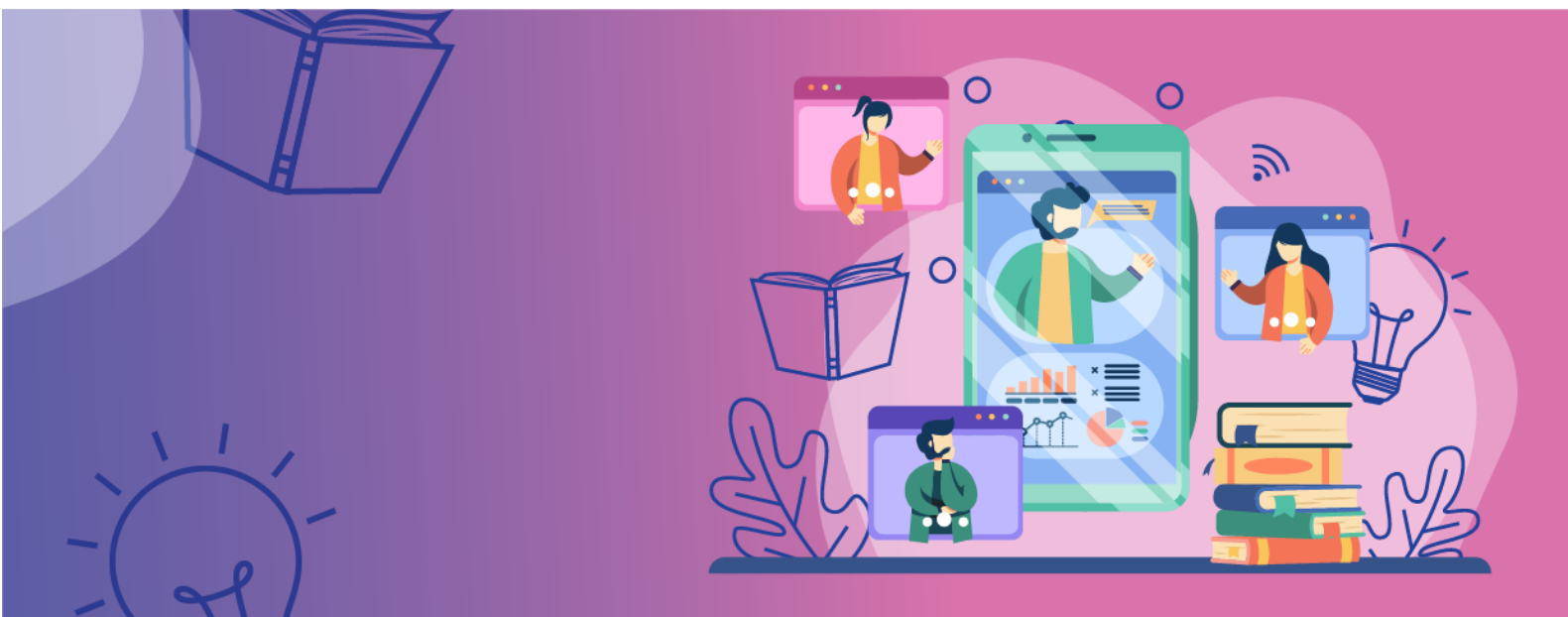




DIGITAL FIRST

Digital Tech as the First Language:
Informatics for Digital Natives

D4.2 Competencies map for informatics teachers in primary school



Co-funded by
the European Union



Project 101132761

LEGAL NOTICE

The information and views set out in this report are those of the authors and do not necessarily reflect the official opinion of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA nor any person acting on their behalf may be held responsible for the use which may be made of the information contained therein.

© Digital First Consortium, 2025

Reproduction is authorized provided the source is acknowledged.



Co-funded by
the European Union

TABLE OF CONTENTS

1	INTRODUCTION.....	5
2	METHODOLOGY.....	6
2.1	The Combined Knowledge and Competency Model.....	6
2.2	Informatics Reference Framework – Informatics4All.....	7
2.3	Digital Competence Framework for Educators – DigCompEdu.....	8
2.4	Findings from WP2 – What We Want Our Children to Learn.....	9
2.5	Co-creative Workshop in Ljubljana	11
2.5.1	<i>Survey Results for ISCED 1 Informatics Teacher Competencies.....</i>	<i>11</i>
2.6	Connection to WP3 – Dialog Clubs.....	13
2.7	Findings from Interviews with Teachers.....	15
2.8	Validation of Competencies.....	17
2.8.1	<i>Results from the Survey.....</i>	<i>17</i>
2.8.2	<i>Survey summary.....</i>	<i>27</i>
3	COMPETENCIES FOR INFORMATICS TEACHERS IN PRIMARY SCHOOL.....	28
3.1	Introduction.....	28
3.2	Subject-Specific Competencies.....	29
3.3	Pedagogical Competencies	31
3.4	Transversal Competencies.....	32
3.5	Glossary of key terms.....	34
4	CONCLUSION.....	37
5	REFERENCES.....	38
6	ANNEXES.....	39
	Annex 1 – Competencies Reviewed in the Co-Creative Workshop.....	39
	Annex 2 – Practical Examples for Subject-Specific Competency Areas.....	41
	<i>Data and Information</i>	<i>41</i>
	<i>Algorithms.....</i>	<i>42</i>
	<i>Programming.....</i>	<i>43</i>
	<i>Computing Systems.....</i>	<i>44</i>
	<i>Digital Creativity and Design</i>	<i>45</i>

<i>Modelling and Simulation</i>	47
<i>Privacy, Safety and Security</i>	48
<i>Artificial Intelligence</i>	49
<i>Computational Thinking (CT)</i>	50

LIST OF TABLES

Table 1 SWOT Matrix Based on Interviews with Primary School Teachers	16
Table 2 Current level of subject-specific competencies of primary school informatics teachers.....	19
Table 3 Desired level of subject-specific competencies of primary school informatics teachers	19
Table 4 Current level of pedagogical competencies of primary school informatics teachers.....	21
Table 5 Desired level of pedagogical competencies of primary school informatics teachers.....	21
Table 6 Current level of transversal competencies of primary school informatics teachers	23
Table 7 Desired level of transversal competencies of primary school informatics teachers	23

LIST OF FIGURES

Figure 1 Roles of the participants	17
Figure 2 Experience in teaching informatics or digital skills to primary school students	18
Figure 3 Comparison between current and desired levels of informatics teachers in primary school for subject-specific competency areas.....	20
Figure 4 Comparison between current and desired levels of informatics teachers in primary school for pedagogical competency areas.....	22
Figure 5 Comparison between current and desired levels of informatics teachers in primary school for transversal competency areas.....	24
Figure 6 Relevance of competencies	25
Figure 7 Clarity of competencies.....	26
Figure 8 Feasibility of competencies.....	26
Figure 9 Integrating knowledge and competency models into a catalogue of competencies for Informatics teachers..	29

1 Introduction

In the rapidly evolving digital landscape, informatics education is increasingly recognized as a fundamental pillar of 21st-century learning. As digital technologies become more embedded in everyday life, it is crucial to equip students with the skills to navigate, create, and critically engage with digital systems rather than remain passive consumers. The Digital First project seeks to enhance informatics teaching by developing innovative, competence-based approaches that align with the needs of digital-native students.

One of the most pressing challenges in informatics education is ensuring that teachers are adequately prepared to guide students in developing computational thinking, problem-solving abilities, and digital literacy from an early age. To address this, Work Package 4 (WP4) – *The Picture of the Teacher* also focuses on defining the core competencies required for informatics teachers, specifically within Task 4.2, which places a strong emphasis on primary school educators (ISCED 1). This deliverable, *D4.2 Competencies map for informatics teachers in primary school*, contributes to this goal by establishing a structured framework that outlines the knowledge, skills, and attitudes essential for effective informatics instruction at the primary level.

Developing this competencies map requires a rigorous, research-based approach that ensures alignment with both educational theory and practical classroom realities. By combining insights from prior research, stakeholder input, and structured pedagogical models, this document not only defines the necessary competencies but also lays the groundwork for teacher education and training initiatives.

The competency framework is structured into three key domains:

- Subject-specific competencies – The foundational informatics knowledge teachers need to deliver curriculum content effectively.
- Pedagogical competencies – The instructional strategies and methodologies required for engaging and effective informatics teaching.
- Transversal competencies – Broader professional and cross-disciplinary skills that support innovation, collaboration, and continuous learning.

Structure of This Document

- Section 2 outlines the methodology, detailing the Combined Knowledge and Competency Model, Informatics Reference Framework, Digital Competence Framework for Educators, connections to other work packages data, and the validation process.
- Section 3 presents the competencies for informatics teachers in primary school, providing a structured overview of essential competencies.
- Section 4 offers the conclusion and summarises key insights.

By establishing clear and well-defined teacher competencies, this deliverable contributes to the broader objectives of the Digital First project: improving informatics teacher education, enhancing teaching quality across European schools, and ensuring that future generations of students are equipped with the digital skills necessary to thrive in an increasingly technology-driven society.

2 Methodology

The development of the competencies map for informatics teachers in primary school is based on a comprehensive and systematic methodological approach that ensures a research-driven and practice-oriented framework. The methodology combines theoretical models, empirical data, and stakeholder engagement, creating a multi-faceted foundation for defining the essential competencies required for primary informatics educators.

To construct a rigorous and applicable competency framework, this deliverable draws from both educational theory and direct input from practitioners. The approach is designed to:

- Ensure alignment with international and European educational standards.
- Incorporate diverse perspectives by engaging multiple stakeholders, including researchers, teachers, and policymakers.
- Balance theoretical and practical needs, ensuring that competencies are conceptually sound and classroom ready.

This methodology is structured around the following key components:

- **The Combined Knowledge and Competency Model**, which serves as the conceptual framework for structuring and categorizing competencies.
- **Informatics Reference Framework** and **Digital Competence Framework for Educators**, which serve as the main frameworks for the catalogue of competencies.
- **Findings from WP2**, which provide insights into student learning objectives and inform the alignment of teacher competencies.
- **A co-creative workshop** in Ljubljana, where project partners collaboratively reviewed, discussed, and refined the competency framework through structured activities and evaluations.
- **Connection to WP3 – Dialog Clubs**, where partner countries organised seven dialog clubs with teachers and discussed about functional approach to teaching informatics.
- **Interviews with teachers**, which will provide direct insights into the realities of informatics teaching and the competencies teachers perceive as most critical.
- **Competency validation**, ensuring that the final framework aligns with European educational frameworks, best practices, and real-world teaching requirements.

This multi-step methodological approach ensures that the competencies map is not only theoretically grounded but also practical, adaptable, and reflective of the evolving needs of informatics education. The following sections detail each methodological component, outlining its role in shaping the final competencies framework.

2.1 The Combined Knowledge and Competency Model

To effectively define the competencies required for informatics teachers in primary school, we employ the *Combined Knowledge and Competency (CKC) Model*, which integrates traditional knowledge-based curricular frameworks with competency-oriented approaches. Historically, curricula have been designed using either a knowledge model – which structures content into knowledge areas, knowledge units, and learning outcomes – or a competency model, which organizes content around tasks and the necessary knowledge, skills, and dispositions required to perform them. The CKC model synergistically combines these two approaches, ensuring that teachers not only acquire structured disciplinary knowledge but also develop practical competencies relevant to teaching informatics in a primary school setting.

Structure of the CKC Model

The CKC model provides a dual perspective on learning:



- The knowledge model organizes content into knowledge areas and knowledge units, ensuring teachers have a well-defined epistemological foundation in informatics. This structure facilitates course design, enabling educators to systematically develop their expertise.
- The competency model emphasizes competency statements that align knowledge, skills, and dispositions with real-world teaching tasks. This ensures that informatics teachers are not only well-versed in theoretical knowledge but can effectively apply it in a classroom environment.

By aligning teaching competencies with a well-structured knowledge framework, the CKC model ensures a balanced approach – providing informatics teachers with both content expertise and pedagogical competence. This is particularly relevant in primary education, where informatics instruction must be engaging, developmentally appropriate, and integrated with broader learning objectives.

Application in Defining Informatics Teacher Competencies

In the context of informatics education at the primary level, the CKC model will be used to map core knowledge areas in informatics to competency-based teaching tasks. For instance, informatics teachers must not only understand computational thinking but also be able to design interactive learning experiences, assess students' digital literacy development, and adapt instructional strategies to diverse learning needs. The CKC model ensures that these teaching competencies are systematically linked to foundational knowledge areas, ensuring both theoretical and practical alignment. Additionally, the CKC model incorporates dispositions, which are essential for fostering an effective learning environment. Dispositions such as collaboration, adaptability, and reflective practice are integrated into competency specifications, ensuring that teachers are not only skilled in informatics but also equipped with the necessary professional attributes to engage and inspire young learners.

Conclusion

By utilizing the CKC model, the competencies defined for primary school informatics teachers are comprehensive, structured, and aligned with both pedagogical and disciplinary requirements. This approach not only strengthens teacher preparation but also supports the long-term goal of fostering high-quality informatics education at the primary school level.

2.2 Informatics Reference Framework – Informatics4All

The *Informatics Reference Framework for School*, developed by the *Informatics for All* coalition, offers a foundational vision for integrating informatics as a key discipline in general education. It advocates for the inclusion of informatics from early primary through to upper secondary education (ISCED levels 1-3), asserting that informatics should hold equal curricular status to mathematics, languages, and the natural sciences.

This vision recognises the essential role informatics plays in today's society – not only as the basis for digital technologies, but as a scientific discipline that fosters problem-solving, innovation, and critical thinking. The framework emphasises that all students should develop a solid understanding of informatics concepts and practices in order to engage meaningfully with the digital world and help shape a fair and inclusive digital society.

The *Informatics Reference Framework* outlines five overarching aims of informatics education. These include enabling students to (1) use digital tools responsibly and creatively, (2) understand fundamental informatics phenomena and principles, (3) apply algorithmic thinking to problem-solving, (4) model and analyse real-world systems, and (5) reflect critically on the ethical and social implications of informatics. These aims inform the development of both student learning outcomes, and the teacher competencies needed to support such outcomes.

At the core of the framework is a structured set of twelve topic areas:

- Data and information – understanding how data is collected, structured, analysed, and used to represent and model real-world scenarios.
- Algorithms – developing, specifying, and evaluating algorithms as tools for problem-solving.



- Programming – using programming languages to express computational solutions and create digital artefacts.
- Computing systems – exploring how systems operate, their components, and limitations.
- Networks and communication – understanding how digital devices communicate and how networks influence functionality and risk.
- Human-computer interaction – investigating the interface between humans and computing systems to improve usability and accessibility.
- Design and development – planning, creating, and iteratively improving digital artefacts based on stakeholder needs and contextual understanding.
- Digital creativity – expressing ideas and constructing digital content across media types.
- Modelling and simulation – using digital tools to explore and understand real or imagined phenomena.
- Privacy, safety and security – recognising risks in the digital environment and strategies for protection.
- Responsibility and empowerment – evaluating the social and ethical impact of informatics systems and promoting critical engagement.

These topic areas are intended as flexible reference points that can be adapted to national contexts. They reflect a coherent vision of informatics as both a rigorous academic subject and a way for learners to better understand and shape the digital world.

Within the context of the *Digital First* project and this deliverable, the *Informatics Reference Framework* serves as a key source for identifying subject-specific content knowledge that informatics teachers should possess. It also shapes pedagogical and transversal competencies by highlighting the ethical, social, and interdisciplinary aspects of informatics education. The framework's emphasis on data literacy, computational thinking, design skills, and digital citizenship resonates strongly with the findings from WP2, which outlined what we want our children to learn, and with the functional approach explored in WP3's Dialog Clubs. Together, these inputs guide the articulation of teacher competencies that are not only technically sound but also based on strong values and suited to how young children learn best.

Finally, the *Informatics Reference Framework* supports a two-tier strategy for informatics education: positioning informatics both as a standalone discipline with its own conceptual foundations, and as a cross-cutting enabler that enriches learning in other subjects. This dual function requires teachers who are not only well-experienced in the discipline itself but are also capable of collaborative and interdisciplinary teaching.

In summary, the framework highlights the importance of creative and responsible use of digital technologies, interdisciplinary learning, and the development of computational thinking. These priorities are fully aligned with the goals of WP4, which aims to equip informatics teachers with the competencies needed to teach informatics.

2.3 Digital Competence Framework for Educators – DigCompEdu

The *European Framework for the Digital Competence of Educators* (DigCompEdu) offers a structured and widely adopted model for describing the digital competencies educators need in the context of teaching and learning. It responds to the growing importance of digitalisation in education and supports national and institutional efforts to promote professional development in this area. As such, it serves as a highly relevant reference for the *Digital First* project and the development of this deliverable.

DigCompEdu focuses on how digital technologies can be used not only to enhance teaching practices, but also to support teacher collaboration, facilitate communication with stakeholders, and empower learners to become competent, critical users of digital tools. The framework outlines six key areas of competence:

1. Professional engagement – using digital tools for communication, collaboration, and professional development.



2. Digital resources – sourcing, creating, and sharing digital content responsibly and effectively.
3. Teaching and learning – planning and implementing digital-supported instruction and learning processes.
4. Assessment – using digital tools to monitor learner progress and adapt teaching.
5. Empowering learners – tailoring digital use to different learner needs and promoting active participation.
6. Facilitating learners’ digital competence – enabling students to develop their own digital skills safely and critically.

These areas are further broken down into 22 specific competencies, covering aspects such as accessibility, differentiation, learner feedback, and responsible digital use. Each competence is described across six proficiency levels, from A1 (Newcomer) to C2 (Pioneer), providing a nuanced progression model that recognises different stages of development and helps educators reflect on their digital growth.

Within the context of this document, the DigCompEdu framework is instrumental in analysing how informatics teachers in primary schools use digital technologies in their teaching and professional practice. The structure of DigCompEdu complements the three-part competency model in this deliverable – subject-specific, pedagogical, and transversal competencies – by offering a set of descriptors that help to locate teacher practices within a broader European benchmark.

DigCompEdu is particularly valuable in understanding pedagogical and transversal competencies. For example, the framework’s emphasis on using digital tools for inclusion and personalisation directly supports the pedagogical dimension of teaching informatics to diverse learners in primary school. Similarly, its focus on professional engagement and collaboration aligns with the transversal skills, such as cooperation among colleagues, openness to innovation, and participation in professional learning networks.

Finally, DigCompEdu promotes an understanding of digital competence that is not limited to tool use, but rooted in reflective, ethical, and inclusive teaching. This mirrors the project’s emphasis on fostering not just technical skills, but also the ability to use digital technologies in ways that promote critical thinking, creativity, and responsible citizenship among students. As such, DigCompEdu reinforces the project’s vision of informatics education that is meaningful, learner-centred, and future-ready.

2.4 Findings from WP2 – What We Want Our Children to Learn

Work Package 2 (WP2) explored what is taught and learned in informatics education across European countries, how it is taught, and how stakeholders perceive its effectiveness. The findings from WP2, particularly from Deliverables *D2.2 Transnational Research* and *D2.3 Recommendations*, provide crucial insights that inform the development of competencies for informatics teachers in WP4. These insights guide the alignment between student learning goals and teacher competencies, ensuring that educators are well-equipped to teach the informatics knowledge and skills students need.

Key Findings on What Students Learn in Informatics

WP2 revealed significant variations in how informatics education is structured across different countries. However, certain core content areas were consistently emphasized:

- Widespread Focus Areas:
 - Data and Information
 - Programming and Algorithms
 - Privacy, Safety, and Security
- Less Consistently Covered Topics:
 - Human-Computer Interaction
 - Modelling and Simulation
 - Responsibility and Empowerment



At the primary school level (ISCED 1), teaching mainly focuses on:

- Computing systems, digital creativity, and problem-solving
- Basic programming concepts and logical reasoning
- Limited emphasis on privacy, security, and responsible digital behaviour

This highlights a gap in early informatics education, as critical digital ethics and data privacy skills are not yet integrated into many primary curricula. WP2 suggests that these topics should be introduced earlier to prepare students for responsible and safe engagement with digital technologies (D2.2).

Alignment with WP4 and Teacher Competencies

Since teacher competencies must align with student learning objectives, WP2 findings directly impact the competency framework for informatics teachers in WP4. The catalogue of competencies developed in D4.2 must ensure that teachers are prepared to:

- Teach foundational informatics knowledge while integrating computational thinking across subjects.
- Develop students' problem-solving skills through programming and algorithmic reasoning.
- Instil responsible digital citizenship, covering online safety, data protection, and ethical computing.
- Apply interactive and interdisciplinary teaching strategies, integrating informatics with subjects like mathematics, science, and the arts (D2.3).

Teaching Methods and Pedagogical Competencies

WP2 found that teacher-centred approaches (demonstration, direct instruction) remain dominant in informatics education. However, there is a growing emphasis on student-centred methodologies, particularly:

- Project-Based Learning: Encouraging students to solve real-world problems using informatics.
- Collaborative Learning: Group-based problem-solving and peer-supported activities.
- Multimodal Learning: Combining digital tools, gamification, and interactive teaching strategies.

For WP4, this means that informatics teachers must be trained not only in subject-specific content but also in pedagogical strategies that engage students in active, inquiry-driven learning (D2.3).

Challenges and Opportunities in Informatics Education

WP2 also identified barriers to effective informatics teaching, including:

- Lack of well-trained informatics teachers, particularly in primary education.
- Curriculum fragmentation, with informatics often integrated into broader subjects rather than being taught as a standalone discipline.
- Resource limitations, such as inadequate access to updated software, devices, and infrastructure.

To address these challenges, WP4 must ensure that the competency framework in this deliverable, D4.2, supports:

1. Professional Growth of Informatics Teachers – Continuous training and upskilling to keep pace with evolving technologies.
2. Cross-Disciplinary Integration – Enabling teachers to connect informatics with other subjects.
3. Professional Values – Encouraging ethical responsibility, inclusivity, and innovation in teaching informatics (D2.3).

Conclusion

The findings from WP2 serve as a foundation for defining teacher competencies in WP4. The catalogue of competencies in D4.2 must equip teachers to provide high-quality informatics education, bridging existing gaps in student learning and ensuring alignment between teaching practices and digital competency goals. By addressing subject-specific, pedagogical, and transversal competencies, WP4 contributes to raising the overall quality and accessibility of informatics education in primary schools across Europe.



2.5 Co-creative Workshop in Ljubljana

During the in-person meeting in Ljubljana, a co-creative workshop was conducted to collaboratively refine and validate the competencies required for informatics teachers across different educational levels. The workshop was structured to ensure that project partners could systematically review and provide input on the competency framework. To facilitate this process, participants were divided into three groups based on the International Standard Classification of Education (ISCED) levels: ISCED 1 (primary education), ISCED 24 (lower secondary education), and ISCED 34 (upper secondary education).

Before engaging in a detailed review of the competency lists, all project partners completed a survey designed to assess both the importance of each competency and the required proficiency level expected of an effective informatics teacher. The survey aimed to establish a shared understanding of the significance and depth of expertise needed for each competency. Participants were asked to rate the importance of each competency on a scale from 1 to 4, with 1 representing "Not Important" and 4 representing "Very Important." Additionally, they provided a proficiency rating on a similar 1 to 4 scale, where 1 indicated a basic awareness was sufficient and 4 signified an advanced, expert level mastery was necessary. These ratings allowed for a quantitative evaluation of how essential each competency is for teaching informatics at different ISCED levels, as well as the expected level of mastery for teachers in their respective educational contexts.

The competencies included in the survey were aligned with Deliverable D2.2, which defines what we want our children to learn at different educational levels in informatics. By first assessing the perceived relevance and proficiency requirements of these competencies, the workshop ensured that discussions were grounded in a structured evaluation process rather than solely subjective opinions. The results of the survey provided a baseline for the subsequent group discussions, helping to identify areas of consensus and divergence among project partners.

Following the survey, each group received printed versions of the competency lists, which were structured into three key categories: subject-specific competencies, focusing on informatics content knowledge; pedagogical competencies, covering instructional methodologies and strategies; and transversal competencies related to professional growth, cross-disciplinary integration, and professional values. Each group engaged in collaborative discussions, critically analysing the proposed competencies in light of the survey results. They deliberated on whether certain competencies should be revised, merged, specified or expanded, ensuring that the framework accurately reflects the needs and expectations for informatics teachers.

Throughout these discussions, participants shared insights from their national and institutional perspectives, highlighting variations in how informatics is taught across different contexts. Their feedback was carefully documented, and key suggestions were noted for further refinement of the competency framework. By combining structured evaluation through the survey with in-depth qualitative discussions, the workshop provided a comprehensive and co-creative approach to defining competencies for informatics teachers. The outcomes of this collaborative effort will contribute to the finalized catalogue of competencies, ensuring that it is both theoretically robust and practically applicable across diverse educational settings.

2.5.1 Survey Results for ISCED 1 Informatics Teacher Competencies

To assess the relevance and mastery levels required for informatics teachers in primary education, a survey was conducted with project partners. Participants rated each competency on two dimensions:

1. Importance for effective teaching (Scale: 1 = Not Important, 4 = Very Important)
2. Required proficiency level (Scale: 1 = Basic Awareness, 4 = Advanced Mastery)

The survey covered three key competency domains:

- Subject-Specific Competencies (Core informatics knowledge and digital technologies)
- Pedagogical Competencies (Teaching methods, assessment, and inclusion)
- Transversal Competencies (Professional growth, cross-disciplinary integration, and professional values)



Key Insights from the Survey

1. Alignment Between Importance and Proficiency

For most competencies, importance ratings closely align with required proficiency levels, meaning that the more important a competency is, the higher the proficiency level expected for teachers. However, some notable discrepancies stand out ($\text{difference} = |\text{importance} - \text{required proficiency}|$):

- Understanding data and information (representation, organization, analysis) ($\text{difference} = 0,67$)
- Fostering computational thinking mindset in young learners ($\text{difference} = 0,56$)
- Knowledge of algorithms and computational thinking principles ($\text{difference} = 0,44$)
- Understanding basic programming concepts and creating simple programs ($\text{difference} = 0,44$)
- Sharing experiences with other primary informatics teachers ($\text{difference} = 0,44$)

These discrepancies suggest that while certain competencies are recognized as critical for informatics teaching at the primary level, teachers are not necessarily expected to master them at an advanced level. Instead, the focus appears to be on familiarity with concepts, effective instructional strategies, and collaboration with peers rather than deep technical expertise. This highlights the need for continuous professional development, ensuring that teachers receive targeted support in key areas without unrealistic expectations of technical mastery.

2. Top-Rated Competencies

The highest-rated competencies in both importance and required proficiency focus on pedagogical effectiveness rather than technical mastery. The top competencies include:

- Ability to design age-appropriate learning activities (Highest importance and proficiency)
- Skills in implementing active and experiential learning
- Ability to facilitate collaborative learning
- Ability to adapt content for different learning needs

These findings emphasize the crucial role of strong teaching methodologies in primary informatics education, ensuring that students engage with computing concepts in an accessible and meaningful way.

3. Competencies with Lower Ratings

Competencies that received the lowest importance ratings include:

- Knowledge of computer systems and networks fundamentals
- Competence in evaluating practical programming skills
- Basic knowledge of computer graphics and multimedia

While these are still considered relevant, their lower importance scores suggest that primary level informatics teaching prioritizes computational thinking and pedagogy over deep technical expertise in computing systems.

4. Subject-Specific vs. Pedagogical Focus

The results reinforce the idea that primary informatics teachers do not need to be technical experts but must be skilled in delivering informatics concepts effectively. Subject-specific competencies (e.g., data science, programming fundamentals) received mixed importance ratings, whereas pedagogical competencies were consistently rated highly.

Conclusion

The survey results highlight that strong pedagogical skills are more critical than deep informatics expertise at the primary level. The most valued competencies focus on effective teaching strategies, student engagement, and adaptive learning methods, while more specialized computing topics such as robotics, networks, and multimedia are considered secondary priorities. These insights will inform the finalized competency catalogue, ensuring that it aligns with real-world expectations for primary school informatics teachers.

2.6 Connection to WP3 – Dialog Clubs

The insights gathered from the *Dialogue Clubs (WP3)* provide valuable context for understanding the broader roles that digital technologies play in children's lives. Each Dialogue Club explored one of the seven language functions based on Halliday's framework, uncovering how students interact with digital technologies to express themselves, access information, collaborate with others, solve problems, and shape their digital presence. These findings are highly relevant for identifying key competencies that informatics teachers in primary schools should possess. By connecting each language function to the teaching of informatics, we can better understand how digital literacy, programming, media creation, and safe digital engagement should be supported from the earliest stages of formal education. The following sub-sections present short summaries of each function and outline how their results inform the competencies for informatics teachers in primary school.

Personal function

The findings from the Dialogue Club on the *personal function* of digital technologies highlight the importance of supporting students in expressing their emotions, identities, and personal experiences through digital means. For primary school informatics teachers, this underscores the need for competencies related to teaching media literacy, guiding the safe and responsible use of social media, and fostering students' understanding of digital identity. Teachers must be able to design age-appropriate learning activities that support personal storytelling and creative expression, use digital tools to promote emotional literacy, and help students reflect on their online behaviours. They should also be equipped to explain the implications of social networks in age-appropriate ways and manage issues around online privacy and digital safety. Additionally, fostering a classroom culture that values authenticity, self-awareness, and respectful communication is key to supporting children's emotional and social development in digital contexts.

Representational/informative function

The Dialogue Club on the *representational/informative function* of digital technology emphasised how children use digital tools to organise, structure, and communicate factual information. For primary school informatics teachers, this highlights the need for competencies in teaching data representation and processing, supporting the development of digital presentations and infographics, and fostering critical thinking when searching for and verifying online information. Teachers should be equipped to introduce age-appropriate tools for creating visual content, guide students in safe and effective information handling (such as using search algorithms and mind maps), and promote the responsible use of AI. They also need pedagogical skills to scaffold student learning through storytelling, interactive activities, and collaborative projects, as well as transversal skills to help students present ideas clearly, explain concepts, and reflect on their own digital outputs.

Interactional function

The Dialogue Club on the *interactional function* of digital technology showed how children use digital platforms to connect with peers, collaborate on projects, and maintain social relationships. These insights point to the importance of informatics teachers fostering digital communication and collaboration skills in primary education. Teachers should be able to design inclusive activities that promote peer interaction and respectful digital behaviour and introduce age-appropriate tools that support group work and shared creation. Competencies in managing digital classroom platforms, guiding online etiquette, and promoting balanced digital habits are essential. Moreover, teachers should help students distinguish between private and school digital identities and address the emotional and social dimensions of online interaction – supporting a safe, ethical, and socially enriching use of digital tools from the earliest years.

Instrumental function

The Dialogue Club on the *instrumental function* of digital technology demonstrated how children use digital tools to express needs, solve problems, and make decisions in everyday and learning contexts. These findings emphasise the importance of fostering problem-solving and computational thinking skills in primary education. Informatics teachers need the ability to introduce digital tools that support children's autonomy, such as classroom communication platforms, collaborative apps, and visual programming environments. Teachers should also help students break down



problems, evaluate information sources (including AI-generated content) and experiment with creative and practical solutions. Building digital literacy, including safe and effective use of virtual assistants, messaging tools, and online collaboration platforms, is key. Furthermore, teachers must nurture emotional awareness and responsible behaviour in digital environments, enabling students to use technology not only efficiently, but also ethically and reflectively.

Imaginative function

The Dialogue Club on the *imaginative function* of digital technology explored how children use digital tools for storytelling, role-play, and creative expression. Platforms like Pixton, Scratch, Minecraft, and Book Creator enable students to create digital narratives, fostering imagination, collaboration, and digital literacy. Teachers should support students in using age-appropriate tools, guide creative processes, and balance digital and traditional storytelling methods. Digital creativity also promotes cognitive and emotional development by enhancing planning, problem-solving, and emotional expression. Informatics teachers need competencies in designing inclusive, cross-curricular projects, integrating AI tools thoughtfully, and modelling ethical digital behaviour. Supporting students in sharing their creations safely through platforms such as Padlet, Scratch, or Google Classroom helps build confidence and digital citizenship. By encouraging creative exploration and responsible sharing, teachers can make informatics a space for innovation, engagement, and meaningful learning.

Heuristic function

The Dialogue Club on the *heuristic function* of digital technology highlighted how children use digital tools to ask questions, explore new topics, and discover information through self-directed learning. Students engage in research and content creation, often within project-based and inquiry-driven activities that promote curiosity, critical thinking, and resilience. Teachers play a key role in guiding structured exploration, fostering digital literacy, and modelling how to evaluate information sources effectively. To support heuristic learning, informatics teachers need the ability to scaffold discovery-based activities, promote reflective thinking, and integrate digital inquiry with real-world contexts. This includes encouraging safe and critical use of AI tools, supporting collaboration, and helping students connect findings across disciplines. By cultivating an environment where curiosity is valued and mistakes are seen as part of learning, teachers empower students to become independent, thoughtful, and confident digital learners.

Regulatory function

The Dialogue Club on the *regulatory function* of digital technology examined how children use digital tools to manage group dynamics, lead collaborative tasks, and influence others' behaviour. Students take on leadership roles in both formal and informal contexts, using digital environments to delegate tasks, enforce norms, and model responsible online behaviour. These spaces can empower quieter students and foster communication, teamwork, and ethical awareness. To support this, informatics teachers need to guide students in digital self-regulation, online conduct, and collaborative decision-making. They should introduce structured discussions on digital rights and responsibilities, model ethical technology use, and integrate principles from regulations like the Digital Services Act and AI Act. Teachers must also address challenges such as digital inequality and exclusion. By promoting digital citizenship and leadership, educators can help students navigate and shape digital communities with confidence, respect, and responsibility.

Conclusion

The findings from the Dialogue Clubs confirm that digital technologies are deeply embedded in students' social, emotional, cognitive, and creative experiences. Each explored function highlights how students use digital tools not only to express themselves and solve problems, but also to collaborate, lead, and explore the world around them. These diverse forms of engagement underline the need for informatics education in primary schools to address more than just technical skills. Informatics teachers must be prepared to foster ethical awareness, support creativity and autonomy, and create inclusive, inquiry-driven learning environments. By drawing on these insights, we can ensure that primary informatics teaching equips students with the competencies they need to participate confidently, responsibly, and reflectively in an increasingly digital society.

2.7 Findings from Interviews with Teachers

This chapter presents selected findings from teacher interviews conducted as part of Task 4.1, *Informatics teacher education and training in the EU today – research*, within the *Digital First* project. For the purposes of this deliverable, only those findings that are directly relevant to the development of the catalogue of competencies have been included.

Overview of Key Findings

A **major distinction** across countries is whether informatics is delivered as a **standalone subject** or **integrated across disciplines**. In countries such as **Bulgaria, Cyprus, Finland, Italy, Spain, and Portugal**, informatics is most commonly integrated into subjects like mathematics, science, language, environmental studies, or broader STEAM-based activities. Here, informatics is not explicitly timetabled but emerges through thematic projects and classroom practices, such as building circuits in science, storytelling with digital tools in language classes, or using educational robotics in problem-solving contexts. In contrast, **Croatia, Greece, Lithuania, and Slovenia** demonstrate more structured delivery, with informatics being taught as a separate subject in many schools, often supported by dedicated timetables and curricular guidelines. However, even in these contexts, interdisciplinary links are actively encouraged, particularly through project-based work.

Across all countries, teachers reported covering several **core I4All topic areas**, particularly **algorithms, programming, digital creativity, and data and information**. More advanced or abstract areas, such as **human-computer interaction, modelling and simulation, or AI and ethics**, appeared less frequently, often introduced through simplified real-world scenarios or optional enrichment activities. Several countries reported emerging engagement with **robotics, privacy and security, and responsibility and empowerment**, although these are typically framed within age-appropriate, contextualised learning tasks.

Pedagogical approaches shared by teachers include **playful, hands-on learning, project work, gamification, and unplugged activities**, allowing students to explore computational thinking concepts in developmentally appropriate ways. Scaffolding is frequently used, progressing from simple instructions and offline logic tasks to block-based programming with tools such as **Scratch, Scratch Jr, Blockly, and Micro:bit**. Teachers also emphasise **student collaboration, peer mentoring, and creativity** as key components of effective informatics learning in primary school.

The **pathways to teacher knowledge** about informatics varied widely. While some teachers have formal training, especially in countries with more structured informatics provision, many rely on **self-directed learning, online MOOCs, peer support networks, and national or international CPD programmes** (e.g. Erasmus+, CUC, Code Week). In several countries, informatics is taught by generalist primary teachers rather than informatics specialists, which may limit the depth of topic coverage but also encourages integration into general teaching practice.

Finally, a recurring pattern is the **dependence on teacher initiative and school level discretion**, particularly in integrated settings. In the absence of clear national curricular mandates, the inclusion of informatics topics at ISCED 1 level often hinges on available equipment, teacher confidence, and institutional support for experimentation. Where school-wide or national policies support informatics teaching, as in **Croatia, Greece, or Slovenia**, topic coverage tends to be broader and more consistent. Where support is more fragmented, such as in **Italy or Spain**, implementation varies significantly from classroom to classroom.

SWOT Analysis of Primary Level Informatics Teaching

The following table summarises the key strengths, weaknesses, opportunities, and threats identified across countries, based on teachers' responses related to the I4All core topics and interdisciplinary informatics teaching at the primary level.

Table 1 SWOT Matrix Based on Interviews with Primary School Teachers

Strengths	Weaknesses
- Broad coverage of I4All core topics across most countries, especially programming, algorithms, data and information, and digital creativity. - Rich use of interdisciplinary methods in some countries where informatics is meaningfully connected to subjects such as math, science, language, history, and art. - Creative use of hands-on tools (e.g. Scratch, Scratch Jr, Bee-Bot, Micro:bit, LEGO) and project-based learning supports early informatics understanding. - Teachers often develop strong competencies through CPD, MOOCs, and informal learning, despite variability in formal training.	- In countries with no standalone informatics subject, integration is inconsistent and relies heavily on individual teachers' initiative. - Lack of structured progression through I4All topics when informatics is only integrated informally or opportunistically. - Gaps in teacher training result in uneven confidence and coverage, especially in topics like AI, networks, or design & development.
Opportunities	Threats
- Expansion of interdisciplinary teaching (e.g. robotics in math, digital storytelling in languages, ethics and privacy in social sciences) can further strengthen connections with the I4All framework. - Continued development of national or regional CPD programmes aligned with I4All framework. - Growing interest in AI, ethics, and robotics offers new entry points for cross-curricular teaching.	- Over-reliance on informal integration may lead to fragmented or shallow exposure to core informatics topics. - Lack of a structured national curriculum in some countries prevents coherent, equitable informatics education for all pupils. - Rapid digital evolution outpaces the ability of teachers and curricula to keep up, risking outdated or incomplete topic coverage.

Conclusion

At ISCED 1 level, informatics is most commonly integrated into broader subjects such as mathematics, science, language, and general STEAM activities. In a few countries, it is also offered as a separate subject, typically as an elective in the later primary years. Across both models, the teaching of informatics is closely aligned with selected areas from the **I4All framework**, most notably **algorithms, programming, digital creativity, data and information**, and **privacy, safety and security**. Topics such as **responsibility and empowerment, design and development**, and **human-computer interaction** are less frequently addressed and usually emerge through informal exploration or teacher initiative.

Interdisciplinary approaches are widely used, allowing informatics to be meaningfully embedded in subject-specific contexts. Examples include:

- Programming robots to solve math or geometry tasks,
- Using Scratch or Scratch Jr to simulate science processes, and
- Creating digital stories or comics in language classes.

These integrations not only foster computational thinking but also support cross-curricular learning and digital creativity in ways that are engaging and relevant to students.

Teachers across countries employ hands-on, play-based, and project-oriented pedagogies, often combining **unplugged activities, block-based programming**, and **entry level robotics**. Scaffolded instruction and collaborative methods are frequently used to adapt content to students' developmental stages and varying digital skills. However, the **systematic**

inclusion of the full range of I4All topics remains inconsistent, often depending on individual teacher expertise, school priorities, and access to resources.

Persistent challenges affect the depth and consistency of informatics instruction. Teachers are under constant pressure to **stay up to date with evolving technologies**, yet face **unequal access to digital tools**, **limited curriculum time**, and **lack of dedicated support structures**. In countries where informatics is not formally embedded in the curriculum, its implementation often relies on teachers' personal initiative and informal networks, which can lead to uneven learning opportunities for students.

To strengthen the role of informatics at the primary level, more structured integration of the **I4All core topics** is needed, supported by **targeted professional development**, **curriculum alignment**, and **greater institutional support** for interdisciplinary teaching.

2.8 Validation of Competencies

To ensure the quality and relevance of the proposed competencies for primary school informatics teachers, a validation survey was developed and distributed to project partners. The survey gathered feedback on the clarity, feasibility, and applicability of the listed competencies, grouped into three categories: subject-specific, pedagogical, and transversal. Respondents were asked to assess both the current and desired competence levels of primary school teachers in their country across all three categories. In addition, they could provide comments or suggest modifications for each group of competencies. The survey also included questions on the perceived relevance of the catalogue to classroom needs, the clarity of descriptions, and the feasibility of implementation. The responses collected provide valuable insights into the practical suitability of the catalogue and support its further use in the context of primary informatics education.

2.8.1 Results from the Survey

This section presents the results of the validation survey on competencies for informatics teachers in primary schools (ISCED 1). The findings are organised by survey question, offering an overview of the responses collected across partner countries. The aim is to highlight patterns, differences, and key observations that emerge from the data. In total, **16 responses** were submitted, ensuring representation from all partner countries. The number of responses varied, as in some cases more than one person from the same partner organisation completed the survey.

Role

This question asked project partners to indicate their professional role(s) related to primary informatics education. Since it was a multiple-choice question, participants could select more than one option if applicable.

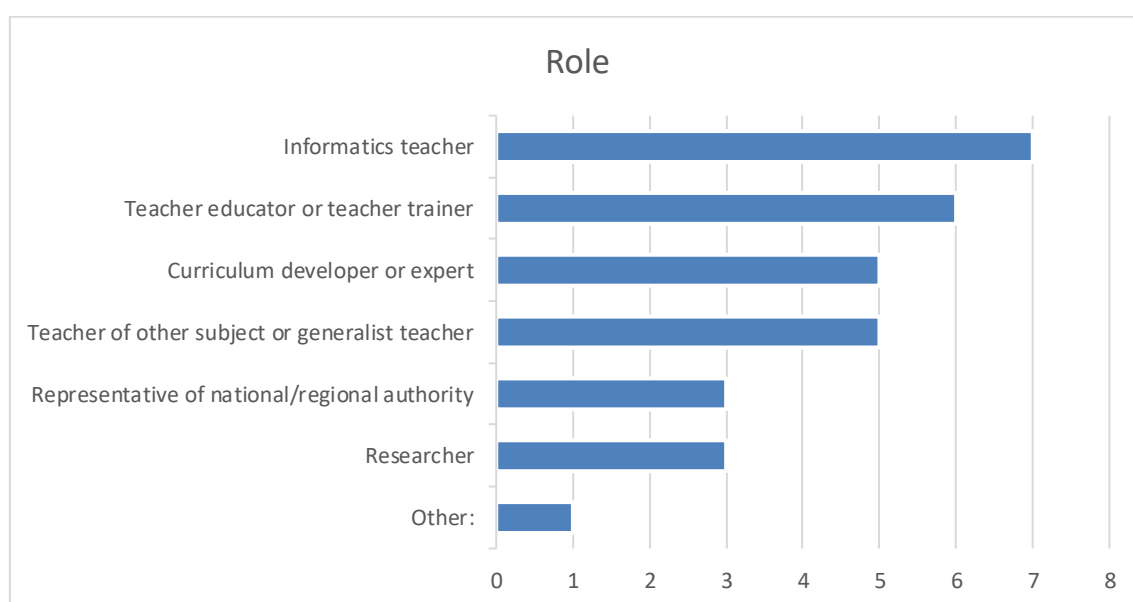


Figure 1 Roles of the participants



The survey responses were provided by project partners representing a range of roles connected to primary informatics education. The largest group identified themselves as **informatics teachers** (7 participants), followed by **teacher educators or trainers** (6), and **teachers of other subjects or generalist teachers** (5). In addition, 5 participants described their role as **curriculum developers or experts**, while smaller groups represented **researchers** (3) and **national or regional authorities** (3). One participant selected “other” and clarified that they are an **English teacher**. This distribution shows that the validation process captured perspectives from different professional backgrounds within the consortium. It combines input from those directly engaged in teaching with insights from partners who develop curricula, train teachers, or work in policy contexts. Such diversity of roles enriches the analysis, as it reflects not only classroom practice but also the broader framework in which informatics education is developed and supported.

Experience

This question asked project partners whether they have experience in teaching informatics or digital skills to primary school students. Each participant could only select one option (Yes/No).

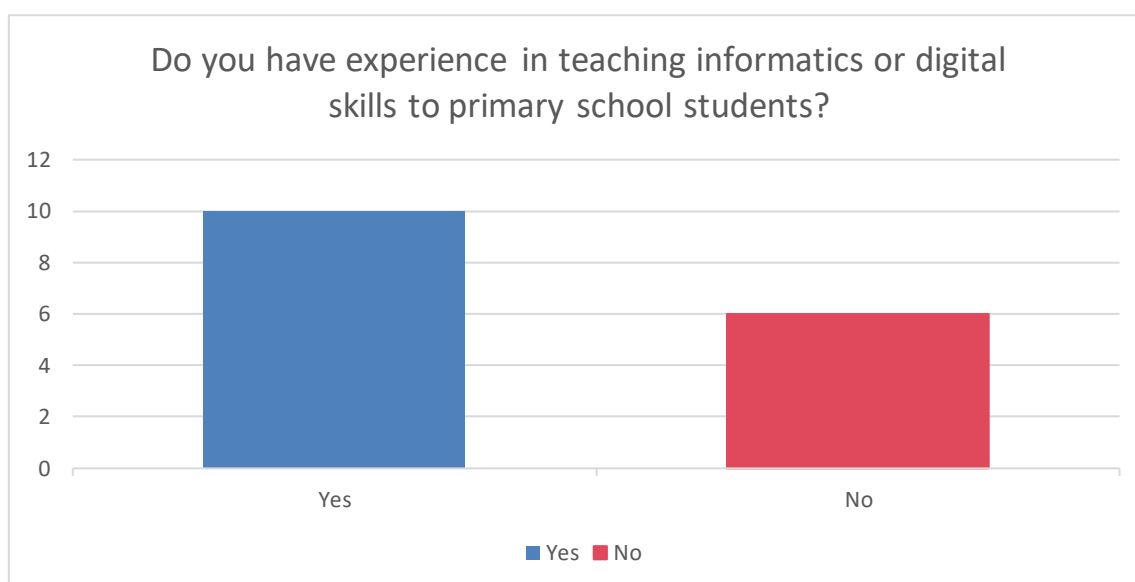


Figure 2 Experience in teaching informatics or digital skills to primary school students

The results show that a majority of respondents, **10 out of 16 (63%)**, reported having direct experience teaching informatics or digital skills in primary schools. The remaining **6 respondents (38%)** indicated that they do not have such experience. This distribution highlights that the survey combined insights from partners with hands-on classroom practice as well as those working in related roles such as curriculum development, teacher training, or policy. The balance ensures that the validation reflects both practical teaching perspectives and broader systemic viewpoints. The strong presence of respondents with classroom experience adds credibility to the assessment of current and desired competence levels.

Subject-Specific Competencies

This section asked project partners to rate the current and desired levels of primary school informatics teachers in their country across different subject-specific competence areas. The scale ranged from **None [1]**, **Basic [2]**, **Intermediate [3]**, to **Advanced [4]**, and mean values were calculated accordingly. Participants were invited to assess both the existing situation (“current level”) and the level they considered necessary (“desired level”).

Current level

The following table summarises the responses regarding the current level of subject-specific competencies.

Table 2 Current level of subject-specific competencies of primary school informatics teachers

Current level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Data and Information	2	9	2	3	2,4	0,96
Algorithms	2	10	3	1	2,2	0,75
Programming	4	6	5	1	2,2	0,91
Computing Systems	4	7	3	2	2,2	0,98
Digital Creativity and Design	1	6	6	3	2,7	0,87
Modelling and Simulation	7	6	2	1	1,8	0,91
Privacy, Safety and Security	2	6	5	3	2,6	0,96
Artificial Intelligence	7	5	3	1	1,9	0,96
Computational Thinking (CT)	3	7	4	2	2,3	0,95

The results for the current level suggest that informatics teachers in primary schools are generally assessed at a **basic level** across most subject-specific competence areas. Some competencies, such as *Digital Creativity and Design* (mean 2,7) and *Privacy, Safety and Security* (mean 2,6), are slightly stronger, reaching closer to the intermediate level. In contrast, competencies such as *Modelling and Simulation* (mean 1,8) and *Artificial Intelligence* (mean 1,9) appear particularly underdeveloped, with most teachers placed at the basic or even “none” level. Overall, the findings indicate that while teachers possess foundational skills, advanced or even intermediate mastery is less common in primary school contexts.

Desired level

The following table presents the responses regarding the desired level of subject-specific competencies.

Table 3 Desired level of subject-specific competencies of primary school informatics teachers

Desired level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Data and Information	0	1	7	8	3,4	0,63
Algorithms	0	3	8	5	3,1	0,72
Programming	0	2	10	4	3,1	0,62
Computing Systems	0	3	7	6	3,2	0,75
Digital Creativity and Design	0	0	7	9	3,6	0,51
Modelling and Simulation	0	5	8	3	2,9	0,72
Privacy, Safety and Security	0	3	2	11	3,5	0,82

Desired level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Artificial Intelligence	0	3	6	7	3,3	0,77
Computational Thinking (CT)	0	1	6	9	3,5	0,63

When asked about the desired levels, respondents consistently rated expectations much higher. Across all areas, the desired level was at least **intermediate**, with many competencies leaning towards **advanced**. The strongest expectations were expressed for *Digital Creativity and Design* (mean 3,6), *Privacy, Safety and Security* (mean 3,5), and *Computational Thinking* (mean 3,5). Even areas that are currently rated lower, such as *Artificial Intelligence* (mean 3,3) and *Modelling and Simulation* (mean 2,9), are expected to be developed at intermediate to advanced levels. This shows a clear consensus among project partners that informatics teachers for ISCED 1 should demonstrate significantly higher competence than is currently observed.

Comparison between current and desired levels

To illustrate the contrast between current and desired levels, the following double bar chart places the two sets of ratings side by side.

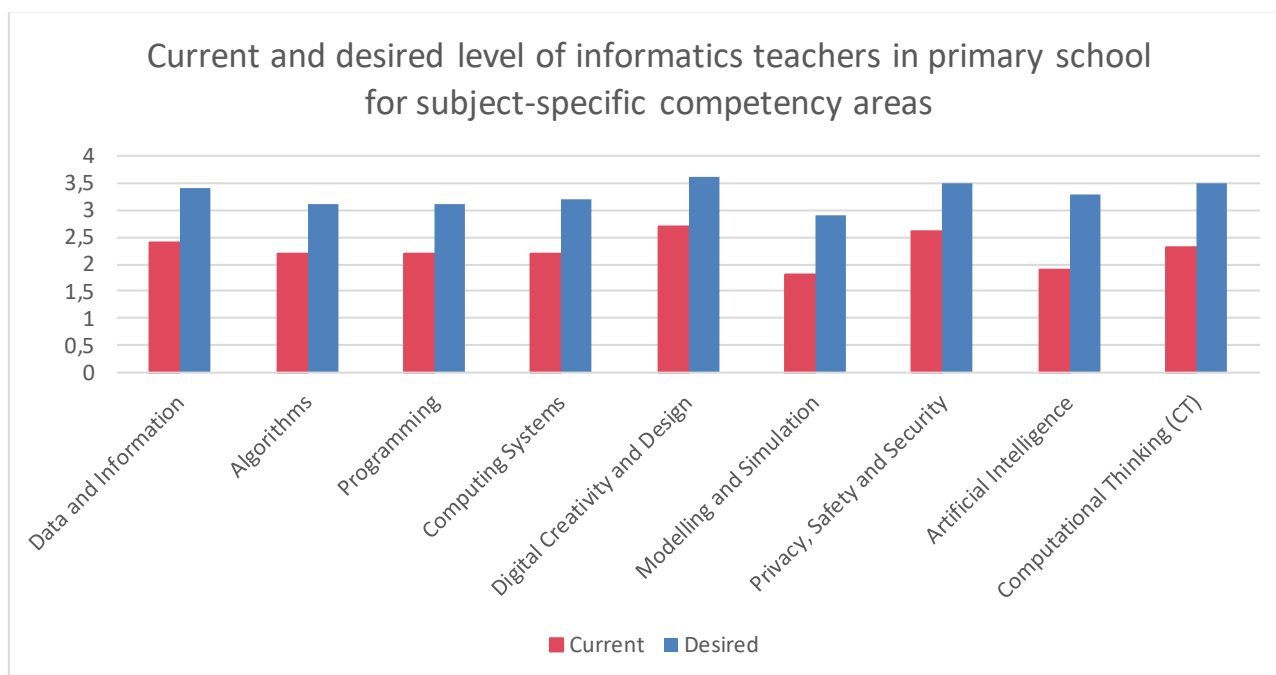


Figure 3 Comparison between current and desired levels of informatics teachers in primary school for subject-specific competency areas

The comparison highlights a consistent gap across all subject-specific competence areas. While current levels cluster around the basic range, desired levels are placed firmly in the intermediate to advanced range. The gap is most visible in areas such as **Artificial Intelligence**, **Computational Thinking**, and **Privacy, Safety and Security**, where teachers are expected to advance by one or more levels. Even in stronger areas, such as **Digital Creativity and Design**, expectations are considerably higher than present levels. Overall, the findings indicate a shared view that significant professional growth is needed for primary school informatics teachers to meet the expectations set for subject-specific competencies.

Comments on Subject-Specific Competencies

This optional question invited project partners to comment on whether any subject-specific competencies should be added, removed, or revised. In total, 7 responses were received.

The comments indicate that the proposed subject-specific competencies are generally seen as relevant and comprehensive, but respondents highlighted several areas where refinements could be beneficial. A recurring theme was the inclusion of broader societal and ethical dimensions of technology, such as **responsible technology use**, **digital wellbeing**, and **sustainability**. In addition, some partners suggested clarifying existing competencies, for instance, reframing **Digital Creativity and Design** as digital content creation, or adjusting the focus of **Artificial Intelligence** towards age-appropriate awareness and curiosity rather than technical implementation. Further proposals included adding new areas, such as **automation and robotics**, **problem decomposition and debugging**, and the use of **unplugged activities**, which are considered highly relevant in primary education. Overall, the comments emphasise the importance of ensuring that competencies remain age-appropriate, practically applicable, and connected to pupils' real-life experiences.

Pedagogical Competencies

This section asked project partners to rate the current and desired levels of primary school informatics teachers in their country across different pedagogical competence areas. The scale ranged from **None [1]**, **Basic [2]**, **Intermediate [3]**, to **Advanced [4]**, and mean values were calculated accordingly.

Current level

The following table summarises the responses regarding the current level of pedagogical competencies.

Table 4 Current level of pedagogical competencies of primary school informatics teachers

Current level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Teaching and Learning	0	4	10	2	2,9	0,62
Assessment and Evaluation	0	10	5	1	2,4	0,63
Differentiation and Inclusion	1	10	4	1	2,3	0,7

The results for the current level suggest that teachers are generally assessed at a **basic level** in most pedagogical competence areas. The strongest area is *Teaching and Learning* (mean 2,9), where teachers are closer to an intermediate level. In contrast, *Assessment and Evaluation* (mean 2,4) and *Differentiation and Inclusion* (mean 2,3) remain at a lower level, showing that practices related to adapting teaching to diverse learners and systematically evaluating learning outcomes are less developed. Overall, the findings indicate that while some teaching practices are in place, more advanced pedagogical strategies are not yet widespread in primary school informatics education.

Desired level

The following table presents the responses regarding the desired level of pedagogical competencies.

Table 5 Desired level of pedagogical competencies of primary school informatics teachers

Desired level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Teaching and Learning	0	1	3	12	3,7	0,6
Assessment and Evaluation	0	0	6	10	3,6	0,5

Desired level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Differentiation and Inclusion	0	0	6	10	3,6	0,5

When asked about the desired levels, respondents expressed very high expectations across all areas. All three pedagogical competencies were rated close to **advanced**, with means between 3,6 and 3,7. This indicates a strong consensus among project partners that informatics teachers should be highly skilled in creating effective learning environments, assessing student progress, and ensuring inclusive practices that address diverse learning needs. In particular, the high ratings for *Differentiation and Inclusion* (mean 3,6) suggest a clear expectation that teachers be able to adapt teaching methods and content to varied learners in the primary classroom.

Comparison between current and desired levels

The following double bar chart illustrates the contrast between current and desired levels.

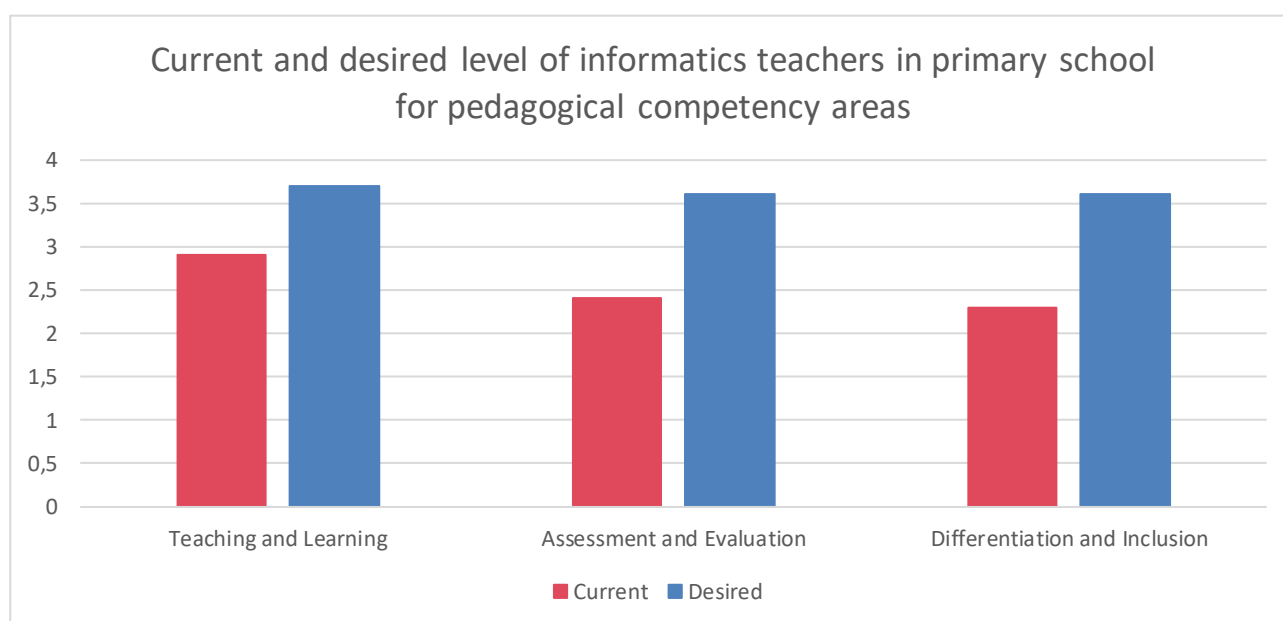


Figure 4 Comparison between current and desired levels of informatics teachers in primary school for pedagogical competency areas

The comparison highlights a clear gap between current and desired levels in all three pedagogical competencies. While teachers are currently assessed as being mostly at a basic level, expectations are set at the intermediate to advanced level across the board. The largest discrepancy is visible in **Differentiation and Inclusion**, where current practice lags significantly behind expectations. The findings suggest that substantial professional development is needed to strengthen pedagogical skills, especially in adapting teaching for diverse learners and in implementing effective assessment strategies.

Comments on Pedagogical Competencies

This optional question invited project partners to comment on whether any pedagogical competencies should be added, removed, or revised. In total, 7 responses were received.

The comments indicate that the three proposed pedagogical competencies (Teaching and Learning, Assessment and Evaluation, and Differentiation and Inclusion) are broadly considered **relevant and appropriate**, yet respondents suggested several refinements to strengthen them. A recurring theme was the need to emphasise more **active, project-based, and inquiry-based learning approaches**, highlighting that students should not only acquire theoretical knowledge but also engage in exploration, experimentation, and problem-solving. Another theme was the importance

of **inclusive practices**, with recommendations to strengthen competencies in accessibility, universal design for learning (UDL), and support for pupils with special educational needs. Respondents also suggested expanding the scope of **Assessment and Evaluation** to include authentic and portfolio-based assessment methods, which are well suited to primary-level informatics activities. Additional proposals included enhancing teachers' skills in **classroom management in technology-rich environments**, fostering **collaboration and peer learning**, and promoting strategies to **motivate and engage learners** (e.g. through gamification or creative challenges). Overall, the feedback reinforces the relevance of the existing framework while pointing to refinements that would ensure pedagogical competencies are more inclusive, practice-oriented, and age-appropriate.

Transversal Competencies

This section asked project partners to rate the current and desired levels of primary school informatics teachers in their country across different transversal competence areas. The scale ranged from **None [1]**, **Basic [2]**, **Intermediate [3]**, to **Advanced [4]**, and mean values were calculated accordingly.

Current level

The following table summarises the responses regarding the current level of transversal competencies.

Table 6 Current level of transversal competencies of primary school informatics teachers

Current level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Professional Growth	0	7	8	1	2,6	0,62
Collaboration and Cross-Disciplinary Integration	0	11	2	3	2,5	0,82
Ethical and Inclusive Practice	0	7	8	1	2,6	0,62
Innovation and Adaptability	0	9	6	1	2,5	0,63
Effective and Responsible Communication	0	7	6	3	2,8	0,77

The results suggest that teachers are generally assessed at a **basic to intermediate level** across the transversal competencies. *Effective and Responsible Communication* (mean 2,8) is the strongest area, while *Professional Growth* (mean 2,6) and *Ethical and Inclusive Practice* (mean 2,6) are also relatively better developed. In contrast, *Collaboration and Cross-Disciplinary Integration* (mean 2,5) and *Innovation and Adaptability* (mean 2,5) remain lower, indicating that teachers currently have limited opportunities or experience in these areas. Overall, the findings show that while some transversal competencies are present, teachers are still largely at the basic stage rather than fully proficient.

Desired level

The following table presents the responses regarding the desired level of transversal competencies.

Table 7 Desired level of transversal competencies of primary school informatics teachers

Desired level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Professional Growth	0	1	2	13	3,8	0,58

Desired level	None [1]	Basic [2]	Intermediate [3]	Advanced [4]	Mean	Std. Deviation
Collaboration and Cross-Disciplinary Integration	0	1	7	8	3,4	0,63
Ethical and Inclusive Practice	0	0	7	9	3,6	0,51
Innovation and Adaptability	0	0	6	10	3,6	0,5
Effective and Responsible Communication	0	1	5	10	3,6	0,63

When asked about the desired levels, respondents consistently placed expectations at the **advanced level** across all areas. *Professional Growth* (mean 3,8) received the highest rating, highlighting the importance of continuous learning and development. High expectations were also expressed for *Ethical and Inclusive Practice* (mean 3,6), *Innovation and Adaptability* (mean 3,6), and *Effective and Responsible Communication* (mean 3,6). Even *Collaboration and Cross-Disciplinary Integration* (mean 3,4), though rated slightly lower, was still well above the current level. These results show a strong consensus that informatics teachers in primary schools should not only master subject-specific knowledge but also actively contribute to broader professional and cross-disciplinary practices.

Comparison between current and desired levels

The following double bar chart illustrates the contrast between current and desired levels.

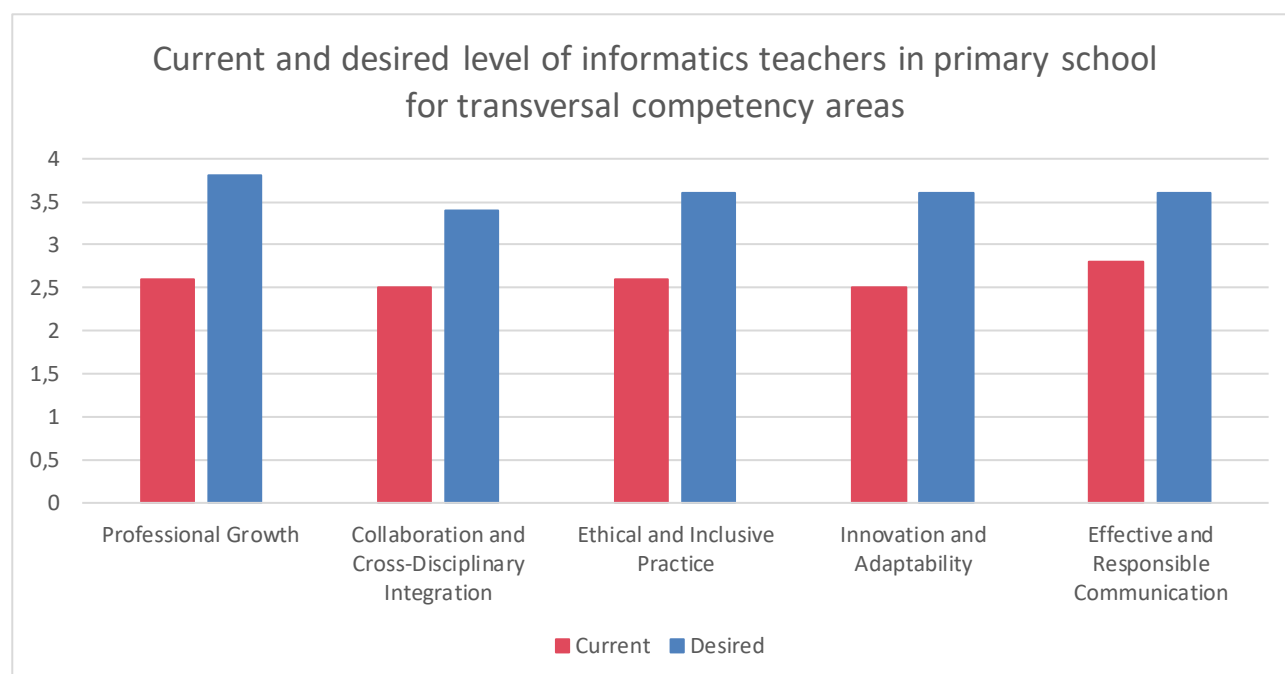


Figure 5 Comparison between current and desired levels of informatics teachers in primary school for transversal competency areas

The comparison highlights a consistent gap between current and desired levels across all five transversal competencies. While current ratings cluster around basic to intermediate levels, desired levels are firmly set at the advanced range. The largest gap appears in **Professional Growth**, where teachers are expected to progress from a basic/intermediate level to advanced. Similarly, significant differences are visible in **Ethical and Inclusive Practice** and **Innovation and Adaptability**, reflecting the view that teachers need to adopt more advanced practices in these areas. Overall, the

findings underline the need for sustained professional development and support to help teachers strengthen transversal competencies that are essential for modern, inclusive, and adaptive teaching in primary schools.

Comments on Transversal Competencies

This optional question invited project partners to comment on whether any transversal competencies should be added, removed, or revised. In total, 7 responses were received.

The comments confirm that the proposed transversal competencies are considered highly relevant, but several refinements were suggested. A recurring theme was the need to strengthen aspects of **leadership and digital transformation**, including mentoring colleagues and supporting school-wide change. Respondents also emphasised the importance of **data-informed decision-making**, **change management**, and **critical reflection on technology in society**. Another cluster of suggestions focused on broadening competencies to include **digital wellbeing**, **school-home communication**, and **project-based learning** as a means of cross-disciplinary integration. Several comments underlined the significance of **peer learning and professional collaboration**, highlighting that continuous development and teamwork are central to effective transversal practice. Overall, while no areas were seen as unnecessary, the feedback points to opportunities for refinement, ensuring that transversal competencies are both realistic for primary teachers and responsive to emerging challenges in informatics education.

Relevance of Competencies

This question asked project partners whether they find the proposed competencies relevant to real classroom needs. Each participant could only select one option.

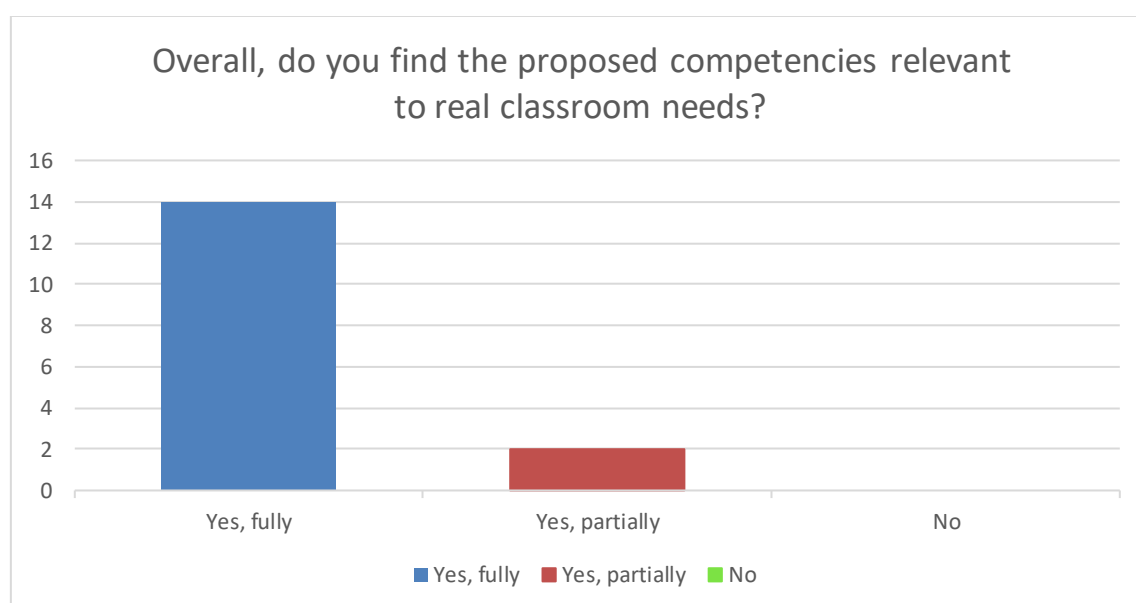


Figure 6 Relevance of competencies

All respondents agreed that the proposed competencies are relevant to classroom practice. The majority, **14 participants (88%)**, indicated that the competencies are “fully relevant,” while **2 participants (13%)** found them “partially relevant.” None considered them irrelevant. This overwhelming consensus demonstrates strong support for the framework and confirms that the proposed competencies address real needs in primary informatics education.

Clarity

This question asked project partners how clearly the competencies are described. Each participant could only select one option.

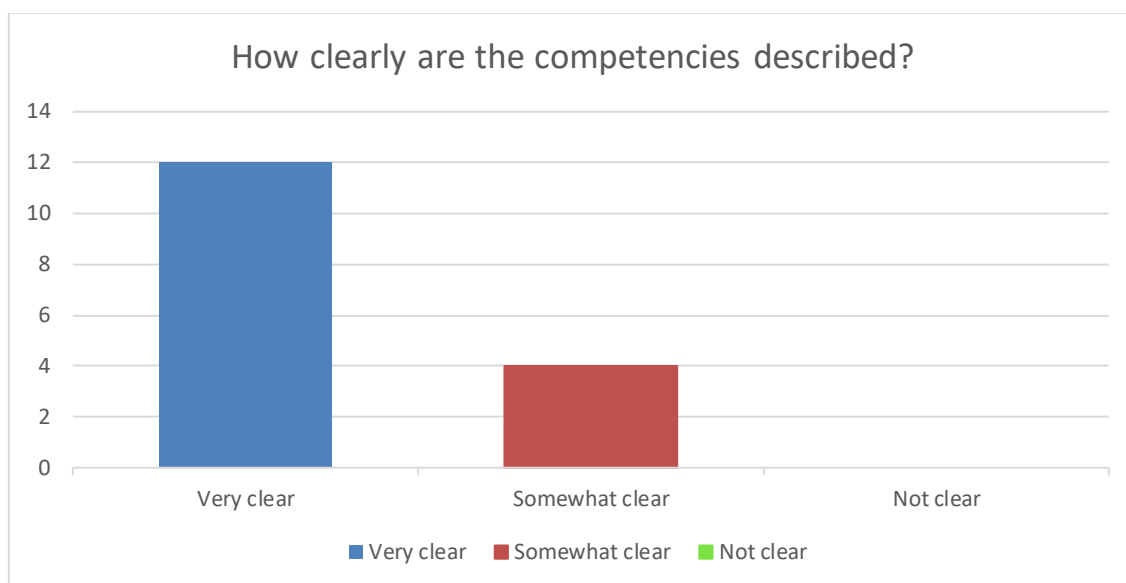


Figure 7 Clarity of competencies

The results show that the competencies were generally well understood. **12 participants (75%)** rated them as “very clear,” while **4 participants (25%)** considered them “somewhat clear.” None selected the option “not clear.” These results indicate that the descriptions are overall accessible and understandable, though a small group of respondents noted that further clarification could improve readability.

Feasibility of Competencies

This question asked project partners whether the competencies are feasible for teachers to develop and apply in the primary school context. Each participant could only select one option.

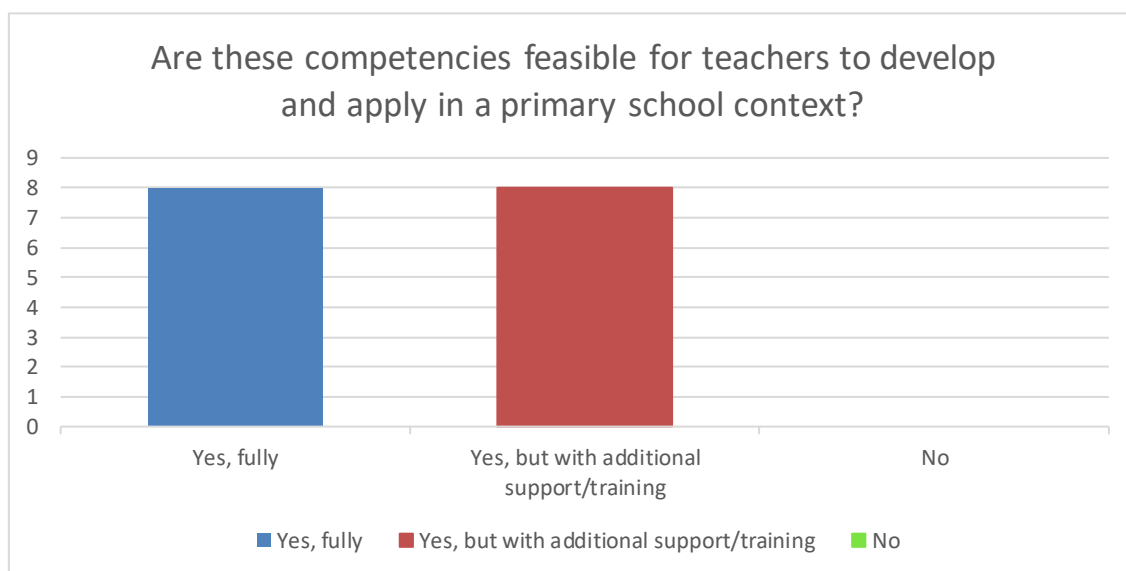


Figure 8 Feasibility of competencies

All participants agreed that the competencies are feasible, though opinions differed on the extent of support needed. **8 participants (50%)** stated that the competencies are “fully feasible,” while the other **8 participants (50%)** indicated that they are feasible only with additional support or training. These results suggest that the competencies are realistic in principle but highlight the importance of providing professional development opportunities and resources to ensure successful implementation in primary schools.

Other Comments

This optional question invited project partners to share any additional comments regarding the proposed competencies or the survey as a whole. In total, 6 responses were received.

The additional comments were generally positive, with respondents affirming the clarity and relevance of the competencies. Several emphasised that the framework already provides a strong foundation for supporting informatics education at the primary level. At the same time, partners highlighted the importance of offering **continuous professional support and training**, particularly in areas such as *Artificial Intelligence (AI)*, to help teachers translate competencies into effective classroom practice. Some respondents suggested providing more **examples adapted to the primary school level**, especially for complex areas such as AI and transversal competencies, in order to enhance clarity and usability. Others recommended developing **tools for teachers to self-assess their digital competencies**, which could help identify strengths, gaps, and professional development needs. Overall, the comments indicate satisfaction with the survey design and competence framework, while also pointing to the need for practical resources, workshops, and training pathways to ensure successful implementation.

2.8.2 Survey summary

This section summarises the main findings of the validation survey on competencies for informatics teachers in primary schools (ISCED 1). In total, 16 responses were collected across partner countries, ensuring representation of different roles and perspectives within the consortium.

The survey results demonstrate strong endorsement of the proposed competencies. Almost all participants fully agreed that the competencies are relevant to classroom practice, with a small number acknowledging partial relevance. Respondents also found the descriptions clear and understandable, although several suggested providing age-appropriate examples for complex areas such as *Artificial Intelligence* and transversal competencies. Importantly, all agreed that the competencies are feasible to develop, though half emphasised that teachers would require additional training and resources to apply them effectively.

A consistent gap emerged between current and desired competence levels. Across all three domains (subject-specific, pedagogical, and transversal) current levels were rated at a basic or lower-intermediate stage, while desired levels were placed firmly in the intermediate to advanced range. This discrepancy was particularly pronounced in subject-specific competencies such as *Artificial Intelligence*, *Modelling and Simulation*, and *Computational Thinking*. In pedagogy, *Teaching and Learning* showed some strength, but significant gaps remained in *Assessment and Evaluation* and especially *Differentiation and Inclusion*. The transversal competencies revealed similar differences, with *Professional Growth*, *Innovation and Adaptability*, and *Ethical and Inclusive Practice* identified as areas requiring substantial improvement.

The qualitative feedback reinforced the framework's value but pointed to areas for refinement. Respondents highlighted the importance of broadening competencies to address ethical and societal dimensions of informatics, including responsible technology use, sustainability, and digital wellbeing. They also suggested adding topics such as robotics, unplugged activities, and age-appropriate approaches to *Artificial Intelligence*. Pedagogical feedback stressed the relevance of active, project-based, and inquiry-based learning, authentic assessment methods, and inclusive teaching practices following universal design for learning principles. Suggestions for transversal competencies included emphasising digital leadership, peer collaboration, project-based integration, and school-home communication.

Taken together, the survey results validate the competence framework while underscoring the need for teacher support. The findings confirm that the framework provides a solid basis for primary informatics education but also reveal that current teacher competencies fall well short of the levels considered necessary. This highlights the importance of continuous professional development, practical training opportunities, and access to resources that enable teachers to bridge the gap between current and desired levels. The survey therefore both validates the proposed framework and points to the next steps required to strengthen teacher preparation in primary informatics education.

3 Competencies for Informatics Teachers in Primary School

3.1 Introduction

Recent advances in computing education emphasize the necessity to evolve from purely knowledge-based to integrated competency-based approaches, especially in informatics teacher education, where modern frameworks must combine disciplinary knowledge, pedagogical skills, and professional dispositions to ensure effective computer science instruction (Clear et al., 2020; Dagiene et al., 2021). This catalogue of competencies for informatics teachers is based on the Combined Knowledge and Competency (CKC) Model, which synergistically integrates knowledge-based structures with competency-oriented outcomes in curriculum design. The essence of the CKC model is that it emphasizes both ends of the learning continuum – facilitating teaching through organized knowledge areas while supporting evaluation through well-defined competency outcomes (Kumar et al., 2023). This integration ensures that teachers not only possess structured disciplinary knowledge (Caspersen et al., 2023) but also develop practical skills relevant to teaching informatics in a primary school setting. This perspective aligns with curriculum design principles that emphasize the integration of informatics concepts, pedagogy, and cognitive development across educational stages, especially in primary school contexts (Dagiene et al., 2021; Kim et al., 2021).

The *Informatics for All* initiative (I4All) was launched in 2018 as a joint effort by ACM Europe, Informatics Europe, and CEPIS, with the aim of establishing informatics as a foundational subject in general education across Europe. The catalogue is informed by the *Informatics Reference Framework* (Caspersen et al., 2022), which presents informatics as a fundamental discipline that should be part of education from primary through upper secondary levels. As noted in the framework, "Education systems have a responsibility to recognize this and to ensure that young people are equipped to be able to drive forward, judge innovation and take part in the development of a just and fair society."

Additionally, this catalogue incorporates insights from the CSTA Standards for Computer Science Teachers, which establish benchmarks for effective CS instruction in support of rigorous computer science education for all K-12 students (CSTA, 2020). The framework also aligns with the European Framework for the Digital Competence of Educators (DigCompEdu) (Redecker & Punie, 2017), which outlines the digital competencies that educators need to effectively teach in our increasingly digital environment.

The competencies outlined here (Figure 9) are organized into three key dimensions (Brinda et al., 2025; CSTA, 2020; Kim et al., 2021):

- **Subject-specific competencies:** What teachers need to know about informatics content.
- **Pedagogical competencies:** How to effectively teach informatics to young learners.
- **Transversal competencies:** Professional skills and dispositions that support effective teaching practice.

The competency structure follows a logical framework that includes:

- **Task:** The specific action required
- **Competency statement:** What the teacher can do
- **Knowledge:** Required theoretical understanding
- **Skills:** Practical abilities needed
- **Dispositions:** Professional attitudes and values
- **DigCompEdu Area:** Connection to established standards

This structure aligns with the CKC model by combining knowledge elements with competency application (Kumar et al., 2023).

Within the *Digital First project*, we are developing targeted competency catalogues for each educational level – primary (ISCED 1), and secondary (lower secondary, ISCED 24, and upper secondary, ISCED 34) – ensuring age-appropriate progression and context-sensitive implementation of informatics teacher education. This decision is supported by findings from WP2 transnational research and recommendations (D2.2, D2.3), which highlight substantial variation in teacher profiles, curricular coverage, and pedagogical approaches across levels and countries (Digital First Consortium,



2024). As also evidenced in the "Teacher education and teacher support associations for informatics and computing in schools" IFIP report (2024) there is substantial diversity in how different European countries approach informatics teacher education, with varying institutional structures, professional associations, and support mechanisms reflecting each nation's unique educational context and priorities. The catalogues of competencies are informed by co-creation workshops and structured interviews with teachers, aligned with the Combined Knowledge and Competency (CKC) model and guided by the Informatics for All framework and DigCompEdu (Brinda et al., 2025; Kumar et al., 2023; Redecker & Punie, 2017).

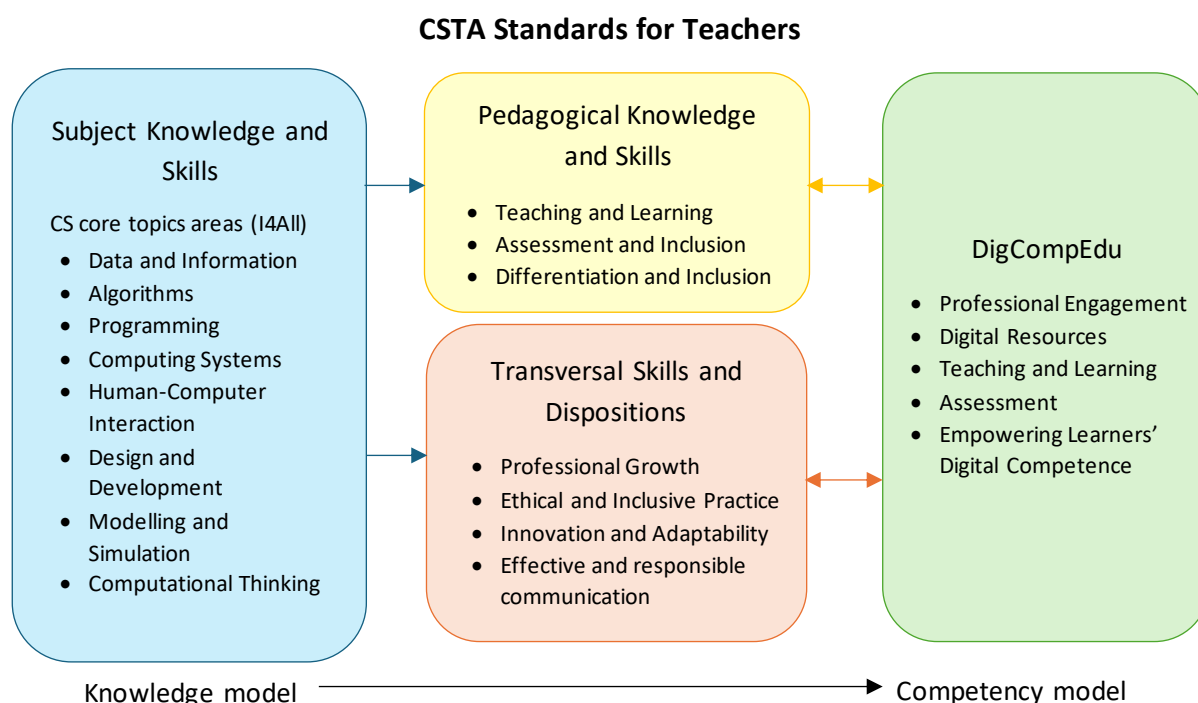


Figure 9 Integrating knowledge and competency models into a catalogue of competencies for Informatics teachers

Each catalogue will distinguish between essential and desirable competencies in the three dimensions: subject-specific knowledge, pedagogical strategies, and transversal professional skills. This approach allows us to address the unique needs of different learner age groups and curricula, support consistent teacher training across Europe, and provide a foundation for formal (tertiary) and informal (continuing) professional development activities, as planned in WP4.

For easier understanding of subject-specific competencies, we have collected a variety of practical examples of activities that teachers can implement in their teaching practice. These activities are linked to the functional approach we are developing within the scope of this project and are presented in Annex 2 – Practical Examples for Subject-Specific Competency Areas.

3.2 Subject-Specific Competencies

Data and Information

- **Task:** Design and implement learning activities that help students collect, sort, organise and present data.
- **Competency statement:** The teacher can design age-appropriate learning experiences that engage students in collecting, sorting, and organizing data, while helping them understand how data becomes meaningful information through interpretation.
- **Knowledge:** Types of data (numerical, textual, categorical), data collection methods (observation, simple surveys), and ways of representing data (simple charts, pictographs, bar graphs).
- **Skills:** Guide data gathering, categorising, charting, and interpreting patterns.

- **Dispositions:** Fosters curiosity, awareness of data ethics and privacy, promotes inclusive discussions.
- **DigCompEdu Area:** 2.1 Selecting digital resources, 3.1 Teaching, 6.1 Information and media literacy, 6.4 Responsible use.

Algorithms

- **Task:** Guide students to recognise, create and evaluate simple algorithms in everyday contexts.
- **Competency statement:** The teacher can support the development of algorithmic thinking by helping students identify everyday sequences, create step-by-step instructions, and reflect on their clarity and efficiency.
- **Knowledge:** Concepts of sequence, repetition (loops), conditions (if statements), modelling of real-life processes.
- **Skills:** Facilitate storytelling, create algorithm-based games, and support students in comparing different solution paths.
- **Dispositions:** Encourages logical thinking, reflection, collaboration, persistence.
- **DigCompEdu Area:** 3.1 Teaching, 3.3 Collaborative learning, 6.3 Digital content creation.

Programming

- **Task:** Introduce students to basic programming using age-appropriate tools and environments.
- **Competency Statement:** The teacher can guide students to explore basic programming constructs such as sequences, loops, and conditionals using block-based programming environments or unplugged activities.
- **Knowledge:** Basic programming constructs, event-driven logic, inputs and outputs.
- **Skills:** Use block-based programming environments (ScratchJr, Scratch, Code.org) or unplugged kits, facilitate debugging activities.
- **Dispositions:** Promotes curiosity, exploration, perseverance, confidence through exploration and trial-and-error.
- **DigCompEdu Area:** 2.2 Creating and modifying digital resources, 3.1 Teaching, 6.3 Digital content creation.

Computing Systems

- **Task:** Facilitate students' understanding of hardware and software components in a computer system.
- **Competency Statement:** The teacher can help students identify key parts of computing systems (e.g. keyboard, mouse, screen, processor, speaker, microphone) and explain their functions using analogies and hands-on exploration.
- **Knowledge:** Basic components of computer systems, input/output devices, the role of software.
- **Skills:** Use real or model devices, design simple input/output activities to explore device functions.
- **Dispositions:** Supports digital curiosity, connect technology to daily life, and values accessibility.
- **DigCompEdu Area:** 2.1 Selecting digital resources, 2.2 Creating and modifying digital resources.

Digital Creativity and Design

- **Task:** Support students in designing and creating digital content.
- **Competency Statement:** The teacher can guide students through creative digital projects, from brainstorming to testing and sharing simple digital artifacts (e.g. stories, posters, animations).
- **Knowledge:** Basics of digital storytelling, images, audio, and video in creative expression.
- **Skills:** Use simple creative tools (Book Creator, Paint, Canva, Scratch), support iterative creation and improvement.
- **Dispositions:** Encourages self-expression, a feedback culture, and digital creativity.
- **DigCompEdu Area:** 2.2 Creating and modifying digital resources, 3.1 Teaching.

Modelling and Simulation

- **Task:** Enable students to explore and use simple simulations of real-world phenomena.
- **Competency Statement:** The teacher can engage students with age-appropriate simulations (digital or physical) to explore cause-and-effect relationships and model simple processes.
- **Knowledge:** Basics of simulation, simple cause-and-effect systems (e.g. traffic lights, water cycle).
- **Skills:** Use tools such as LEGO, Beebots, or animations, design activities to simulate natural or social events.

- **Dispositions:** Fosters inquiry, creativity, and appreciation for uncertainty and variability.
- **DigCompEdu Area:** 2.2 Creating and modifying digital resources, 3.1 Teaching, 5.3 Actively engaging learners, 6.3 Digital content creation.

Privacy, Safety and Security

- **Task:** Promote students' awareness of online safety and responsible digital behaviour.
- **Competency Statement:** The teacher can raise awareness of safe online practices and personal data protection through meaningful discussions and guided digital activities.
- **Knowledge:** Personal data, passwords, online interactions, respectful digital behaviour.
- **Skills:** Explain what personal data is, use scenarios, involve students in creating “digital rules”.
- **Dispositions:** Models responsibility, protects trust, builds empathy and digital ethics.
- **DigCompEdu Area:** 2.3 Managing, protecting and sharing digital resources, 6.1 Information and media literacy, 6.4 Responsible use.

Artificial Intelligence

- **Task:** Introduce students to basic concepts of artificial intelligence and machine learning, showing how machines can learn from data and assist humans.
- **Competency Statement:** The teacher can design age-appropriate activities that help students understand that machines can learn from data, make decisions, and collaborate with people, while recognising that AI is created and guided by humans.
- **Knowledge:** Natural vs. artificial creations, basics of artificial intelligence, simple types of machine learning (supervised, unsupervised, reinforcement), basic ideas/concepts of neural networks.
- **Skills:** Explain AI concepts through stories, analogies, and playful activities, highlight AI examples in daily life, model basic learning from data using tangible examples.
- **Dispositions:** Encourages curiosity about technology, critical thinking about human-machine interaction, and responsible attitudes towards innovation.
- **DigCompEdu Area:** 3.1 Teaching, 5.3 Actively engaging learners, 6.1 Information and media literacy.

Computational Thinking

- **Task:** Support students in applying computational thinking across learning contexts.
- **Competency Statement:** The teacher embeds computational thinking (CT) practices such as pattern recognition, decomposition, and step-by-step reasoning into diverse classroom activities.
- **Knowledge:** Core CT concepts, progression models for CT in primary education, and interdisciplinary applications.
- **Skills:** Identify CT opportunities in storytelling, play, and problem-solving, guide students in systematic exploration and reasoning.
- **Dispositions:** Recognises CT across disciplines, promotes logical reasoning, and values curiosity and step-by-step exploration.
- **DigCompEdu Area:** 3.1 Teaching, 6.5 Digital problem solving.

3.3 Pedagogical Competencies

Teaching and Learning

- **Task:** Plan and implement developmentally appropriate and engaging informatics activities using varied teaching strategies.
- **Competency Statement:** The teacher can design and facilitate informatics learning experiences that incorporate storytelling, play, problem-solving, and hands-on activities, using both digital tools and unplugged methods.
- **Knowledge:** Child development stages, inquiry-based learning, constructivist teaching approaches, storytelling and game-based strategies, and integration of computational thinking.
- **Skills:** Design age-appropriate activities, break down abstract concepts into simple tasks, use unplugged methods (e.g. games, kinaesthetic activities), integrate cross-curricular connections.



- **Dispositions:** Values and promotes student creativity, flexibility, enthusiasm for experimentation, and active learner engagement, adapts to diverse learning styles.
- **DigCompEdu Area:** 2.2 Creating and modifying digital resources, 3.1 Teaching, 3.2 Guidance, 3.3 Collaborative learning, 3.4 Self-regulated learning, 5.1 Accessibility and inclusion.

Assessment and Evaluation

- **Task:** Monitor students' understanding and progress in informatics through formative and authentic assessment practices.
- **Competency Statement:** The teacher can apply a variety of formative assessment strategies to capture students' development in computational thinking and their understanding of basic informatics concepts, while providing meaningful feedback and adapting instruction using tools such as observation, reflective dialogue, and simple digital tasks.
- **Knowledge:** Principles of formative assessment, basic assessment tools, feedback strategies, qualitative observation techniques.
- **Skills:** Observe students' problem-solving, ask guiding questions, design simple self-assessment tools, provide constructive feedback, and adapt instruction based on students' responses.
- **Dispositions:** Encourages growth, openness to student input, recognizes diverse expressions of understanding, and values the learning process over outcomes.
- **DigCompEdu Area:** 4.1 Assessment strategies, 4.2 Analysing evidence, 4.3 Feedback and planning.

Differentiation and Inclusion

- **Task:** Adapt informatics instruction to meet diverse learning needs and foster an equitable, inclusive learning environment.
- **Competency Statement:** The teacher can differentiate instruction and design supportive learning experiences that respond to students' individual abilities, learning styles, and backgrounds.
- **Knowledge:** Principles of Universal Design for Learning (UDL), gender inclusivity in STEM, differentiation strategies, and scaffolding methods/strategies.
- **Skills:** Modify task complexity, provide multilevel challenges, tailor resources and pacing, offer varied forms of representation and expression, manage group work dynamics, and foster an inclusive classroom climate.
- **Dispositions:** Demonstrates empathy, fairness, and commitment to equitable learning opportunities for all students.
- **DigCompEdu Area:** 3.3 Collaborative learning, 5.1 Accessibility and inclusion, 5.2 Differentiation and personalization, 6.4 Responsible use.

3.4 Transversal Competencies

Professional Growth

- **Task:** Engage in continuous professional learning to stay current with developments in informatics education.
- **Competency Statement:** The teacher actively engages in professional development opportunities and reflects on their teaching practices and applies new insights to improve their competence and confidence in teaching informatics.
- **Knowledge:** Trends in computing education, sources of continuing professional development (e.g. MOOCs, teacher networks), and reflective teaching frameworks.
- **Skills:** Identify personal learning goals, implement new practices in the classroom, and contribute to curriculum development within the school community.
- **Dispositions:** Demonstrates curiosity, adaptability, and commitment to lifelong learning.
- **DigCompEdu Area:** 1.3 Reflective practice, 1.4 Digital Continuing Professional Development (CPD).

Collaboration and Cross-Disciplinary Integration

- **Task:** Collaborate with colleagues and integrate informatics concepts across curriculum areas.
- **Competency Statement:** The teacher collaborates with peers to co-design interdisciplinary learning activities and contribute to school-wide informatics education initiatives.



- **Knowledge:** Models of curriculum integration, strategies for collaborative planning, and principles of cross-disciplinary learning.
- **Skills:** Co-plan learning projects, share teaching practices, and build interdisciplinary links with subjects such as mathematics, science, and the arts.
- **Dispositions:** Values teamwork, shared responsibility, and creative curriculum development.
- **DigCompEdu Area:** 1.2 Professional collaboration, 3.1 Teaching, 5.3 Actively engaging learners.

Ethical and Inclusive Practice

- **Task:** Promote inclusive, responsible, and ethical digital engagement in informatics teaching.
- **Competency Statement:** The teacher fosters equitable and supporting learning environments, models responsible digital behaviour, and ensures that students understand privacy, respect, and safe online practices.
- **Knowledge:** Digital ethics, inclusive education principles, accessibility guidelines, and fundamentals of privacy and online safety.
- **Skills:** Model ethical use of digital tools, explain data privacy in age-appropriate ways, adapt content to support diverse learners.
- **Dispositions:** Demonstrates empathy, fairness, and integrity, values digital wellbeing and inclusion.
- **DigCompEdu Area:** 5.1 Accessibility and inclusion, 6.1 Information and media literacy, 6.4 Responsible use.

Innovation and Adaptability

- **Task:** Integrate innovative tools and respond flexibly to technological and curricular change.
- **Competency Statement:** The teacher embraces emerging technologies and evolving educational approaches to enhance informatics learning and foster student engagement.
- **Knowledge:** Frameworks for evaluating new tools, principles of change management in education, and basics of design thinking.
- **Skills:** Pilot new resources, adapt existing teaching materials to technological changes, and respond flexibly to innovation and curriculum updates.
- **Dispositions:** Embraces experimentation, maintains critical awareness, and balances innovation with pedagogical purpose.
- **DigCompEdu Area:** 1.4 Digital Continuing Professional Development (CPD), 2.2 Creating and modifying digital resources.

Effective and responsible communication

- **Task:** Establish and maintain effective and respectful communication with students, parents, and colleagues using appropriate tools and practices, while ensuring privacy, accessibility, and professionalism.
- **Competency Statement:** The teacher selects and uses appropriate communication strategies to interact clearly, respectfully, and responsibly with students, parents, and colleagues, ensuring timely information sharing and upholding professional and ethical standards in both digital and face-to-face contexts.
- **Knowledge:** Professional communication protocols, school policies on digital and face-to-face communication, tools for school-home collaboration, data protection and privacy regulations, and inclusive communication principles.
- **Skills:** Use communication platforms effectively, model respectful dialogue, and adapt communication to different audiences and purposes.
- **Dispositions:** Values clarity, respect, professionalism, and inclusivity in all communication, promotes safe and ethical online interactions.
- **DigCompEdu Area:** 1.1 Organisational communication, 6.2 Digital communication and collaboration, 6.4 Responsible use.

3.5 Glossary of key terms

A

Advocate for Equity and Inclusion - Ensures that all students have fair access to computer science education, regardless of their background, abilities, or identity.

Algorithms - A sequence of steps or instructions to solve a problem. In computer science, algorithms form the basis for programming and computational thinking.

Artificial Intelligence - A field of computer science focused on creating systems that can perform tasks typically requiring human intelligence, such as learning from data, recognizing patterns, solving problems, and adapting to new situations.

B

Block-based programming - A programming approach using visual blocks to teach coding concepts to young learners. Examples: ScratchJr, Scratch

C

CKC Model - Combined Knowledge and Competency model; an approach to designing computer science curricula that combines the knowledge model (what a teacher knows) and the competency model (what a teacher can do).

Competency Statement - A description of what the teacher should be able to do, often starting with an action verb.

Computational Thinking - A problem-solving approach that involves decomposition, pattern recognition, abstraction, and algorithmic thinking.

Computing Systems - An integrated combination of components designed to process, store, and exchange data, including hardware and software.

CS - Abbreviation for Computer Science.

CSTA - Computer Science Teachers Association; an international association of computer science teachers that develops standards and guidelines for teaching computer science.

D

Data and Information - The understanding of how data are collected, organized, analysed, and used to model, represent, and visualize information about real-world artifacts and scenarios.

Debugging - The process of identifying, explaining, and fixing errors in a program to ensure that it behaves as expected.

Design and Development - The planning and creation of computing artifacts considering stakeholders' viewpoints and critically evaluating alternatives and their outcomes.

Design Learning Experiences Using Pedagogical Content Knowledge - Creating effective learning activities based on an understanding of how to teach CS content specifically.

Differentiated Instruction - Adapting teaching to meet diverse student needs by varying content, methods, or assessments.

DigCompEdu - The European Framework for the Digital Competence of Educators, which defines the digital competencies that a modern teacher should possess.

Digital Competencies - The set of knowledge, skills, and attitudes needed for the confident, creative, and critical use of digital technologies.

Digital Creativity - The exploration and use of digital tools to develop and maintain computing artifacts, also using a range of media.



Digital Storytelling - The practice of using digital tools to tell stories, often integrating multimedia elements like images, sound, and animation.

Dispositions - The attitudes, values, and tendencies that influence how a teacher approaches their work.

E

Evidence-based Pedagogy - The use of pedagogical approaches that are supported by research and evidence of effectiveness.

Evidence-Based Practices - Teaching methods and strategies that have been proven effective through systematic research and data analysis.

F

Formative Assessment - Ongoing assessment that provides feedback to improve learning rather than measure final achievement.

Framework for Computer Science - A structured overview of key concepts and practices in computer science that students should master at different levels of education.

H

Human-Computer Interaction - The evaluation, specification, development, and understanding of interaction between people and computing artifacts.

I

I4All - Informatics for All; a joint initiative launched in 2018 by ACM Europe, Informatics Europe, and CEPIS to establish informatics as a foundational subject in general education across Europe.

Inclusive Learning Experiences - Teaching approaches that ensure all students, regardless of their background, abilities, or identity, are fully included in the learning process and have equal opportunities for success.

Informatics - The scientific discipline that studies data, algorithms, computer systems and their applications. In some countries, it is called computing or computer science.

Informatics Reference Framework - A framework that presents informatics as a fundamental discipline that should be part of education from primary through upper secondary levels.

Informatics Teacher - An educator specialized in teaching informatics/computer science concepts, computational thinking, and digital skills to students. They help students develop understanding of how digital technologies work and how to create with them.

Integrated and Separate Approach - Combining informatics both as a cross-curricular theme and as a stand-alone subject.

ISCED - International Standard Classification of Education; a framework used to categorize and report cross-nationally comparable education statistics:

- ISCED 1: Primary education (typically ages 6-11)
- ISCED 24: Lower secondary education (typically ages 12-15)
- ISCED 34: Upper secondary education (typically ages 16-18)

K

K-12 - A term commonly used in North America to describe the school grades from kindergarten (K) through 12th grade (final year of secondary education). In European contexts, this corresponds approximately to ISCED levels 1 through 34.

Knowledge - The information, understanding, and awareness that a teacher needs to have.



M

Modelling and Simulation - The evaluation, modification, design, development, and understanding of models and simulations of natural and artificial phenomena and their evolution.

N

Networks and Communication - The understanding of how networks enable computing systems to share information via interfaces and protocols, and how networks may introduce risks.

P

PCK - Pedagogical Content Knowledge; a teacher's knowledge about how to teach specific content in a way that is appropriate for students.

PD - Professional Development; activities that help teachers develop knowledge and skills for effective teaching.

Pedagogical Competencies - Knowledge and skills related to how to effectively teach a subject, including teaching strategies, assessment methods, and classroom management.

Privacy, Safety, and Security - The understanding of risks when using digital technology, and how to protect individuals and systems.

Professional Growth - Ongoing development of subject expertise, pedagogy, and educational practice.

Programming - Writing and testing instructions in a programming language to solve problems or create digital artifacts. It involves using programming constructs (e.g. sequences, loops, conditionals) and applying computational thinking.

R

Responsibility and Empowerment - The critical and constructive analysis of concrete computing artifacts as well as advanced and potentially controversial techniques and applications of informatics, particularly from an ethical and social perspective.

S

Self-efficacy - An individual's belief in their ability to perform a specific task or achieve a specific goal.

Skills - The abilities and proficiencies that a teacher needs to demonstrate.

Standards for Computer Science Teachers - Guidelines that define what computer science teachers should know and be able to do to be effective in teaching. CSTA standards include five key areas: (1) CS knowledge and skills, (2) equity and inclusion, (3) professional growth and identity, (4) instructional design, and (5) classroom practice.

Subject-Specific Competencies - Knowledge and skills directly related to the content and concepts of a particular subject area.

T

Task - The specific action or responsibility that a teacher needs to perform.

Transversal Competencies - Professional skills and dispositions that support effective teaching practice across different subject areas and contexts.

U

Unplugged activities - Educational activities that teach computing concepts without using digital devices. E.g. sorting games, role-playing algorithms, kinaesthetic pattern activities.

4 Conclusion

This deliverable has outlined a comprehensive competencies framework for informatics teachers in primary education (ISCED 1), developed through a robust and collaborative process within the Digital First project. By integrating theoretical models such as the *Combined Knowledge and Competency (CKC) model* with the *Informatics4All framework* and *DigCompEdu*, and grounding the results in empirical data from WP2, WP3, interviews, and co-creative validation activities, the framework reflects both the pedagogical realities of classrooms and the evolving demands of digital society.

The resulting catalogue is structured across three key domains: subject-specific, pedagogical, and transversal competencies, each essential to preparing teachers who can effectively engage young learners in informatics. The emphasis on age-appropriate pedagogy, ethical digital practice, and interdisciplinary collaboration positions informatics not just as a technical subject but as a foundation for digital literacy, creativity, and critical thinking from the earliest years of schooling.

Survey responses and partner feedback underscore strong support for the proposed competencies, while also highlighting the significant gap between current and desired levels of teacher preparedness. This reinforces the urgent need for sustained investment in teacher education, continuous professional development, and institutional support to ensure that informatics education is equitable, relevant, and future-ready.

Ultimately, this competencies map provides a shared foundation for strengthening teacher preparation across Europe and contributes to the broader goal of establishing informatics as a core component of general education. It offers a practical tool for curriculum developers, teacher educators, and policymakers to support high-quality informatics instruction in primary schools and to empower young learners to participate actively and responsibly in the digital world.



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., Rován, B., & Webb, M. (2022). *Informatics reference framework for school*. National Science Foundation. <https://doi.org/10.1145/3592625>
- Caspersen, M., Gal-Ezer, J., McGettrick, A., & Nardelli, E. (2023). Informatics Education for School. *ACM Inroads*, 14(1), 49–53. <https://doi.org/10.1145/3583088>
- Clear, A., Clear, T., Impagliazzo, J., & Wang, P. (2020). From Knowledge-based to Competency-based Computing Education: Future Directions. *2020 IEEE Frontiers in Education Conference (FIE)*, 1–7. <https://doi.org/10.1109/FIE44824.2020.9274288>
- 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. (2024). *D2.2 Transnational research report: What and how we teach/learn in informatics and how happy we are with the results*. <https://digitalfirstnetwork.eu/deliverables/>
- Digital First Consortium. (2024). *D2.3. Recommendations* [internal report].
- Dialogue Club 1 – Personal function of digital technology [internal document].
- Dialogue Club 2 – Representational/informative function of digital technology [internal document].
- Dialogue Club 3 – Interactional function of digital technology [internal document].
- Dialogue Club 4 – Instrumental function of digital technology [internal document].
- 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>
- Kim, S., Jang, Y., Choi, S., Kim, W., Jung, H., Kim, S., & Kim, H. (2021). Analyzing Teacher Competency with TPACK for K-12 AI Education. *KI - Kunstliche Intelligenz*, 35(2), 139–151. <https://doi.org/10.1007/s13218-021-00731-9>
- Kumar, A. N., Becker, B. A., Pias, M., Oudshoorn, M., Jalote, P., Servin, C., Aly, S. G., Blumenthal, R. L., Epstein, S. L., & Anderson, M. D. (2023). A Combined Knowledge and Competency (CKC) Model for Computer Science Curricula. *ACM Inroads*, 14(3), 22–29. <https://doi.org/10.1145/3605215>
- Passey, D., Brodник, A., Caspersen, M., Černochová, M., Chaikovska, O., Dagiene, V., Keane, T., Leahy, D., Nančovska Šerbec, I., Parslow, T., Reffay, C., Syslo, M., Tuperova, D., & Turcsányi-Szabó, M. (2024). *Teacher education and teacher support associations for informatics and computing in schools, 2024 Edition*. <https://inria.hal.science/hal-04642206>
- Redecker, C., & Punie, Y. (2017). *Digital Competence of Educators DigCompEdu*. <https://doi.org/10.2760/159770> (online)

6 Annexes

Annex 1 – Competencies Reviewed in the Co-Creative Workshop

Digital First: defining informatics teacher competencies ISCED 1 (primary school)

A. SUBJECT-SPECIFIC COMPETENCIES

- A.1. Core Informatics Knowledge
 - A.1.1. Understanding data and information (representation, organization, analysis)
 - A.1.2. Knowledge of algorithms and computational thinking principles
 - A.1.3. Understanding basic programming concepts and creating simple programs
 - A.1.4. Knowledge of computer systems and networks fundamentals
 - A.1.5. Understanding digital security and privacy basics
- A.2. Contemporary Digital Technologies
 - A.2.1. Understanding basics of data science and visualization
 - A.2.1. Knowledge of age-appropriate AI and machine learning concepts
 - A.2.2. Understanding educational applications of robotics
 - A.2.3. Basic knowledge of computer graphics and multimedia
 - A.2.4. Understanding social networks and their implications

B. PEDAGOGICAL COMPETENCIES

- B.1. Teaching methodology
 - B.1.1. Ability to design age-appropriate learning activities
 - B.1.2. Skills in implementing active and experiential learning
 - B.1.3. Capability to use unplugged activities for teaching computing concepts
 - B.1.4. Ability to facilitate collaborative learning
 - B.1.5. Skills in integrating computational thinking across subjects
- B.2. Assessment and Evaluation
 - B.2.1. Ability to assess computational thinking skills
 - B.2.2. Competence in evaluating practical programming skills
 - B.2.3. Skills in implementing formative assessment strategies
 - B.2.4. Ability to provide constructive feedback
 - B.2.5. Capability to use digital assessment tools
- B.3. Inclusive Education
 - B.3.1. Ability to adapt content for different learning needs
 - B.3.2. Skills in providing inclusive computing education
 - B.3.3. Competence in supporting advanced learners
 - B.3.4. Ability to address gender-specific challenges
 - B.3.5. Skills in managing diverse classroom dynamics

C. PROFESSIONAL DEVELOPMENT AND TRANSVERSAL COMPETENCIES

- C.1. Professional Growth
 - C.1.1. Active participation in teacher networks and communities
 - C.1.2. Sharing experiences with other primary informatics teachers
 - C.1.3. Following current trends in primary informatics education
 - C.1.4. Contributing to school's CS/informatics curriculum development
- C.2. Cross-cutting Skills
 - C.2.1. Managing practical programming and CS activities
 - C.2.2. Organising physical and digital classroom resources effectively



- C.2.3. Explaining complex CS concepts in child-friendly ways
 - C.2.4. Collaborating with teachers to integrate computational thinking
- C.3. Professional Values
 - C.3.1. Fostering computational thinking mindset in young learners
 - C.3.2. Supporting colleagues in CS teaching development
 - C.3.3. Advocating for quality CS education at primary level
 - C.3.4. Promoting responsible and sustainable computing practices



Annex 2 – Practical Examples for Subject-Specific Competency Areas

Data and Information

Example 1: Favourite Book

Pupils conduct a class vote on their favourite storybook character, tally results on the board, and then create a pictograph with icons representing each vote.

- Representational function: focus on visualising and evaluating information.

Example 2: Favourite Book

Students log how many minutes they spend on different after-school activities (e.g., playing outside, homework, screen time, helping at home) for three days. In class, they group the activities into categories, calculate total time spent per category, and create bar charts on paper or using a simple digital tool. Together, they interpret the results by asking: “Which activity takes the most time?”, “Are our afternoons balanced?”, “Would we like to change anything?”

- Heuristic function: students explore and interpret patterns in their daily routines.
- Instrumental function: they record time and create bar charts using simple tools.
- Representational function: students present collected data with charts.

Example 3: Weather Tracker

Over one week, students record daily weather conditions (e.g., sunny, cloudy, rainy, snowy) using observation or information from a weather website. They sort the data by category, create a simple bar chart or table, and then compare the week’s weather with their expectations for the season. The teacher prompts discussion: “Was this week typical for the time of year?”, “How could this data be useful to plan activities?”

- Heuristic function: students explore relationships between observed data and seasonal patterns.
- Instrumental function: they enter weather observations into a simple data table and create charts.
- Imaginative function: they suggest possible uses of the data (e.g., for planning a school trip, planting in the garden).
- Representational function: students present collected data with charts.

Example 4: Classroom Library Tracker

Students track how many books are borrowed from the classroom library over one month. Each week they record the number of borrowed books, the genre (e.g., picture books, stories, popular science), and returned books. At the end of the month, they group the data by genre, create bar charts (by hand or digitally), and compare which genres are the most popular. They discuss possible reasons for the popularity of certain genres and whether new books should be added based on student interests.

- Heuristic function: students explore patterns in book choices and discuss possible reasons.
- Instrumental function: they record numbers and create charts.
- Representational function: they visually present results using diagrams.
- Imaginative function: they suggest ideas for new titles or library improvements based on the collected data.

Example 5: Leftovers Survey

(1) Collecting the Data

The teacher introduces the project by explaining that students will act as “food detectives” to see how much lunch is left over at school. Each day, students from different classes record how many pieces of fruit, bread rolls, or portions of hot food are left uneaten after lunch. They use tally marks or a simple count chart. This data collection continues for a set period (e.g., one week or one month).

(2) Organising the Data

Once the recording period ends, the class works together to organise the data. They create simple tables showing the number of leftovers per day and per grade level (e.g., Year 3 vs Year 4). Older students can also help to calculate totals and averages.

(3) Representing the Data

Students represent their findings visually. Younger children might use pictograms (e.g., an apple symbol for each leftover fruit), while older students can create bar charts or line graphs using paper or digital tools (Excel, Google Sheets, or classroom apps). These visuals make it easier to compare leftovers across days and between grade levels.

(4) Analysing and Discussing the Data

Together, the class examines the graphs: Do some days have more leftovers than others? Is there a difference between year groups? Are there particular foods that are always leftover?

The teacher guides students in discussing possible reasons (e.g., “On rainy days, fewer people eat outside, so less fruit gets eaten” or “Year 5 seems to leave more bread than Year 4”). Students can also suggest improvements, like menu changes, reminders to finish food, or ways to reduce waste.

- Instrumental function: Students use both digital and non-digital tools to collect, organise, and visualise data on leftover lunch. They practise recording observations, creating tables, and producing charts or pictograms that help make patterns clear.
- Heuristic function: Students decide how to organise and represent the data, transforming raw figures into meaningful information about their weekly lunch habits. In doing so, they practise self-organisation, periodic updating of records, and prioritising tasks such as preparing graphs or leading discussions.
- Representational function: Students communicate their findings visually by creating diagrams, graphs, and tables. These representations allow them to compare results across time periods and grade levels, making trends and differences easy to see and interpret.
- Imaginative function: Students use the insights gained from their data to propose creative ideas, such as new menu titles, suggestions for reducing waste, or broader improvements to the school lunch experience. In this way, they move from data analysis to innovation.

Example 6: Exploring Data with Bebras Tasks

Students work with Bebras tasks (from their national Bebras challenge or printed Bebras task cards) that focus on data-related concepts such as data understanding, data representation, and data mining. They solve these tasks in pairs, then discuss their reasoning and compare different solution strategies.

- Heuristic function: students use Bebras tasks to explore how observed data can be represented and to discover patterns or relationships hidden within the data.
- Imaginative function: after solving the Bebras tasks, they propose possible applications of the data in their own daily lives (e.g., planning routines, tracking activities, or recognizing trends in habits).

Algorithms

Example 1: Morning Routine Algorithm

Students work in small groups to break down the process of getting ready for school into step-by-step instructions. They write each step on separate cards, arrange them in order, and then test the sequence by acting it out. If the process fails (e.g., they “put on shoes before socks”), they debug by rearranging the steps. The teacher introduces the concepts of sequence and debugging in a real-life context.

- Heuristic function: students explore how processes work and learn from mistakes.
- Instrumental function: they use the step-by-step format as a tool for organising actions logically.
- Representational function: the physical cards visually represent the algorithm.



Example 2: Everyday Process Deconstruction

Students choose a familiar routine (e.g., making a sandwich, tying shoelaces). In groups, they break it down into precise algorithmic steps, identify possible errors (e.g., forgetting to open the bread bag), and present the “debugged” version to the class.

- Representational function: students abstract everyday experiences into structured, testable algorithms, reinforcing clarity and sequencing.

Example 3: Robot on the Map

Students draw a simple map where a “robot” must follow a set of steps to reach a goal (e.g., deliver a package). They create the path and write clear, step-by-step instructions using arrows or action cards (e.g., “move forward”, “turn right”). In pairs, they follow each other’s instructions to test the algorithm and revise it if needed.

- Imaginative function: students design a playful robot scenario with a goal and a visual path.
- Instrumental function: they build and test step-by-step instructions using cards (unplugged) or block-based programming tools.

Example 4: Treasure Hunt with Conditions

The teacher sets up a classroom “treasure hunt” with a set of conditional instructions (e.g., “If you see a red card, turn left; if you see a blue card, pick it up and move forward two steps”). Students follow the instructions to reach the “treasure” (a small prize or fun message). Afterwards, they discuss how conditions change the flow of an algorithm and how similar logic is used in digital games.

- Heuristic function: students test and refine their understanding of conditional logic.
- Instrumental function: they apply simple “if... then” structures to solve a problem.
- Imaginative function: they experience algorithms as an interactive, game-like adventure.

Example 5: Unplugged Bebras Tasks

Students work in pairs using Bebras challenge tasks, which can be downloaded from the Bebras website or prepared on printed cards. The teacher selects tasks related to the topic being introduced. In pairs, students solve the tasks by discussing possible solutions, checking each other’s reasoning, and exploring different approaches. This encourages collaboration, critical thinking, and clear explanation of ideas.

- Heuristic function: students test and refine their understanding of core computer science concepts such as conditional statements, loops, and sequences.
- Instrumental function: they apply these concepts (e.g., simple if...then rules or repeated actions) to solve problems step by step.

Programming

Example 1: Guide a character

Student guides a character from one position on the grid to another, avoiding obstacles and repeating the same pattern of movements.

- Instrumental function: students solve simple problems.

Example 2: Unplugged Robot Commands

In pairs, one student is the “robot”, and the other is the “programmer.” The programmer gives step-by-step instructions (e.g., “Move forward 3 steps,” “Turn left,” “Pick up the ball”). The teacher introduces loops for repeated actions and conditionals (“If you see a red card, turn right”). Students then swap roles and test different “programs.”

- Heuristic function: students test, observe, and debug their instructions.
- Instrumental function: they use algorithms to control actions in a physical space.
- Regulatory function: they follow and adapt rules to ensure the “robot” completes the task.



- Representational function: the verbal or written instructions represent program code.

Example 3: Animate a Story

Students use ScratchJr or Scratch to create a short animation of a simple story, such as “A cat goes to the park.” They program the characters to move, speak, and react using sequences, loops (e.g., repeated walking), and conditionals (e.g., “If it’s raining, open an umbrella”). After programming, they present their animations to the class and explain which programming blocks they used.

- Heuristic function: students explore how combining blocks changes the program’s outcome.
- Instrumental function: they apply programming constructs to achieve a creative goal.
- Representational function: animations visually represent the sequence and logic.
- Imaginative function: students bring original story ideas to life through code.

Example 4: Treasure Hunt Maze

Students program a simple maze in Scratch where a character moves with arrow keys to reach a treasure. They use basic blocks for movement and add a condition like “if touching wall, go back.” Students test each other’s mazes and suggest small improvements.

- Instrumental function: they use basic blocks to make a working game.
- Heuristic function: they test and correct mistakes in the maze.
- Imaginative function: they design their own maze paths and treasure themes.

Example 5: Interactive Quiz Creator

Using a block-based environment (e.g. ScratchJr, Scratch), students design a quiz that reacts to user inputs (e.g., multiple-choice answers with immediate feedback). The teacher encourages debugging and iterative improvement.

- Instrumental function: students apply programming constructs (loops, conditions, events) to build functional, interactive tools.
- Heuristic function: pupils explore conditional logic through discovery.

Computing Systems

Example 1: Everyday Use of Computer

Students present examples of tasks for which they use computers and present which input and output devices they need for each task.

- Representational function: students connect names of devices with their use.

Example 2: Computer Parts Scavenger Hunt

The teacher places pictures or real examples of computer components (e.g., mouse, keyboard, screen, speaker, USB drive, microphone) around the classroom. Students work in pairs to find each item, identify whether it is an input or output device, and note its function. After the activity, the class discusses how these components work together in a computer system and shares real-life examples of when they have used them.

- Heuristic function: students actively search for and identify components through exploration.
- Instrumental function: they categorise devices as input or output and record their functions.
- Representational function: they present their findings to the class using a chart or diagram.
- Imaginative function: they suggest new or improved devices that could help in everyday school or home activities.

Example 3: Tech Match-up

Students receive cards (optionally Bebras cards) with the names of devices (e.g., keyboard, mouse, camera, monitor) and cards with descriptions of their functions. Working in pairs, they match each device name to the correct description.



They then classify the devices as input or output and discuss how they work together to complete a task (e.g., writing a document, recording a video).

- Heuristic function: students discover the functions of devices through a matching task.
- Instrumental function: they classify devices as input or output.
- Representational function: they visually present the device-function pairs.

Example 4: Digital Systems in Everyday Life

Students analyse different contexts (e.g., school library, hospital, supermarket) to identify the role of hardware, software, and networks. They then present system maps showing how these elements interact to deliver services.

- Representational function: students create conceptual models linking abstract computing concepts to tangible real-world examples.

Digital Creativity and Design

Example 1: Creating Digital Artefacts

In pairs, students design a digital poster or short animation introducing an imaginary animal (e.g., "a rainbow-striped flying rabbit"). They describe its special features, habitat, and favourite food using images, text, and sounds.

- Imaginative function: they invent and express creative ideas through their animal.
- Interactional function: they collaborate, discuss, and co-create in pairs.

Example 2: My Digital Comic Strip

Students create a short digital comic about a school event or an imaginary story using a simple tool (e.g., Canva, Pixton). They include images, speech bubbles with text, and sound effects. After creating the first version, they exchange feedback in pairs and make improvements before presenting the final comic.

- Imaginative function: they create an original story and characters.
- Interactional function: they collaborate during the feedback and improvement phase.
- Instrumental function: they use digital tools to create and edit the comic.
- Representational function: they present the completed comic to the class or online.

Example 3: Digital School Campaign

Students work in small groups to design a digital poster or short video campaign promoting a positive school value (e.g., kindness, recycling, healthy eating). They use simple creative tools such as Canva, Book Creator, or Scratch to combine images, text, and audio. After presenting their work to the class, they gather peer feedback and make small improvements before publishing it on the school website or notice board.

- Imaginative function: students generate original ideas for their campaign message and visual style.
- Interactional function: they collaborate in groups, share tasks, and discuss design choices.
- Instrumental function: they use creative software tools to produce and edit the content.
- Representational function: they present their finished work to the class and/or school community.

Example 4: Create New Machine

In pairs students use a digital program to create original machines (monstrous, fanny, etc.) using simple blocks and features. They explain their creation to other students. In a second step, students, guided and supported by teachers, can make a lab to realise those machines concretely with various materials (e.g. recyclables, etc.)

- Imaginative function: students can express their creativity thinking to new strange machines with unusual functions. Explaining peculiarities and functions of the machines to the other students it is a way to further reinforce creative and out of the box thinking.

Example 5: Educational Video

In small groups, students plan and create a short educational video to explain a mathematics concept (e.g., multiplication, division, fractions). The video is then presented to the whole class for feedback and reflection. Steps:

- (1) Planning the video: Each group selects a maths topic (e.g., “How to multiply by 10” or “Division as sharing”). Students brainstorm creative ways to explain the concept (storytelling, role-play, drawing on a whiteboard, stop-motion with objects, digital animation tools). They write a short script or storyboard, outlining key points and visual elements.
- (2) Producing the video: Using tablets, smartphones, or classroom computers, groups record their explanation. They may combine live action, drawings, voice-over narration, or digital apps (such as iMovie, Clips, or Canva) to bring their ideas to life. The emphasis is on clear communication and creative presentation, rather than technical perfection.
- (3) Presenting and sharing: Each group shows their finished video to the class. Classmates and the teacher give constructive feedback on clarity, creativity, and accuracy of the mathematical explanation.
- (4) Reflecting and improving: Students discuss what worked well and what could be improved in terms of both mathematical content and digital design choices. If time allows, groups may refine their videos based on feedback.
 - Instrumental function: Students use digital tools (e.g., tablets, smartphones, editing software, animation apps) to plan, record, and edit a short educational video. They learn to apply concrete skills such as storyboarding, filming, and combining audio-visual elements.
 - Heuristic function: Students make decisions about how best to explain and present a mathematical concept, experimenting with different formats (role-play, stop-motion, drawings, animations). Through this process, they develop problem-solving strategies, practise self-organisation within groups, and refine their ideas based on feedback.
 - Representational function: Students transform abstract mathematical concepts into accessible visual and auditory representations. By scripting and designing the video, they communicate knowledge in multimodal forms (images, sounds, text, movement), making mathematical reasoning visible and understandable for peers.
 - Imaginative function: Students apply creativity to design engaging explanations – perhaps turning multiplication into a short story or illustrating division through a cooking or sports scenario. They explore innovative ways to make mathematics relatable and use feedback to further enhance their creative designs.

Example 6: Design Thinking with Digital Prototypes

Students identify a classroom problem (e.g., lost items, noisy transitions) and use design thinking steps: empathize, define, ideate, prototype, test. They create digital mock-ups (apps, posters, or interactive boards) to propose solutions.

- Heuristic function: students explore problem-solving through cycles of testing and improvement, learning by designing solutions iteratively.

Example 7: Bebras Task Design

Students work in small groups to create their own Bebras-style tasks on informatics topics (e.g., sequences, logical conditions, or simple algorithms). Using a Bebras task template (e.g., from bebras.org), they draft the text in a text editor and prepare supporting visuals with simple drawing tools or image editors. After presenting their tasks to the class, they gather peer feedback and refine the wording, visuals, or logic. The final tasks can be shared on the school website or displayed on a notice board to inspire other student.

- Imaginative function: students generate original ideas for their Bebras task theme, informatics concepts, and visual style.
- Representational function: they present their completed tasks to the class and/or school community, collecting feedback and refining their work.



Modelling and Simulation

Example 1: Traffic Light System

Students create a basic traffic light system using block-based programming environments (red, yellow, green sprites with timing), programmable robots with motion sensors that trigger light changes when objects approach, or unplugged role-play with coloured paper circles and timer commands.

- Instrumental function: students build and automate a system using digital tools (Scratch, robots).
- Heuristic function: they test, adjust, and understand how timing and inputs affect system behaviour.

Example 2: Automatic Plant Watering Model

Students create a simple simulation of an automatic plant watering system. They attach a moisture sensor to the micro:bit (or simulate moisture levels using button inputs). The micro:bit displays a water droplet icon when the “soil” is dry and a happy plant icon when it’s moist. They can add a buzzer to signal when watering is needed. Students discuss how similar systems work in agriculture.

- Regulatory function: the micro:bit controls a simulated response based on input data.
- Instrumental function: students use micro:bit hardware to model an automated process.
- Imaginative function: they design and personalise the plant watering alert system.

Example 3: Extreme weather response simulator

Students use micro:bit, Scratch, or another block-based tool to create a simulation of a town’s emergency response system for extreme weather (e.g., heavy rain, heatwave, snowstorm). They program the system so that when a certain “weather” input is given (button press, sensor reading, or sprite selection), the simulation triggers appropriate responses such as closing roads, sending safety alerts, or showing evacuation routes. Students can expand the simulation by adding multiple variables like wind speed, temperature, or rainfall level.

- Heuristic function: students experiment with different weather inputs and observe how changes affect the simulated response.
- Instrumental function: they design and code automated responses using block-based programming or sensors.
- Regulatory function: their simulation models rule-based actions for public safety.
- Imaginative function: they customise the town, the events, and the emergency responses.

Example 4: Designing a Virtual Ecosystem

Students use a simple simulation tool (e.g. NetLogo, Scratch, or an age-appropriate app) to model an ecosystem with plants, herbivores, and predators. They adjust parameters such as the number of animals, food availability, or reproduction rates and observe how the system changes over time.

- Heuristic Function (exploratory learning): Students explore cause-and-effect relationships (e.g. what happens if too many predators are added?). They learn through discovery by testing hypotheses.
- Instrumental Function (problem-solving): The tool helps structure and visualise otherwise complex interactions. It supports students in applying logic and rules to manage variables and outcomes.
- Imaginative Function (creative modelling): Students creatively design their own ecosystem, inventing species, rules, or environmental conditions, stimulating imagination and divergent thinking.

Example 5: Math in Motion

Students use a simulation tool to model geometric shapes or patterns (e.g., tessellations, fractals). They change parameters (angle, repetition, scaling) and observe how patterns emerge.

- Instrumental function: students use digital tools to generate, test, and refine mathematical models.



Privacy, Safety and Security

Example 1: Private vs. Public

Students play a sorting game where they receive cards showing different types of information (e.g., photo of a pet, home address, favourite food). They decide whether each item is safe to share online or not and place the cards into "Safe to share" and "Keep private" boxes.

- Regulatory function: they practice making decisions about what information is safe.
- Instrumental function: they physically sort and classify information.
- Personal function: understanding of privacy.

Example 2: Escape the Hacker

The classroom is set up as a "digital escape room" where students must solve a series of password-related challenges to "lock out" a fictional hacker. Each challenge teaches an aspect of password safety: decoding a weak password to see why it's unsafe; rearranging mixed-up characters to form a strong password; identifying phishing messages that try to steal passwords. Teams must complete all tasks to escape, with the final step being the creation of a class "Strong Password Rulebook."

- Heuristic function: students discover password safety principles through problem-solving.
- Instrumental function: they apply these principles to create strong passwords.
- Interactional function: they collaborate in teams to complete challenges.
- Regulatory function: they establish and agree on safe password rules.
- Imaginative function: they work within a game storyline to engage with the topic creatively.

Example 3: Respectful Chat Simulation

In small groups, students role-play an online group chat scenario with prepared "message cards." Some messages show positive, respectful behaviour, while others show inappropriate or unsafe communication. Students sort the messages into "Respectful" and "Needs improvement" categories, then rewrite the unsafe messages to make them appropriate.

- Regulatory function: students apply rules for safe and respectful online communication.
- Instrumental function: they sort and rephrase messages.
- Personal function: they reflect on how online tone and language affect others.
- Imaginative function: they creatively reword inappropriate messages into positive ones.

Example 4: Digital Footprint Detective

Students trace the kind of information that might be left behind when using websites or apps (through teacher-prepared examples). They role-play as "detectives" who identify what others could learn about them from this data.

- Heuristic function: students critically explore hidden aspects of online behaviour and learn to question risks.

Example 5: Create a Digital Footprint Map

Students brainstorm and then draw (on paper or using a simple mind-mapping tool) the types of personal information they might share online (e.g. name, birthday, school, photos). They then classify which pieces are safe to share publicly, which should only be shared with friends/family, and which must stay private.

- Heuristic Function (exploratory learning): Students discover what counts as sensitive information and why, by analysing examples and discussing possible consequences.
- Instrumental Function (problem-solving): The mapping activity provides a concrete method to categorise information, helping them apply rules for safe online behaviour.
- Imaginative Function (creative modelling): Students imagine scenarios (e.g. "What if someone I don't know sees this photo?") and role-play responses, developing empathy and foresight.

Artificial Intelligence

Example 1: Teach the Robot

Students show different pictures (e.g., cats and dogs) to a "robot teacher" (themselves or a classmate) and help it learn to tell them apart by giving "yes/no" feedback.

- Heuristic function: they explore and correct the robot's guesses.
- Instrumental function: they use examples to improve the robot's sorting.

Example 2: AI Learns Rock, Paper, Scissors

Students use a simple AI tool (e.g., Teachable Machine with video input) to train a model that recognises hand shapes for rock, paper, and scissors. Students hold their hands in front of a plain background so only the hand gesture is visible. They collect a set of training images for each gesture.

They test the AI by showing it new hand shapes and seeing if it guesses correctly. Afterwards, students discuss: Why the AI might confuse similar gestures. How lighting, background, and gesture size affect recognition. How adding more varied examples could improve accuracy.

- Heuristic function: students experiment with training data to see what affects recognition.
- Instrumental function: they use a digital tool to build and test a simple AI model.
- Regulatory function: they adjust datasets to improve the model.
- Imaginative function: they invent new gestures and see how the AI reacts.

Example 3: Scratch Translator & Speaker

Students build a mini translator that takes a typed word and plays the translated word out loud. Using the Translate and Text to Speech extensions, they: choose two languages (e.g., Croatian → English), design a simple input interface, and program: when the user types a word, the project translates it and then speaks the result. Students add a "slow/fast" pronunciation toggle and a small "word bank" that stores successful translations with icons (picture dictionary). They test edge cases (accents/diacritics, uppercase/lowercase) and discuss why some words need context to translate well.

- Instrumental function: students combine Translate and Text to Speech blocks to build and test the tool.
- Heuristic function: they experiment with spelling, diacritics, and different target languages to observe effects.
- Interactional function: they work in pairs, test each other's words, and give quick feedback ("Did it sound right?").
- Regulatory function: they set class rules for safe use (no personal data; appropriate vocabulary).
- Representational function: they present a short demo and a simple flowchart showing "input → translate → speak."

Example 4: AI Sorting Hat

Students design a 'magic hat' game where an AI tool (or a simplified mock-up activity with pre-labelled cards) suggests which hobby, book, or sport a student might like based on a few preferences (e.g., 'I like animals', 'I enjoy running'). Afterwards, they compare the AI's guesses with their real choices and reflect on how recommendation systems work and why they sometimes miss.

- Heuristic function: they explore how recommendations rely on patterns and preferences.
- Instrumental function: they use a tool or structured dataset to generate suggestions.
- Regulatory function: they discuss mismatches and biases in automated recommendations.

Example 5: Examples of Good and Bad

All students should try to provide a set of examples of "good" and "bad" behaviour to the "AI" (a classmate or teacher), and they should try to evaluate the capacity of the AI to distinguish between good and bad, to check if the ethical side of things has been covered.

- Regulatory function: they practice making decisions about what information is save.



Example 6: AI in Daily Life Debate

The teacher introduces examples of AI (voice assistants, recommendation systems, chatbots). Students research one use case and present its benefits and risks (e.g., convenience vs. privacy issues). Class holds a debate on “Should AI make decisions for us?”

- Representational function: students analyse and articulate societal implications of AI through structured argumentation.

Computational Thinking (CT)

Example 1: Creating Presentation

Students need to make a group presentation of selected habitat. They must create a plan for their research, divide the task into smaller parts, find and select reliable resources, and combine this data for presentation.

- Interactional function: students collaborate, divide roles, and co-create a shared digital product.
- Representational function: they express ideas visually using text, images, and layout to communicate a message.

Example 2: Robot story path

Using a large sheet of paper or the classroom floor with taped squares, students create a “story map” where each square represents part of a narrative (beginning, middle, end, problem, solution). In pairs, one student plays the “robot” and the other gives step-by-step instructions to move through the story map in the correct sequence. This helps students practise decomposition (breaking the story into parts) and algorithmic thinking (step-by-step commands).

- Heuristic function: students explore how to break down a story into logical parts.
- Interactional function: they work in pairs, switching roles.
- Instrumental function: they create and follow step-by-step instructions.
- Representational function: they display the final story path visually.

Example 3: Mystery pattern detectives

The teacher gives students a series of “mystery cards” with shapes, numbers, or symbols that follow hidden patterns. Working in small groups, students must identify the pattern, predict the next item, and explain their reasoning. They then create their own pattern sequences for another group to solve.

- Heuristic function: students discover and explain underlying rules in patterns.
- Interactional function: they collaborate and challenge each other’s thinking.
- Instrumental function: they design and test their own pattern puzzles.
- Representational function: they present their findings and solutions clearly to others.

Example 4: Unplugged Sorting Algorithms

Students physically act out sorting a set of cards by height, weight, or colour. They try different strategies (bubble sort vs. divide-and-conquer) and compare efficiency.

- Representational function: students embody computational methods to understand algorithmic efficiency.

Example 5: Unplugged Computational Thinking with Bebras Tasks

Students work with Bebras tasks (these can be taken from the national Bebras challenge, translated into their language, or prepared as printed Bebras task cards). The teacher selects tasks that highlight different areas of computational thinking. For example, students might solve a task focusing on abstraction, another on pattern recognition, or one on decomposition. Working individually or in pairs, they discuss strategies, compare solutions, and reflect on how these approaches relate to problem-solving in informatics.

- Instrumental function: they apply these concepts systematically, for example by solving a task step by step using sequences, loops, or conditional statements.



- Representational function: students embody and represent computational methods, comparing different strategies and reflecting on algorithmic efficiency.

Example 6: Guide to Treasure

Students design step-by-step instructions (an algorithm) to guide a classmate from the classroom door to a “hidden treasure” (e.g. a toy or sticker) using directional commands (forward, turn left, turn right). Later, they test and debug the instructions together.

- Heuristic function (exploratory learning): Students test, fail, and refine their algorithms, discovering the importance of precision and sequence in problem-solving.
- Instrumental function (problem-solving): The algorithm acts as a tool to break down a big task into smaller steps, applying structured thinking and debugging strategies.
- Imaginative function (creative modelling): Students invent playful scenarios (pirate maps, robot guides, maze adventures) to frame their algorithmic instructions in a story-rich way.