computing: CUDA concepts, parallelism in neural network training (conceptual level) ○ Distributed training of machine learning models: data parallelism, model parallelism, parameter servers ○ Federated learning and privacy-preserving distributed computation (overview) ○ Scalability patterns: load balancing, sharding, consistent hashing

	○ Pedagogical approaches to teaching concurrency and distributed concepts at upper secondary level

Table 96 Description of the elective course Quantum Computing for Education

Course title	Quantum Computing for Education
ECTS	4
Course aims	Introduce future informatics teachers to the foundational concepts of quantum computing, with emphasis on building a conceptually accurate and pedagogically grounded understanding of quantum information, key quantum algorithms, and the implications for cryptography and computing. The course complements the cryptographic perspective of Cybersecurity by focusing specifically on the computational and educational dimensions of quantum technologies, suitable for teaching at upper secondary level.
Learning outcomes	• Explain the fundamental principles of quantum computing: superposition, entanglement, and measurement. • Describe quantum bits (qubits) and quantum gates as a computational model and contrast them with classical bits and logic gates. • Explain the conceptual structure and educational significance of key quantum algorithms, including Shor's and Grover's algorithms. • Discuss the implications of quantum computing for current cryptographic systems and digital security, extending the cryptographic foundations covered in Cybersecurity. • Use basic quantum circuit simulators to explore quantum computation at a conceptual level. • Design learning activities that introduce upper secondary students to quantum computing concepts in an accessible and scientifically accurate way. • Reflect critically on the societal, ethical, and security implications of advances in quantum computing.
Course content	○ Classical vs. quantum information: bits, qubits, probability amplitudes, superposition, entanglement, and measurement ○ Quantum gates and quantum circuits: Hadamard, CNOT, Toffoli (conceptual level and circuit diagrams) ○ Quantum parallelism and interference ○ Grover's algorithm: quadratic speedup for search (conceptual and visual) ○ Shor's algorithm and implications for RSA cryptography, building on cryptographic foundations from Cybersecurity ○ Quantum key distribution and post-quantum cryptography: BB84, lattice-based algorithms, NIST standardisation process ○ Quantum computing hardware: ion traps and superconducting qubits (overview) ○ Quantum circuit simulators and educational tools (Qiskit, Quirk); societal and ethical implications

	○ Teaching quantum computing at upper secondary level: analogies, misconceptions, and classroom approaches

Table 97 Description of the elective course Natural Language Processing and Text Analysis

Course title	Natural Language Processing and Text Analysis
ECTS	4
Course aims	Develop a principled understanding of the full NLP pipeline, from classical text processing methods to modern neural approaches. Building on prior knowledge of generative AI, the course goes deeper into the linguistic, statistical, and algorithmic foundations of language technology, with emphasis on domain-specific NLP applications, critical evaluation of language systems, and preparing informatics teachers to teach NLP concepts at upper secondary level.
Learning outcomes	• Explain classical NLP representations (bag-of-words, TF-IDF, n-grams) and their role as foundations for modern approaches. • Describe word embedding models and explain how distributional semantics captures meaning computationally. • Explain sequence modelling approaches (n-gram language models, hidden Markov models, recurrent architectures) and their relationship to transformer-based models covered in prior coursework. • Critically analyse domain-specific NLP applications: sentiment analysis, named entity recognition, machine translation, and text summarisation. • Evaluate NLP systems with respect to bias, fairness, hallucination, and reliability, beyond the introductory treatment in Generative AI in Education. • Design learning activities that help upper secondary students understand how language models work, critically evaluate AI-generated text, and apply NLP tools responsibly.
Course content	○ Classical NLP pipeline: tokenisation, text normalisation, bag-of-words, TF-IDF, and n-gram language models ○ Word embeddings: word2vec, GloVe, fastText (distributional semantics and vector arithmetic) ○ Sequence models: hidden Markov models and recurrent neural networks (conceptual overview, connecting to transformer knowledge from Generative AI in Education) ○ NLP-specific evaluation: BLEU, ROUGE, and perplexity (beyond general ML metrics) ○ Domain NLP tasks: sentiment analysis, named entity recognition, and information extraction ○ Machine translation and text summarisation: from phrase-based to neural approaches

	○ Systemic bias in NLP: gender bias in embeddings, training data representation gaps, and debiasing methods, deepening the treatment from Generative AI in Education ○ Reliability and hallucination in production NLP: detection methods, mitigation strategies, and implications for educational deployment, building on prior coursework ○ Teaching NLP at upper secondary level: classroom tools, critical AI literacy, and demystifying language models

3.2.4 Concluding Remarks

The Master's programme provides advanced academic and professional preparation for teaching informatics at upper secondary level and for assuming broader educational responsibilities within schools. Through the integration of advanced disciplinary knowledge, specialised didactics, research competence, and extended teaching practice, graduates develop the capacity to teach complex informatics concepts with conceptual clarity, pedagogical sensitivity, and ethical awareness.

The programme reflects the functional approach promoted by the Digital First project by emphasising meaningful application of informatics knowledge, critical engagement with emerging technologies, and responsible digital citizenship. Graduates are prepared not only to guide students through advanced computing topics, but also to contribute to curriculum development, mentor colleagues, and participate in evidence-based educational innovation. Together with the Bachelor level, the Master's programme completes a coherent 3+2 model for the comprehensive preparation of future-ready informatics teachers in Europe.

4 Conclusion

This deliverable establishes a **foundational structure for the tertiary education of prospective informatics teachers**. The study program was prepared by combining insights from deliverables *D4.1 Informatics teacher education and training in the EU today – research*, *D4.2 Competencies map for informatics teachers in primary school* and *D4.3 Competencies map for informatics teachers in secondary school*, together with an analysis of exemplary university programmes from project partner countries within Task T4.1, as well as partner input regarding national frameworks and accreditation requirements. Based on this combined evidence, the report proposes a coherent 3+2 study programme model for the education of future informatics teachers.

The proposed 3+2 study programme reflects a competence-based and constructively aligned model for the comprehensive preparation of future informatics teachers. It integrates subject-specific informatics knowledge, pedagogical theory, specialised didactics, and progressively structured school practice, ensuring a balanced development of disciplinary, pedagogical, and transversal competences. The intended learning outcomes are explicitly aligned with the competence maps developed in deliverables *D4.2* and *D4.3* and correspond to European Qualifications Framework (EQF) descriptors at Levels 6 and 7. Through a gradual progression from foundational knowledge at Bachelor level to advanced disciplinary, didactic, and research-informed competences at Master level, the programme supports increasing professional autonomy and responsibility. The functional approach promoted by the Digital First project serves as its conceptual backbone, positioning informatics not merely as a technical discipline, but as a meaningful, socially embedded, and pedagogically transformative field within contemporary school education.

Drawing on the national regulatory and accreditation frameworks presented in Annex 1, the proposed 3+2 study programme for future informatics teachers can be most feasibly implemented within established universities that already hold degree-awarding powers in teacher education and operate accredited programmes at EQF Levels 6 and 7. These institutions possess the necessary academic staff, infrastructure, internal quality assurance mechanisms, and experience with external evaluation procedures, making them structurally well positioned to integrate or adapt the proposed programme within existing faculties of education, informatics, or teacher training. In most partner countries, both public universities and accredited private institutions operate under unified national quality assurance agencies aligned with the European Standards and Guidelines (ESG), which provide a stable and transparent framework for programme approval.

To establish such a programme, coordinated engagement is required between university leadership, academic departments (education and informatics), internal quality assurance units, national ministries responsible for higher education, and external accreditation agencies. In several systems, consultation with professional bodies, labour market authorities, and schools is also either required or strongly recommended, particularly to demonstrate societal relevance and workforce demand. Stakeholders are primarily motivated by compliance with national qualification standards, institutional development strategies, labour market needs for qualified informatics teachers, and alignment with European policy priorities related to digital education. For higher education institutions, the programme represents an opportunity to strengthen their academic portfolio in a strategically important field and respond to increasing demand for digitally competent teachers. For ministries and accreditation bodies, the key argument lies in ensuring a structured, quality-assured pathway for preparing informatics teachers in line with national qualification frameworks and European standards, thereby contributing to long-term educational system development.

Based on the accreditation cycles described across partner countries, the establishment of a new study programme typically requires approximately 6-12 months in systems with streamlined procedures, and up to 18 months in cases involving field evaluation or more complex approval processes. Overall, the regulatory environments analysed indicate that the proposed programme is structurally compatible with existing higher education systems and can be realistically introduced within the current legal and quality assurance frameworks.

Looking ahead, the insights presented in this deliverable lay the groundwork for a more coherent and harmonised structure for the education of future informatics teachers in primary and secondary schools. By providing a transferable and quality-assured model, the proposed programme supports the long-term strengthening of informatics education across partner countries and beyond.

In this respect, the deliverable not only fulfils its analytical and design objectives but also provides a strategic reference framework for future curriculum development and policy alignment. The proposed model offers a scalable and transferable solution that can support national reforms and foster greater coherence in informatics teacher education across Europe.

5 References

Brinda, T., Diethelm, I., Dittert, N., Humbert, L., Kramer, M., Losch, D., & Schmitz, D. (2025). *Informatics Competencies for All Teachers - Development of Recommendations for Teacher Education* (pp. 143–154). https://doi.org/10.1007/978-3-031-88744-4_14

Caspersen, M. E., Diethelm, I., Gal-Ezer, J., McGettrick, A., Nardelli, E., Passey, D., Rovan, B., & Webb, M. (2022). *Informatics reference framework for school*. National Science Foundation. https://doi.org/10.1145/3592625

CSTA. (2020). *Standards for Computer Science Teachers*. https://csteachers.org/teacherstandards

Dagiene, V., Hromkovic, J., & Lacher, R. (2021). Designing informatics curriculum for K-12 education: From Concepts to Implementations. *Informatics in Education*. https://doi.org/10.15388/infedu.2021.22

Digital First Consortium. (2025). *D4.1 Informatics teacher education and training in the EU today – research*. https://digitalfirstnetwork.eu/deliverables/

Digital First Consortium. (2025). *D4.2 Competencies map for informatics teachers in primary school*. https://digitalfirstnetwork.eu/deliverables/

Digital First Consortium. (2025). *D4.3 Competencies map for informatics teachers in secondary school*. https://digitalfirstnetwork.eu/deliverables/

European Commission: Joint Research Centre, Redecker, C., & Punie, Y. (2017). *European framework for the digital competence of educators: DigCompEdu*, (Y. Punie, editor) Publications Office. https://data.europa.eu/doi/10.2760/159770

Nardelli, E., Futschek, G., Gal-Ezer, J., & Webb, M. (2025). *A European survey on requirements for the informatics school curriculum and informatics school teachers' education*. Informatics for All. https://www.informaticsforall.org/a-european-survey-on-requirements-for-the-informatics-school-curriculum-and-informatics-school-teachers-education/

Co-funded by the European Union

6 Annexes

Annex 1 – National Frameworks and Accreditation Requirements

This annex provides an overview of how each partner country defines and regulates the launch of new study programmes in higher education, including alignment with occupational and qualification standards. It also summarizes the necessary steps and documentation required for national accreditation. While these two topics are closely related, they are presented together here to offer a clear picture of each partner country's regulatory and procedural context. The descriptions are organized in alphabetical order of the participating countries. At the end of the country-specific sections, a concise summary synthesizes the main features and common elements across all partner countries, providing an integrated view of national frameworks and accreditation practices.

Bulgaria

In Bulgaria, teacher education programmes are offered at both Bachelor's and Master's levels, depending on the qualification route. Bachelor's degree programmes in teacher education are positioned at Level 6 of the Bulgarian National Qualifications Framework for Higher Education and correspond to EQF Level 6, typically comprising 240 ECTS. Master's degree programmes for teacher qualification or subject-specific teacher education are positioned at NQF Level 7 and correspond to EQF Level 7, with a total workload of 300 ECTS (including the preceding Bachelor's degree). The Bulgarian National Qualifications Framework is formally adopted at national level and referenced to the European Qualifications Framework.

The accreditation of higher education programmes is regulated by the Higher Education Act and applies equally to public and private higher education institutions. The responsible governmental authority is the Ministry of Education and Science of the Republic of Bulgaria. Quality assurance and accreditation are carried out by the National Evaluation and Accreditation Agency (NEAA) (https://www.neaa.government.bg), which is a full member of ENQA and listed in the European Quality Assurance Register for Higher Education (EQAR). NEAA is the sole national body responsible for both institutional accreditation and programme accreditation in professional fields and regulated professions, including teacher education. The same accreditation standards, criteria, and procedures apply to all higher education institutions, regardless of their public or private status.

For programme accreditation, higher education institutions must submit a comprehensive self-evaluation report supported by documentary evidence. This includes the legal basis for launching the programme and its alignment with national higher education policy; clearly defined objectives and learning outcomes aligned with the NQF/EQF; detailed curriculum structure with syllabi, ECTS allocation, workload distribution, and teaching and assessment methods; documentation of academic staff qualifications, research output, and teaching experience; and evidence of adequate material resources, infrastructure, and digital and library facilities. Institutions must also demonstrate effective internal quality assurance mechanisms, links between teaching and research or innovation, transparent admission and progression rules, and stakeholder involvement (students, employers, and professional bodies). The requirements are aligned with the ESG standards.

The accreditation process consists of submission of documentation to NEAA, a preliminary compliance review, an external expert evaluation including a mandatory site visit, preparation of an evaluation report with recommendations and a proposed accreditation grade, and a final decision by the Accreditation Council of NEAA. The procedure typically takes between six and twelve months, depending on the complexity of the programme and completeness of the documentation. Programme accreditation is granted for a period of up to six years; where weaknesses are identified, accreditation may be awarded for a shorter period and be subject to follow-up evaluation.

Croatia

In Croatia, teacher education programmes are positioned at higher levels of the Croatian Qualifications Framework (CROQF) and aligned with the European Qualifications Framework (EQF). Integrated undergraduate–graduate teacher education programmes (e.g., for primary and secondary school teachers) typically correspond to CROQF Level 7 and EQF Level 7, equivalent to a 300 ECTS Master's qualification. Specialist programmes may be aligned with CROQF Levels 7 or 8 (EQF Levels 7–8), depending on workload and defined learning outcomes. The accreditation system is centralized and ensures alignment with the CROQF and the European Standards and Guidelines (ESG), applying equally to public and private higher education institutions.

The core legislative framework consists of the Zakon o visokom obrazovanju i znanstvenoj djelatnosti (Official Gazette NN 119/22) and the Zakon o osiguravanju kvalitete u visokom obrazovanju i znanosti (Official Gazette NN 151/22). The responsible governmental authority is the Ministry of Science, Education and Youth. External quality assurance and initial accreditation of study programmes are conducted by the Agency for Science and Higher Education (ASHE/AZVO) (https://www.azvo.hr), which operates independently and is listed in the European Quality Assurance Register for Higher Education (EQAR) and is a full ENQA member. Accreditation standards and procedures are uniform for public and private institutions.

In the initial accreditation procedure, the higher education institution submits a formal request to the Ministry (which forwards it to ASHE), including a study programme proposal prepared according to prescribed instructions and annexes, and a qualification standard from the Croatian Registry of Qualifications. Required documentation includes a feasibility study; detailed curriculum (subjects, ECTS allocation, and learning outcomes); proof of adequate facilities and equipment; evidence of sufficient qualified teaching staff (employment contracts with scientific-teaching or teaching staff); proof of secured funding for programme delivery; and an opinion from the Croatian Employment Service confirming labour market alignment. Documentation must be submitted in Croatian (and in English if requested), in written and electronic formats.

The accreditation process begins with submission and a completeness check, followed by appointment of an expert panel that evaluates the programme against defined quality standards (including internal quality assurance, study programme design, teaching process and student support, and teaching capacities and infrastructure). The expert panel prepares a report, on the basis of which the Accreditation Council adopts a reasoned proposal and the Agency issues a decision to grant or deny a licence for programme delivery. The higher education institution may lodge a complaint within 30 days, with the Complaints Committee required to respond within 15 days. A Follow-up Committee monitors implementation during the first delivery cycle, and the institution must adopt an action plan within three months of licensing and subsequently report on its implementation. The request for initial accreditation must be submitted at least one year before announcing student enrolment. While timelines depend on the specific case, the overall process typically lasts 6–12 months. The system is based on granting a licence to deliver the programme, followed by periodic re-accreditation every five years, as well as renewed initial accreditation in cases of substantial programme changes.

Cyprus

In Cyprus, initial teacher education programmes are positioned between Level 6 and Level 8 of the national qualifications framework, depending on the type and academic orientation of the programme. Undergraduate initial teacher training programmes are typically placed at Level 6, while specialised and academic teaching qualifications may extend to Levels 7 and 8, corresponding to Master's and doctoral-level studies. These levels are aligned with the European Qualifications Framework (EQF).

Teacher education programmes, as well as all other higher education programmes in Cyprus, are accredited under a single national quality assurance framework established by the Law on the Quality Assurance and Accreditation of Higher Education and the Establishment and Operation of an Agency on Related Matters (commonly referred to as the CYQAA Law, enacted in 2015). The responsible ministry is the Ministry of Education, Culture, Sport and Youth, whose Department of Higher Education oversees policy but does not conduct accreditation. External evaluation and accreditation are carried out by the Cyprus Agency of Quality Assurance and Accreditation in Higher Education (CYQAA) (https://www.enqa.eu/membership-database/cyqaa-cyprus-agency-of-quality-assurance-and-accreditation-in-higher-education-2/), an independent and autonomous authority established by law. CYQAA is the sole national body responsible for accrediting new and existing programmes, higher education institutions, departments, and distance-learning provision, and it is registered in the European Quality Assurance Register for Higher Education (EQAR). The same accreditation framework applies to all higher education institutions.

Cyprus operates a structured and evidence-based accreditation process. Institutions must first complete internal programme design, including curriculum development, definition of learning outcomes aligned with the CyQF/EQF, staffing plans, and infrastructure planning. A comprehensive application dossier is then submitted to CYQAA, including the official application form, full curriculum and syllabi, staff CVs and qualifications, infrastructure documentation, internal quality assurance policies, budget and sustainability plans, distance-learning documentation (where applicable), and a Self-Assessment Report. After a completeness check, CYQAA appoints an External Evaluation Committee composed of independent academics in the relevant field, a quality assurance specialist, and, where appropriate, a student and professional representative. The committee conducts an on-site or hybrid visit to verify documentation, meeting institutional leadership, academic and administrative staff, and reviewing facilities and digital learning environments.

Following the visit, the External Evaluation Committee prepares a detailed report outlining strengths, weaknesses, compliance with standards, recommendations, and a proposed decision (accreditation, accreditation with conditions, or non-accreditation). The institution has the opportunity to submit a formal response (rejoinder), including clarifications or corrective measures. The final decision is taken by the CYQAA Council and published on the agency's website. Accreditation is typically granted for five years. Accredited programmes are officially recognised and entered into the national register, after which student admissions may begin. Re-accreditation occurs every five years and includes updated documentation, a new evaluation, and a further accreditation decision to ensure continued compliance with national and European quality standards.

Finland

In Finland, teacher education is regulated at national level and is based on university degrees defined in the Decree on the Qualifications Required of Teaching Staff (986/1998). Early childhood teachers complete a Bachelor's degree in educational science (180 ECTS), which qualifies them to work as kindergarten and preschool teachers. Classroom teachers and subject teachers complete a five-year university programme leading to a Master's degree (300 ECTS). All teacher education programmes include pedagogical studies of 60 ECTS and supervised teaching practice, typically implemented in university training schools or other designated schools. In the Finnish National Qualifications Framework (FiNQF), a Bachelor's degree corresponds to Level 6 and a Master's degree to Level 7, aligned with EQF Levels 6 and 7 respectively. A doctorate (FiNQF/EQF Level 8) is relevant for academic careers but is not required for entry into the teaching profession.

External evaluation of higher education in Finland is carried out by the Finnish Education Evaluation Centre (FINEEC) (https://www.karvi.fi/en/about-us), an independent authority operating as a separate unit of the Finnish National Agency for Education. FINEEC is responsible for the national evaluation of education provided by universities and

universities of applied sciences. According to available information, accreditation of degree programmes by FINEEC is voluntary and does not determine the legal right to award degrees. The Ministry of Education and Culture (https://okm.fi/en/frontpage) is responsible for higher education policy and oversight. For teacher education specifically, programmes must comply with the Decree on the Qualifications Required of Teaching Staff (986/1998). No separate accreditation requirements for public versus private higher education institutions are indicated in the provided information.

Based on the available FINEEC accreditation model, institutions seeking accreditation submit a self-assessment report and supporting evidence prior to a site visit. The self-assessment report (maximum approximately 40 pages, excluding appendices) is submitted in accordance with a timeline agreed between the institution and FINEEC, typically at least four weeks before the review visit. In addition, institutions prepare an electronic evidence room containing representative samples of final theses across grade levels, assessed examinations and assignments, examples of student feedback and summaries at programme and course level, and recent research or RDI publications relevant to the degree programme. The accreditation evaluates the organisation, planning, implementation, and development of the programme, and whether its learning outcomes correspond to defined reference outcomes.

The accreditation process includes: a request by the higher education institution; a formal agreement between FINEEC and the institution defining scope, timeframe, language, publicity, and costs; appointment and training of an independent review team (including academic, professional, and student members); preparation of the self-evaluation report and evidence; and a site visit of at least two days involving meetings with programme management, academic and support staff, students, alumni, and external stakeholders. The review team prepares a report and recommendation, which the institution may check for factual accuracy. The final decision is taken by the relevant FINEEC committee and defines the validity period of the accreditation, including any conditions and related deadlines. The accreditation result and report are published by FINEEC. The validity period is defined case by case in the accreditation decision.

Greece

Greece applies the Hellenic Qualifications Framework (HQF), which comprises eight levels and is aligned with the eight levels of the European Qualifications Framework (EQF). Higher education qualifications are positioned at HQF Levels 6–8. In practice, initial teacher education provided through first-cycle university degrees is typically awarded at HQF Level 6 / EQF Level 6, subject to applicable professional requirements. Postgraduate teacher education and education specialisations are positioned at HQF Level 7 / EQF Level 7, while doctoral studies in education correspond to HQF Level 8 / EQF Level 8.

Quality assurance and accreditation in Greek higher education were introduced by Law 3374/2005 and further developed into a national accreditation system by Law 4009/2011. The current national external quality assurance body is the Hellenic Authority for Higher Education (HAHE / ΕΘΑΑΕ), established by Law 4653/2020, which succeeded the former agency. HAHE operates under the legality oversight of the Ministry of Education, Religious Affairs and Sports and is registered in the European Quality Assurance Register for Higher Education (EQAR). Traditionally, higher education institutions in Greece have been public. A more recent framework, introduced by Law 5094/2024, regulates the licensing, establishment, and operation of non-profit branches of foreign universities in Greece; programme-level quality assurance remains subject to external evaluation and certification, with HAHE responsible for programme accreditation.

For programme accreditation, the higher education institution submits an Accreditation Proposal prepared by its internal Quality Assurance Unit (QAU/MODIP), demonstrating compliance with HAHE standards and supported by evidence. The proposal (typically 40–50 pages, excluding annexes) includes a structured self-assessment against the relevant standards and is accompanied by key documentation and annexes. Evidence generally covers the academic profile and orientation of the programme, defined learning outcomes and qualifications, curriculum structure, teaching quality and staffing, links to research, and labour-market relevance, in line with HAHE's programme certification criteria.

The main stages of the accreditation process include: submission of a request to HAHE and agreement on terms of reference and timetable (institutions are advised to initiate the process approximately three months before the expected site visit); preparation and submission of the Accreditation Proposal; technical scrutiny by HAHE (with a two-week window for revisions if required); appointment of an External Evaluation and Accreditation Panel (EEAP) from HAHE's expert register; document review and site visit (with guidance recommending at least six weeks between receipt of documentation and the visit); drafting of the Accreditation Report; decision by HAHE's competent council; and publication of the outcome, followed where necessary by follow-up procedures. The overall timeline typically spans several months. Programme accreditation validity is commonly set at five years (earlier decisions sometimes indicated four years, with later adjustments to five).

Italy

In Italy, teacher education programmes are positioned at Level 7 of both the National Qualifications Framework (NQF) and the European Qualifications Framework (EQF). Qualification for teaching in pre-primary, primary, and secondary education requires a Master's degree (Laurea Magistrale or Laurea Magistrale a ciclo unico), which corresponds to Level 7 within the Italian qualifications framework.

Accreditation of higher education programmes is governed primarily by Legislative Decree No. 19/2012, which established the AVA system (Self-assessment, Evaluation, and Accreditation), and further specified by Ministerial Decree No. 1154/2021, defining the technical requirements and standards for the current AVA3 cycle. The responsible ministry is the Ministry of University and Research (MUR – Ministero dell'Università e della Ricerca). External quality assurance is conducted by the National Agency for the Evaluation of Universities and Research Institutes (ANVUR – https://www.anvur.it), which is registered in EQAR and operates in compliance with the ESG. Accreditation requirements apply uniformly to state (public) and legally recognised non-state (private) universities; however, private institutions are subject to additional financial sustainability monitoring to ensure long-term programme delivery.

For programme accreditation, institutions must submit the "Scheda Unica Annuale del Corso di Studio" (SUA-CdS), which serves as the central planning and information document for each study programme. It includes clearly defined educational objectives and learning outcomes, professional profiles, detailed curriculum design with ECTS (CFU) allocation, admission requirements, and assessment methods. A technical self-assessment report (Rapporto di Riesame) must also be provided, documenting stakeholder consultation with the "world of work," student-centred learning approaches, and internal quality assurance processes coordinated by the Presidio della Qualità and Joint Teaching-Student Committees. Evidence of adequate human and structural resources is required, including minimum numbers of permanent professors assigned to the programme (as defined by Ministerial Decree No. 1154/2021), appropriate facilities and laboratories, and administrative support. Financial sustainability must be demonstrated through budgetary plans ensuring full programme delivery, particularly for non-state institutions.

The accreditation process follows the AVA3 model. After internal approval by the Academic Senate, the university submits the proposal digitally through the SUA-CdS platform to ANVUR. ANVUR appoints a Commission of Evaluation Experts (CEV) to conduct a technical review and, where necessary, a site visit. Following a phase in which the university may respond to preliminary findings, ANVUR issues a binding technical opinion. The Ministry of University and Research then adopts the formal decree of accreditation. The process follows an annual cycle aligned with the academic calendar: submissions typically occur between December and February, evaluation between March and June, and ministerial approval by July, allowing programme launch in October. Initial accreditation is granted for one year; programmes are subsequently monitored every three years, while institutional periodic accreditation of universities is conducted every five years to maintain degree-awarding status.

Lithuania

Lithuania applies the Lithuanian Qualifications Framework (LTKS/NQF), consisting of eight levels fully referenced to the European Qualifications Framework (EQF). Teacher education programmes are positioned at NQF/EQF Levels 6–8, depending on the study cycle. Initial teacher education is delivered at first-cycle level and leads to a Professional Bachelor's or Bachelor's degree with a teacher qualification (Level 6). Advanced teacher education is provided at second-cycle level through Master's degree programmes in education sciences (Level 7), while doctoral studies in education correspond to Level 8. Teacher qualifications may be obtained through bachelor studies in education, bachelor studies in another field combined with a pedagogical module of at least 60 ECTS (including at least 30 ECTS of teaching practice and at least 12 ECTS in didactics), or through pedagogical professional studies of 60 or 120 ECTS, in accordance with the Teacher Education Regulation Order No. V-856.

Accreditation of higher education programmes is regulated by the Law on Science and Studies of the Republic of Lithuania, the Law on Education of the Republic of Lithuania, and the Order No. V-835 on the Procedure for External Evaluation and Accreditation of Studies. The responsible governmental authority is the Ministry of Education, Science and Sport of the Republic of Lithuania. External evaluation and accreditation are carried out by the Centre for Quality Assessment in Higher Education (SKVC) (https://skvc.lrv.lt/), which is registered in EQAR and operates in compliance with the ESG. The same accreditation requirements apply to both public and private higher education institutions, which may deliver only accredited programmes within the unified national quality assurance framework.

Under the Study Field Evaluation framework developed by SKVC, external evaluation assesses seven quality areas: study objectives, outcomes and content; links between research (or artistic) activities and studies; student admission and support; study processes, academic achievement and graduate employability; teaching staff; study facilities and learning resources; and study quality management and public information. Institutions must submit a self-evaluation report demonstrating alignment with LTKS/EQF levels and national legal requirements, together with a detailed programme description (curriculum and ECTS structure), evidence of academic staff qualifications and pedagogical experience, documentation on the organisation of teaching practice (particularly relevant for teacher education), material and learning resources, internal quality assurance procedures, and stakeholder involvement. For new or unaccredited fields, ex-ante programme evaluation may be combined with field evaluation, requiring evidence of labour market relevance and sufficient human, material and methodological resources. External evaluation includes expert panels and site visits, leading to a report and recommendation. Accreditation is typically granted for up to seven years (or three years in cases of partial compliance), and the overall process from application to decision generally spans approximately 9–18 months.

Portugal

In Portugal, teacher education follows the Bologna three-cycle structure and is aligned with the National Qualifications Framework (QNQ) and the European Qualifications Framework (EQF). Initial teacher education is organised as a two-step pathway. A first-cycle Bachelor's degree (180–240 ECTS), positioned at NQF Level 6 / EQF Level 6, provides the scientific and academic foundation but does not confer professional qualification for teaching. Professional qualification for access to the teaching profession is obtained through a second-cycle Master's degree in Teaching (90–120 ECTS), positioned at NQF Level 7 / EQF Level 7, which integrates subject knowledge, educational sciences, didactics, and supervised teaching practice. Doctoral programmes (third cycle), positioned at NQF Level 8 / EQF Level 8, focus on advanced research and academic specialisation in education or subject-specific fields and are not required for entry into the teaching profession.

The accreditation of higher education programmes is regulated by Law no. 38/2007 of 16 August (legal framework for quality assurance in higher education) and Decree-Law no. 369/2007 of 5 November, which established the Agency for Assessment and Accreditation of Higher Education (Agência de Avaliação e Acreditação do Ensino Superior – A3ES). The responsible ministry is the Ministry of Education, Science and Innovation (MECI), with the Directorate-General for Higher Education (DGES) responsible for coordination, registration of study programmes, and system oversight. External quality assurance and accreditation are carried out by A3ES (https://www.a3es.pt/pt/), which is registered in the European Quality Assurance Register for Higher Education (EQAR) and operates in compliance with the European Standards and Guidelines (ESG). A3ES is responsible for prior accreditation of new programmes, periodic accreditation of existing programmes, and certification of internal quality assurance systems. The same legal framework and accreditation requirements apply to both public and private higher education institutions.

For prior accreditation of a new study programme, institutions must submit a formal proposal through A3ES's electronic platform, demonstrating compliance with national legislation and ESG principles. The documentation includes a detailed programme description and rationale (objectives, degree awarded, scientific area, alignment with institutional strategy and labour market needs), defined learning outcomes and curriculum structure (study plan, ECTS distribution, duration, teaching and assessment methods) aligned with NQF and EQF descriptors, academic staff qualifications and employment status, and evidence of compliance with legal staffing requirements. For master's and doctoral programmes, research capacity must be demonstrated. Institutions must also provide evidence of adequate infrastructure and resources (facilities, equipment, learning materials, and partnerships for supervised practice where applicable) and describe their internal quality assurance system, including monitoring and student participation mechanisms. Evaluation is conducted by external expert panels, and accreditation may be granted, granted conditionally, or refused.

The accreditation process comprises submission of the proposal, preliminary technical verification of completeness by A3ES, external evaluation by an independent expert panel (Comissão de Avaliação Externa – CAE), issuance of a preliminary report and opportunity for institutional response, and a final decision by A3ES's Management Board. For new programmes, the process typically lasts approximately 6–9 months from submission to final decision, depending on the completeness of documentation and the need for clarifications. Accreditation is normally granted for up to six years and may be granted for a shorter conditional period if specific requirements must be fulfilled. Prior accreditation is mandatory before a new programme can be registered and offered.

Slovenia

In Slovenia, teacher education programmes are positioned at Level 8 of the Slovenian Qualifications Framework (SQF/NQF, out of 10 levels) and correspond to Level 7 of the European Qualifications Framework (EQF, out of 8 levels). Initial teacher education is delivered at Master's level (300 ECTS), reflecting the qualification requirements for entry into the regulated teaching profession.

The accreditation of new higher education programmes is governed by the Higher Education Act (Zakon o visokem šolstvu – ZViS). Overall responsibility for higher education lies with the Ministry of Higher Education, Science and Innovation, while programme accreditation is conducted by the Slovenian Quality Assurance Agency for Higher Education (NAKVIS) (http://www.nakvis.si/), which is listed in the European Quality Assurance Register for Higher Education (EQAR). Final accreditation decisions are adopted by the NAKVIS Council. As teaching is a regulated profession, accreditation of teacher education programmes additionally requires the consent of the Ministry of Education in accordance with Article 32(3) of the Higher Education Act, ensuring compliance with professional qualification standards. The accreditation requirements and procedures are the same for public and private higher education institutions.

Higher education institutions must submit a comprehensive application to NAKVIS using official templates and demonstrate compliance with national legislation and the ESG standards. Required documentation includes a detailed programme description (objectives, learning outcomes, competencies, curriculum structure, and syllabi), justification of academic or professional relevance, and alignment within the relevant scientific or professional field. Evidence of labour market needs or societal demand must be provided, depending on the programme type, together with detailed practical training arrangements where applicable. Institutions must document qualified academic staff with appropriate appointments and achievements, as well as adequate material conditions (premises, equipment, library resources, and access to databases). Internal regulations covering admission, recognition of prior learning, progression, completion requirements, and quality assurance procedures must also be included. All documentation must be evidence-based, and site visits are conducted when deemed necessary by the expert panel.

The accreditation procedure consists of six main stages: submission of the application; formal review of completeness; appointment of an independent expert panel (including subject specialists and a student representative); expert evaluation and preparation of an assessment report (normally within at least three months); opportunity for the institution to respond to the report; and final decision by the NAKVIS Council within three months of receiving complete documentation. Appeals may be lodged with the NAKVIS Appeal Committee. The overall process typically lasts between six and nine months, depending on complexity. Study programmes are accredited without a fixed expiry date but are subject to periodic external evaluation within the institutional re-accreditation cycle. Institutional accreditation is granted for five years (extending to seven years under the amended legislation effective from 9 August 2025), and re-accreditation is required within the prescribed timeframe, including mandatory site visits assessing compliance with quality standards and ESG principles.

Spain

In Spain, teacher education programmes are positioned at different levels of the Spanish National Qualifications Framework for Lifelong Learning (MECU) and the European Qualifications Framework (EQF), depending on the teaching level. Initial teacher education for early childhood and primary education (Grado en Educación Infantil / Educación Primaria, 240 ECTS) corresponds to MECU Level 6 (via MECES Level 2 – Grado) and EQF Level 6. Teacher education for secondary, upper-secondary, vocational education and training (VET), and language teaching requires completion of a one-year University Master's in Teacher Training (Máster Universitario en Formación del

Profesorado, 60 ECTS) following a bachelor's degree; this qualification is positioned at MECU Level 7 (via MECES Level 3 – Máster) and corresponds to EQF Level 7. Spain's MECU framework comprises eight levels aligned with the EQF, while MECES functions as the higher education sub-framework mapped to EQF Levels 5–8.

The legal framework for accreditation includes Organic Law 2/2023 on the University System (LOSU), Royal Decree 822/2021 (regulating verification, monitoring and renewal of accreditation of official degrees), Royal Decree 640/2021, and Royal Decree 99/2011. The responsible ministry is the Ministry of Science, Innovation and Universities. External quality assurance is carried out either by the national agency Agencia Nacional de Evaluación de la Calidad y Acreditación (ANECA) (https://www.aneca.es/en/aneca) or by authorised regional agencies of the Autonomous Communities, all of which must comply with ESG standards and be listed in EQAR. The accreditation procedure is the same for public and private universities offering official degrees; differences relate mainly to institutional recognition rather than programme-level quality assurance.

Accreditation follows a standard three-stage cycle defined in Royal Decree 822/2021: (1) initial accreditation (Verificación), (2) monitoring (Seguimiento/MONITOR), and (3) renewal of accreditation (Renovación/ACREDITA). For new programmes, the university prepares and internally approves a "Memoria para la solicitud de verificación del plan de estudios," structured according to Annex II of Royal Decree 822/2021 (maximum 10,000 words), including programme objectives, curriculum design, learning outcomes, staffing, resources, internal quality assurance mechanisms, and alignment with ESG standards. The proposal undergoes external ex-ante evaluation by ANECA or a regional agency; based on the report, the Council of Universities verifies the degree, after which the relevant Autonomous Community authorises it and it is registered in the national register (RUCT). Once implemented, programmes are subject to mandatory monitoring (at least one report three years after implementation for centres without institutional accreditation) and periodic renewal of accreditation, including an external review and site visit. Accreditation must be renewed within six years for most bachelor (240 ECTS), master's and doctoral programmes, and within eight years for long bachelor programmes (300/360 ECTS). This VERIFICA → MONITOR → ACREDITA cycle is repeated to ensure continued compliance with national legislation and ESG standards.

Summary

The following summary synthesizes the national frameworks and accreditation requirements for teacher education programmes across the ten partner countries (Bulgaria, Croatia, Cyprus, Finland, Greece, Italy, Lithuania, Portugal, Slovenia, and Spain) presented in this annex.

Teacher education programmes across these countries are generally positioned at EQF levels 6–8, depending on the study cycle and qualification. Initial teacher education is typically provided at the Bachelor's level (EQF 6), while subject-specific or professional teacher qualifications generally require completion of a Master's level programme (EQF 7). Doctoral-level studies in education (EQF 8) are offered for advanced academic or research-oriented training but are not typically required for access to the teaching profession. This framework ensures that teacher education aligns with both national qualification standards and the European Qualifications Framework, supporting comparability, professional mobility, and the recognition of qualifications across the European Higher Education Area.

Accreditation procedures in all countries are governed by national legislation and coordinated by the respective ministries of education or higher education, while independent quality assurance agencies conduct the formal evaluation of study programmes. All agencies are registered in the European Quality Assurance Register for Higher Education (EQAR) and follow the European Standards and Guidelines (ESG). Required documentation typically includes a detailed programme description, defined learning outcomes, curriculum structure, staff qualifications,

infrastructure and resources, internal quality assurance procedures, and evidence of stakeholder involvement. The accreditation process generally involves submission and formal review of documentation, external expert evaluation (including site visits when necessary), institutional responses to evaluation reports, and a final decision by the agency or council. Accreditation validity varies by country, usually ranging from five to seven years, with periodic monitoring, re-accreditation, or external evaluation to ensure ongoing quality and compliance with professional and academic standards.

Co-funded by
the European Union

Annex 2 – Distribution of Mandatory Courses in BSc and MSc Study Programme Across Clusters

In the Figure 1 you can see the distribution of mandatory courses in Bachelor's and Master's study programme across different clusters.

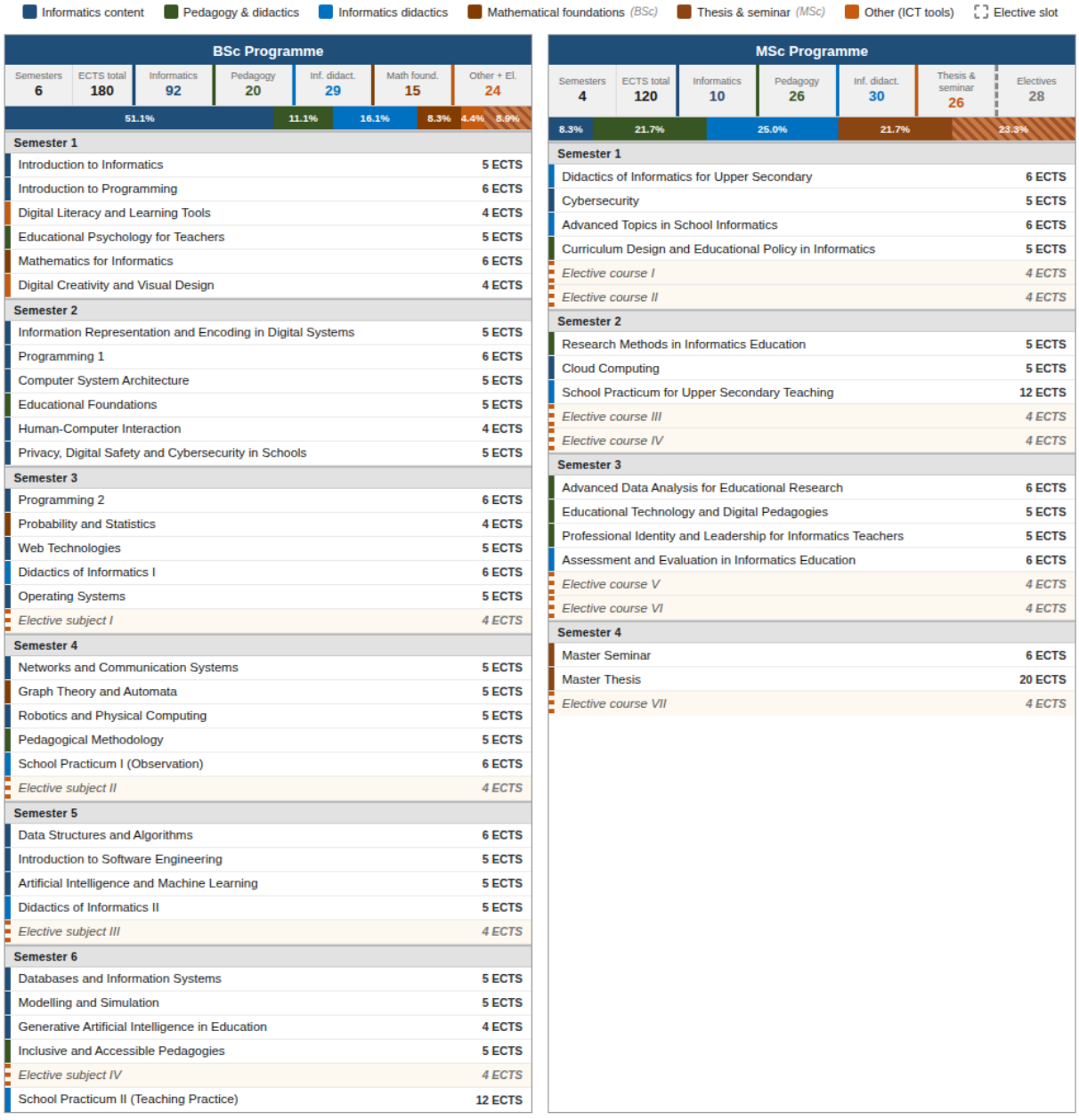


Figure 1 Distribution of Mandatory Courses in BSc and MSc Study Programme Across Clusters

Annex 3 – Distribution of Elective Courses in BSc and MSc Study Programme Across Clusters

In the Figure 2 you can see the distribution of elective courses in Bachelor's and Master's study programme across different clusters.

BSc electives — suitable for primary school teaching (17 courses)
- Technical computing & programming
- Data, AI & digital systems
- Professional & pedagogical skills
- Context, society & history
- Mathematics & sciences

MSc electives — suitable for secondary school teaching (26 courses)
- Pedagogy, curriculum & inclusion
- Advanced AI & emerging technologies
- Research methods & data science
- Philosophy, society & international perspectives
- Foundations, systems & theory

BSc Elective Courses

Suitable for primary school teaching · 17 courses × 4 ECTS · Students select 4 (16 ECTS)

TECHNICAL COMPUTING & PROGRAMMING	
3D Modelling and Digital Fabrication	4 ECTS
Game Design and Development	4 ECTS
Functional and Logic Programming	4 ECTS
Mobile Application Development	4 ECTS
Foundations of Computer Graphics	4 ECTS
DATA, AI & DIGITAL SYSTEMS	
Use of AI in Education	4 ECTS
Data Science and Visualisation	4 ECTS
Internet of Things and Smart Environments	4 ECTS
Learning Analytics and Data-Informed Teaching	4 ECTS
PROFESSIONAL & PEDAGOGICAL SKILLS	
Career Development and Teamwork	4 ECTS
Communication and Digital Pedagogy	4 ECTS
Digital Communication and Professional Skills	4 ECTS
CONTEXT, SOCIETY & HISTORY	
History of Computing and Informatics	4 ECTS
MATHEMATICS & SCIENCES	
Linear Algebra for Computing	4 ECTS
Mathematical Analysis for Modelling	4 ECTS
Physics for Computing	4 ECTS
Introduction to Electronics	4 ECTS

MSc Elective Courses

Suitable for secondary school teaching · 26 courses × 4 ECTS · Students select 7 (28 ECTS)

PEDAGOGY, CURRICULUM & INCLUSION	
Community Engagement through Digital Technologies	4 ECTS
Digital Inequality, Equity, and Informatics Education	4 ECTS
Game-Based Learning and Gamification	4 ECTS
Interdisciplinary Project Design in Informatics	4 ECTS
Teaching Gifted Students in Informatics	4 ECTS
Designing Accessible Informatics Learning Environments	4 ECTS
ADVANCED AI & EMERGING TECHNOLOGIES	
Advanced Topics in AI	4 ECTS
Emerging Technologies in Society	4 ECTS
Human-AI Interaction and Intelligent Interfaces	4 ECTS
Interpretable and Explainable AI in Education	4 ECTS
Computer Vision for Educational Applications	4 ECTS
Internet of Things and Smart School Environments	4 ECTS
Mobile App Development for Education	4 ECTS
Quantum Computing for Education	4 ECTS
Natural Language Processing and Text Analysis	4 ECTS
RESEARCH METHODS & DATA SCIENCE	
Computational Social Science and Digital Research Methods	4 ECTS
Learning Analytics and Educational Data Mining	4 ECTS
Bioinformatics and Computational Biology	4 ECTS
PHILOSOPHY, SOCIETY & INTERNATIONAL PERSPECTIVES	
Comparative Informatics Education: International Perspectives	4 ECTS
Philosophy of Computer Science and Epistemology of Informatics	4 ECTS
Philosophy of Education	4 ECTS
Sociology of Education	4 ECTS
FOUNDATIONS, SYSTEMS & THEORY	
Formal Specification and Program Correctness	4 ECTS
Theory of Programming Languages	4 ECTS
Parallel and Distributed Systems	4 ECTS
Competitive Programming and Algorithmic Problem Solving	4 ECTS

Figure 2 Distribution of Elective Courses in BSc and MSc Study Programme Across Clusters