Co-funded by the
Erasmus+ Programme
of the European Union

Modeling at School

Intellectual Output 2

Final Report, January 2020

JOHANNES KEPLER UNIVERSITÄT LINZ

Universidad Rey Juan Carlos

UNIVERSITY OF JYVÄSKYLÄ

Table of Content

1 Introduction

This report presents the final results of intellectual output 2, the ready-to-use-guideline for modeling. All the three partners from Austria, Spain and Finland have been working on the second output of the Erasmus+ project Modeling at School from March 2019 until the End of January 2020. In order to recall the aims of the first output, a detailed description can be read in the following paragraph.

Output description:

After having identified key competences, situations, contexts, activities, tasks etc. where modeling seems useful (intellectual output O1), it is necessary to select and describe suitable models and their purpose for these different issues in a form that can be understood even without computer science background.

Based on the analysis of primary and secondary school curricula and the developed overview (O1) we will categorize the different key competences, themes, contexts found etc. as well as modeling techniques and diagrams in order to connect them and show teachers where, in which situations, for which activities etc. which model or diagram can be useful. We will develop a guideline for teachers with essential information about the diagrams, their purposes and how to use them. We further present some examples of modeling in different fields of application that will be created in cooperation with teachers in our local workshops (see also A3 Local project activities in section G Project management and implementation). Due to our experiences with modeling in foreign language lessons the first models we use are ER-, class-, activity diagrams and flowcharts, but we will add also other diagram types that seem useful for the different purposes (e.g. use case diagrams, communication diagrams etc.). The best materials produced in these local workshops will be selected and adapted by the JKU team, leader of this output, and will serve as sample activities for the ready-to-use guideline for teachers. All other materials will be collected and added to those produced for output O5, the online collection of COOL teaching and learning materials.

Innovation:

Modeling has not been equally spread in each project partner country as a tool in other subjects than in those closely related (computer science or mathematics) and there is no such guideline up to now. Hence, teachers in project partner countries can learn from each others' practices and the jointly developed framework will be definitely innovative. From our experiences in the project "Informatics - A Child's Play?!" we know that modeling is a useful tool for teachers and learners in primary education, but they are unsure and need more information on how and where to use and apply it.

Impact:

With this practical guideline teachers and also learners will learn to apply a new and innovative method/strategy that supports learning in their subjects and get to know where they can integrate it in their subjects. They will get concrete hints that help them to apply suitable models and diagrams for certain learning situations, competencies, themes etc. in several subjects.

Transferability:

Besides subject specific hints the guideline will inform teachers and learners also how to train different general (learning) competences, like problem solving, using samples of everyday life. So it would be easy to transfer them also to other subjects or levels of education.

Procedure:

After the content analysis of different curricula in order to find out key competences, themes and situations (O1), we list them and group them in some few categories. Then we group the different models, find useful categories that we can finally link to the categories of competences, themes etc. Based on this work and our experiences with modeling in foreign language lessons, we describe suitable models, their purpose as well as necessary shapes and rules for their application. We will certainly integrate models that have already been considered as useful by teachers and students (flowcharts, entity-relationship-, class- and activity diagrams) but we will add others like use case or communication diagrams etc. The final step is writing the ready-to-use guideline for teachers that will be reviewed not only by project partners but also by the different participants (teachers, students, teacher educators).

During the period of Intellectual Output 2, we developed several elements for the guideline. In an initial phase, partners in Johannes Kepler University created introductory guidelines for modeling in German language which were then translated and adapted to English language. The introduction includes background information about modeling, with the following subsections: 1.1 Modeling (what), 1.2 Modeling (why), 1.3 Modeling (for what), 1.4 and 1.5 Modeling (how), 1.6 Modeling (with what), and more detailed information of 2.1 ER-diagram, 2.2 Class diagram, and 2.3 Activity diagram
(see https://drive.google.com/file/d/1b8u80u4Jghoje_9su4ndGJQhH-EeldgT/view?usp=sharing).

The other elements of guidelines are described more specifically in the following chapters. In the first chapter a guideline with the essential information of the different diagrams can be found. The next chapter presents a guideline for the assessment of the modeling competence, which is called "Framework of Reference for Modeling" (ReMo). The ReMo is followed by the categorization of key competences and a glossary to clarify technical terms. These four elements form the generic part of the ready-to-use guideline. Due to the fact that different education systems require different approaches for the implementation and dissemination of modeling, the second part of the guideline consists of country specific content (i.e. how the guidelines were adapted to each country). These country-specific guidelines will be described in greater detail in chapter 6.

2 Ready-to-use-guideline: Fact Sheets - Diagrams

In this chapter, the guideline of five diagrams, that have already been considered as very suitable for many subjects, are presented. These diagrams are the following:

- Entity-Relationship-Diagrams
- Class- & Object Diagrams
- Activity Diagrams
- Use-Case Diagrams
- Flowchart Diagrams

This guideline serves as a first introduction to modeling and therefore, the descriptions and sample diagrams are purposely kept very simple. During the pilot phase, the factsheets were checked for comprehensibility by several teachers and students and we subsequently adapted them accordingly. Besides a first introduction to modeling for multipliers, mentors and tutors, these factsheets can be used as hands-on material when these stakeholders implement their workshops in school. More detailed information about the different diagrams is available in two different versions. On the one hand, complementary information for the first three diagrams (entity-relationship diagram, class & object diagram and activity diagram) is provided in form of tutorial videos for a flipped-classroom approach (online training etc.). The remaining two diagrams (use-case diagram and flowchart), on the other hand, are further elaborated in interdisciplinary teams of the different stakeholders (mentors, multipliers, tutors) in face-to-face trainings. The following pages present the guideline for the diagram factsheets. The videos are described in further detail in chapter 6.

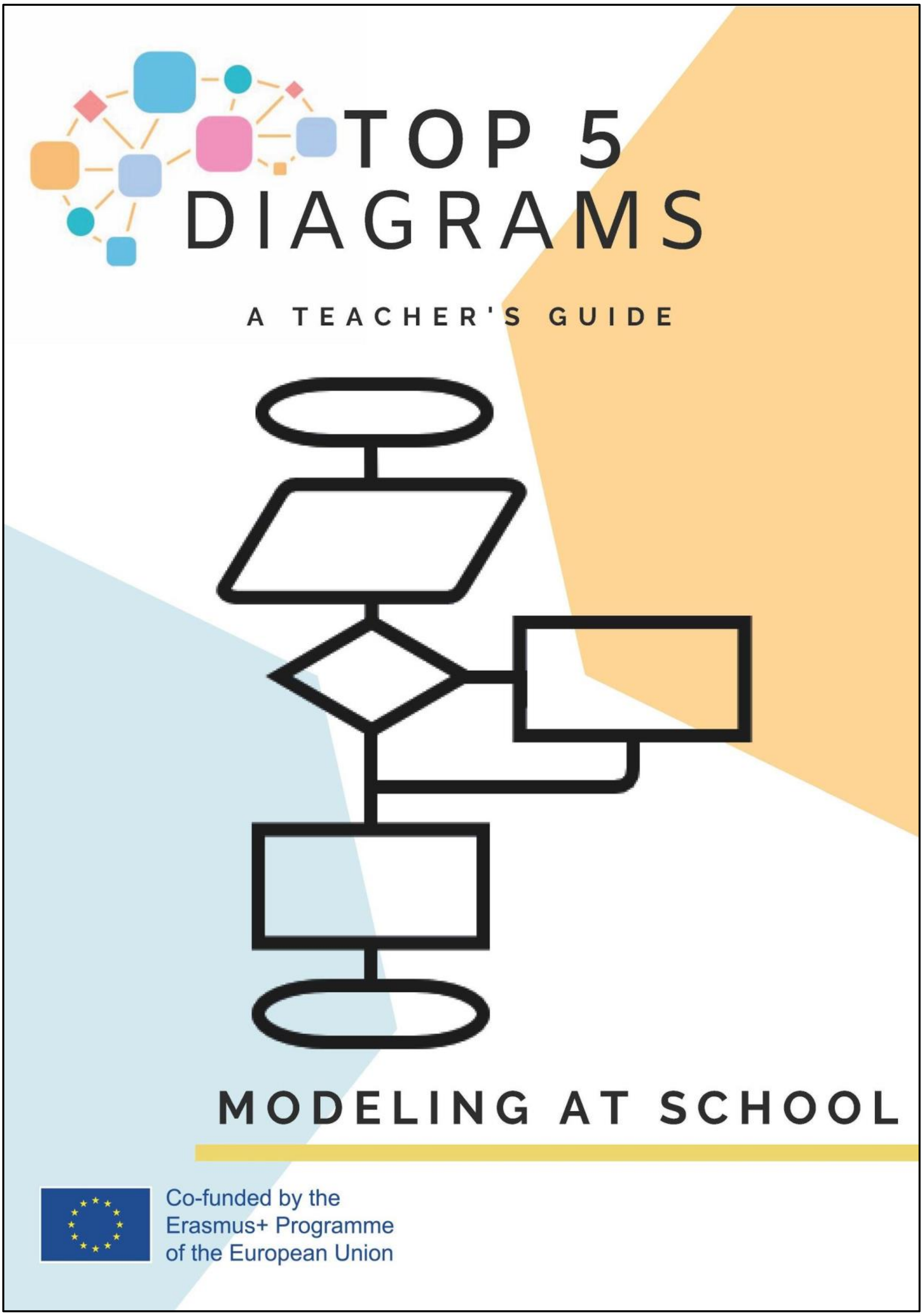


Figure 1: Diagram Guideline Page 1

TOP 5 DIAGRAMS // MODELING AT SCHOOL

01

ENTITY-RELATIONSHIP DIAGRAM

VISUALIZE SITUATIONS, STATES & RELATIONS

"Entity-relationship diagrams are perfect to begin with modeling. With just a few shapes, it is easy to acquire and helps to remove complex syntax by visualizing the most important elements of a text."

The entity-relationship diagram is a very common method of representation. For the reason that it is composed of only three basic elements, it is easy to understand. These elements are entity types, which describe a group of real objects, as in the example below "sheep", "grass" and "meadow". Relationship types between two or more entities represent a relationship, as for example the relationship "to eat" and "grows on". Attributes represent characteristics of entity types or relationship types, as for example "breed" or "size" of a sheep or the "location" and "area" of the meadow.

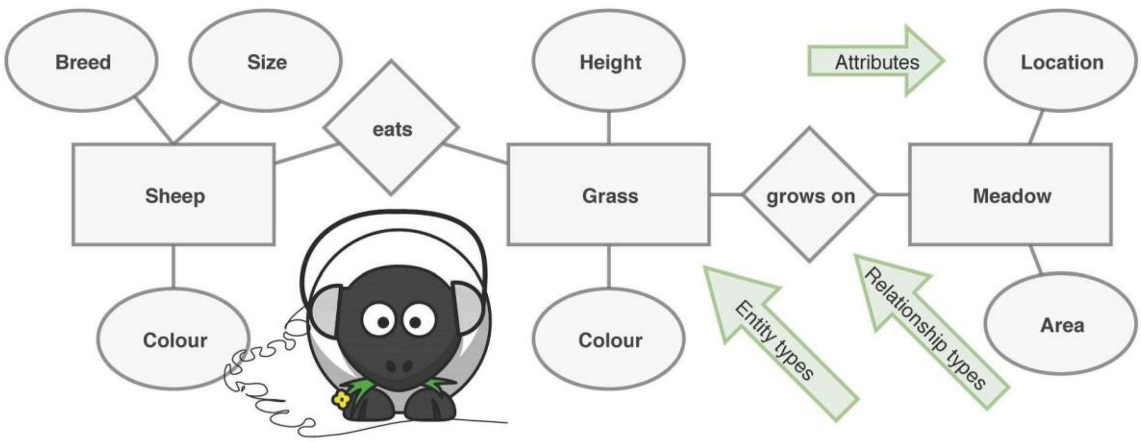


Figure 2: Diagram Guideline Page 2

TOP 5 DIAGRAMS // MODELING AT SCHOOL

02

CLASS & OBJECT DIAGRAM

STRUCTURE, CATEGORIZE AND BRING IMPORTANT TERMS ON PAPER

"The class diagram is an ideal tool to structure vocabulary and visualize hierarchies!"

A class diagram is a useful model to represent the characteristics of one or more objects. This model demonstrates the attributes and operations of single classes and possible interrelationships between the objects. With the class diagram, a representation of "inheritance" or "generalization" is feasible. In other words, the transfer of attributes and behavior from superclasses to subclasses can be shown. As in the example below, every driver is also a person and has therefore all the attributes of the class person. Relationships, that express a "consist of"-connection, are also known as "aggregations".

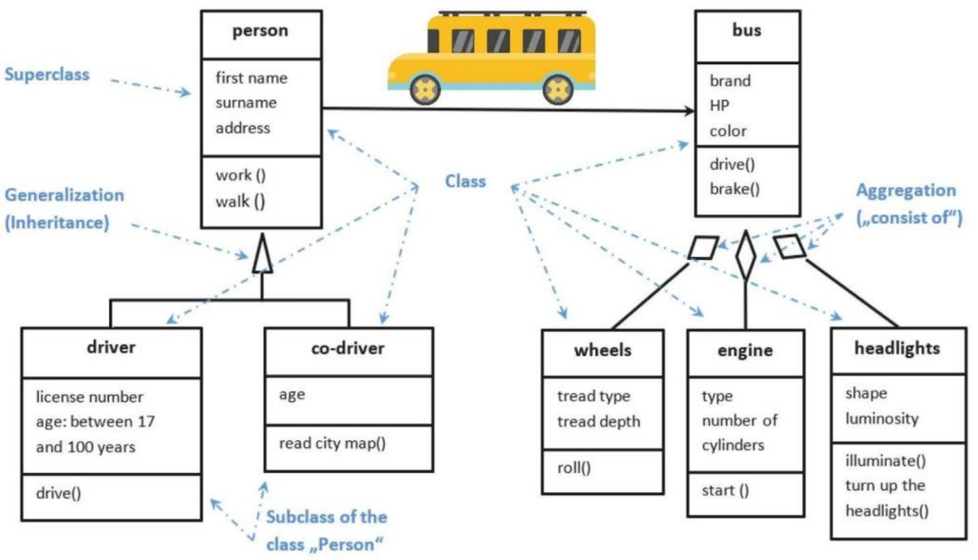
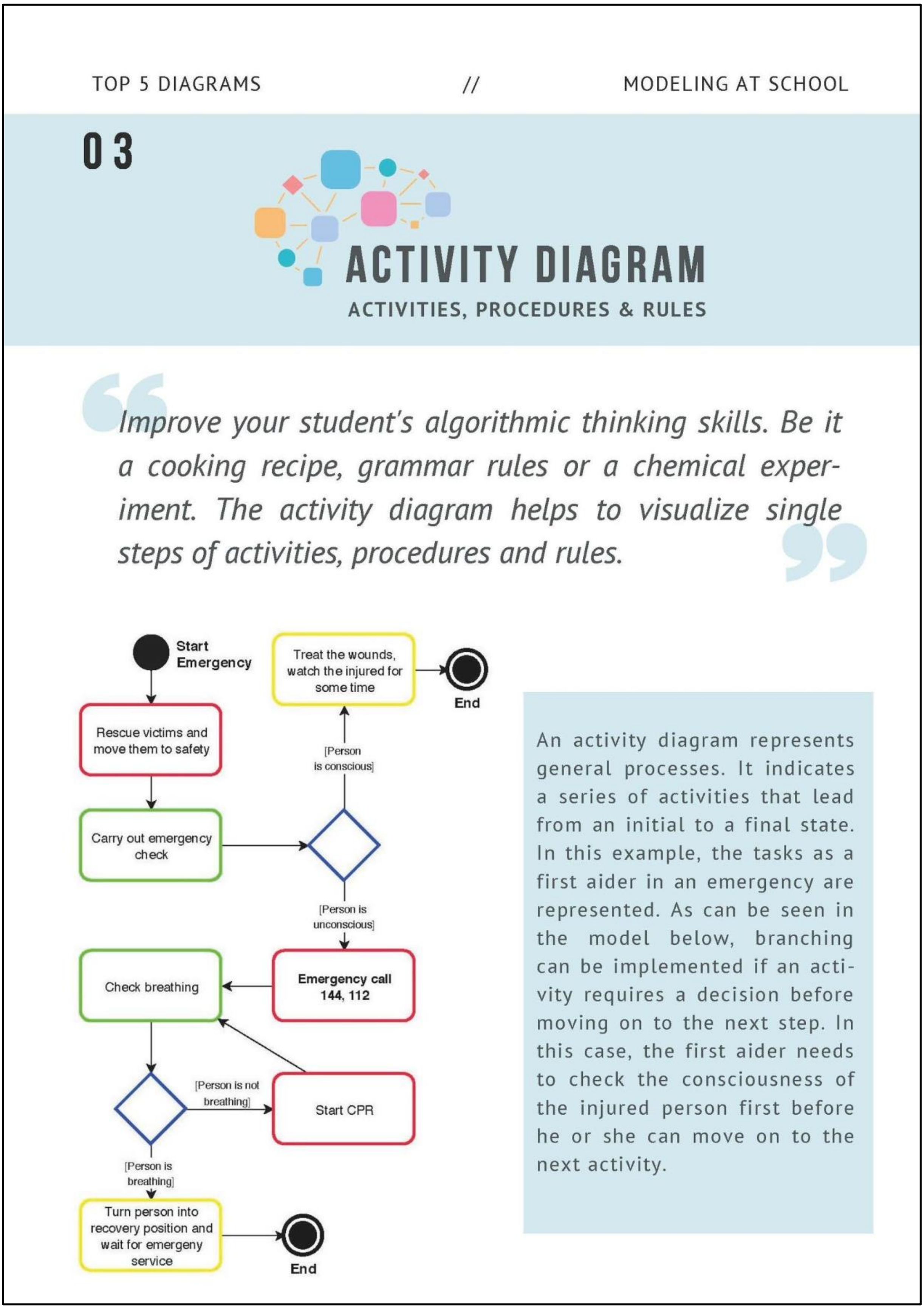


Figure 3: Diagram Guideline Page 3



TOP 5 DIAGRAMS // MODELING AT SCHOOL

03

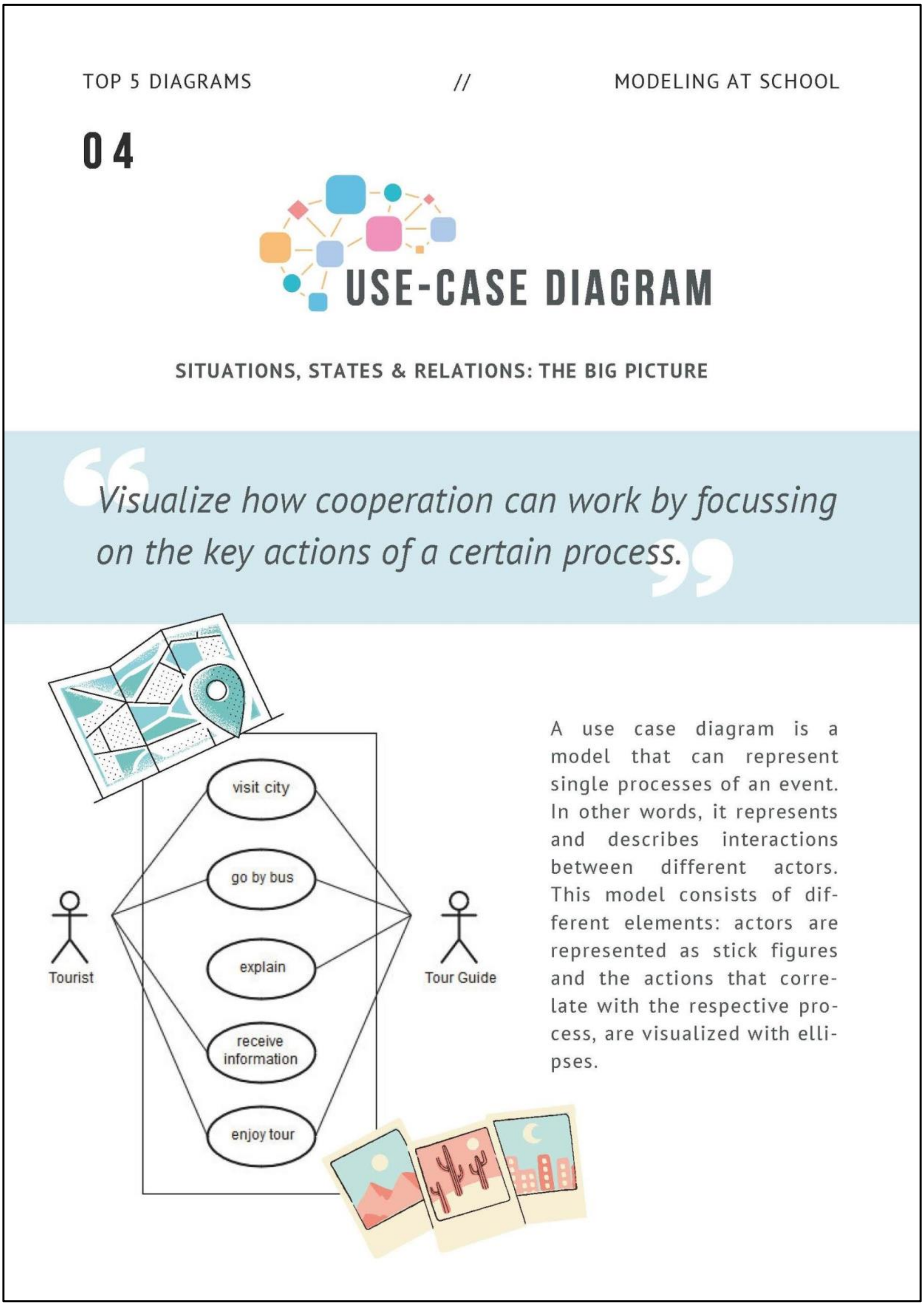
ACTIVITY DIAGRAM

ACTIVITIES, PROCEDURES & RULES

Improve your student's algorithmic thinking skills. Be it a cooking recipe, grammar rules or a chemical experiment. The activity diagram helps to visualize single steps of activities, procedures and rules.

An activity diagram represents general processes. It indicates a series of activities that lead from an initial to a final state. In this example, the tasks as a first aider in an emergency are represented. As can be seen in the model below, branching can be implemented if an activity requires a decision before moving on to the next step. In this case, the first aider needs to check the consciousness of the injured person first before he or she can move on to the next activity.

Figure 4: Diagram Guideline Page 4


TOP 5 DIAGRAMS // MODELING AT SCHOOL

04

USE-CASE DIAGRAM

SITUATIONS, STATES & RELATIONS: THE BIG PICTURE

"Visualize how cooperation can work by focussing on the key actions of a certain process."

A use case diagram is a model that can represent single processes of an event. In other words, it represents and describes interactions between different actors. This model consists of different elements: actors are represented as stick figures and the actions that correlate with the respective process, are visualized with ellipses.

Figure 5: Diagram Guideline Page 5

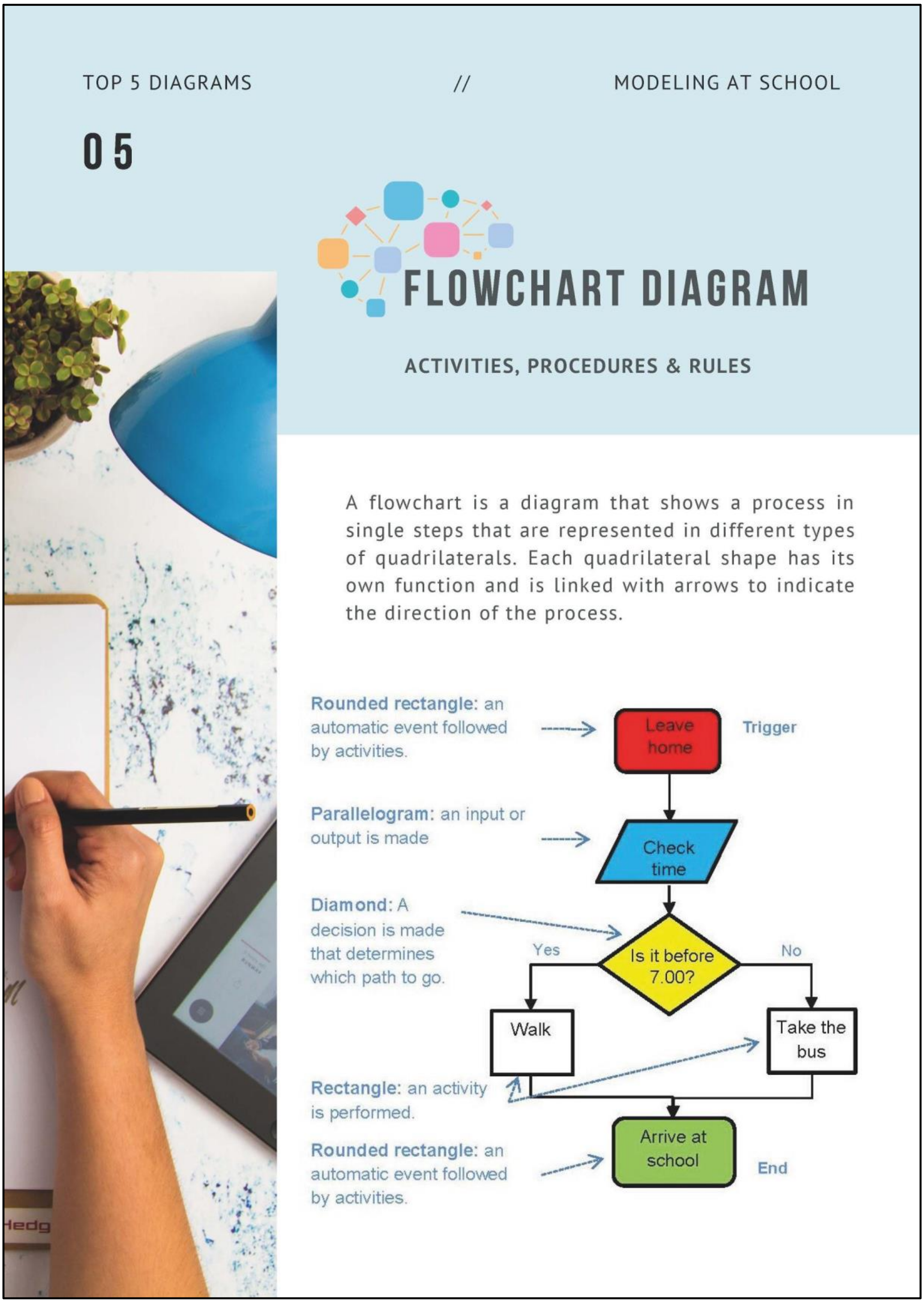


Figure 6: Diagram Guideline Page 6

Figure 7: Diagram Guideline Page 7

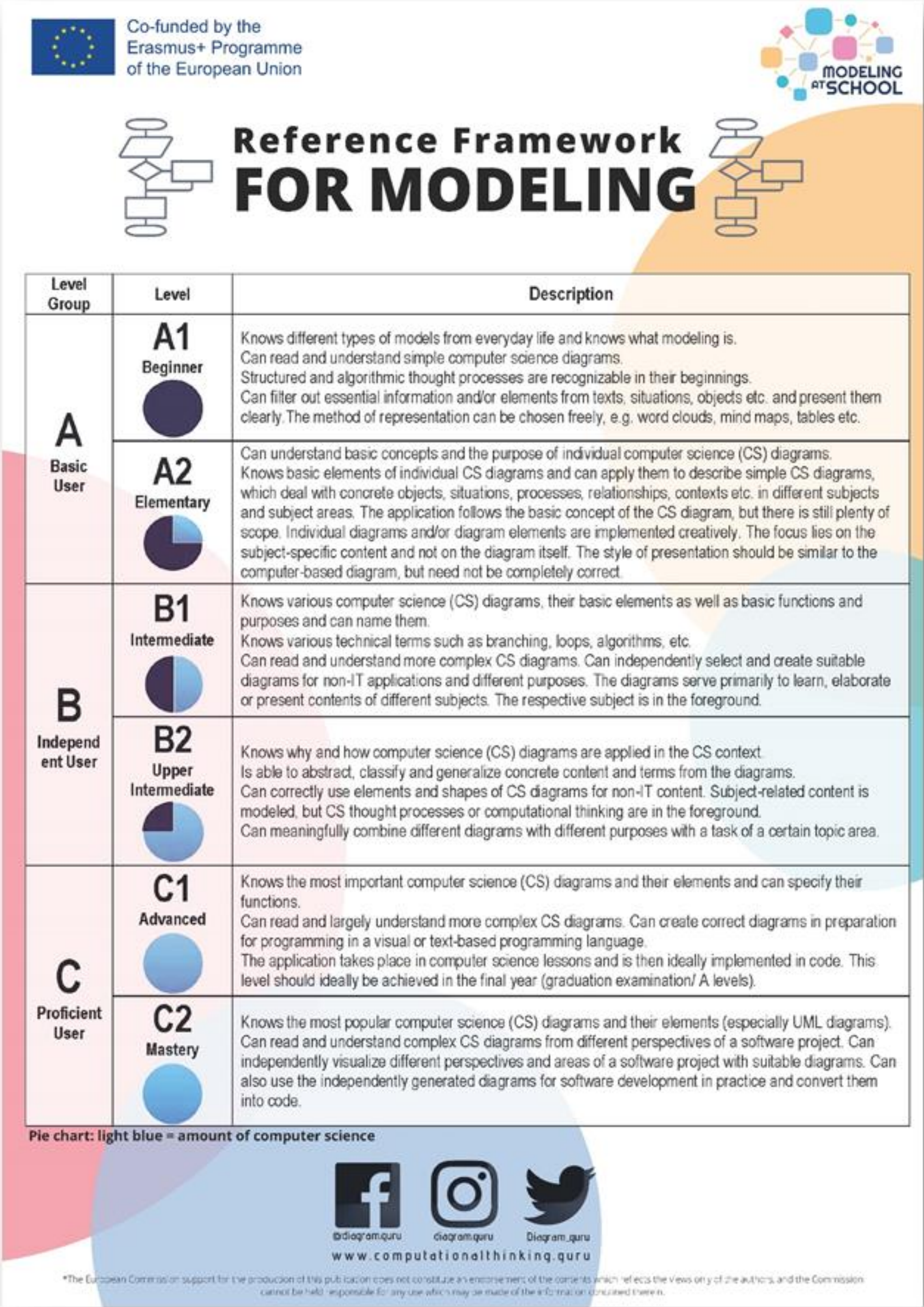
3 Reference Framework of Modeling

In the course of this project phase, a so-called *Framework of Reference for Modeling* (ReMo) has been developed [1]. The need to create such a concept was based on practical experience with schools, where teachers showed uncertainty in the rating of the diagrams. The question, if these diagrams follow a correct syntax based on computer science standards was present. Another critical aspect is, that from the perspective of computer science, diagrams created by people without CS background are non-compliant with the common standards. Using modeling in an interdisciplinary context goes along with blurry boundaries between random visualization of content and the use of computer science diagrams. The ReMo aims at eliminating uncertainties in applying modeling in different subjects. Currently, there is no practical assessment tool to evaluate the quality of diagrams in a non-informatic setting.

We see modeling as a language, hence, the Common European Framework of Reference for Languages (CEFR) [2] serves as basis for the development of the ReMo as Language of Computational Thinking and Computer Science. This framework shall classify and assess modeling competences starting from general competences such as mapping or visualizing knowledge (or visual literacy, the ability to discern meaning conveyed through images [3]) over computational thinking in everyday life situations up to modeling with UML in computer science.

The ReMo is divided into three main sections - A, B and C – which stand for 'Basic User', 'Independent User' and 'Proficient User'. These three main sections are again divided into two subcategories, which are marked with the numbers 1 and 2. The CEFR is a widely recognized standard in the language community across Europe and the reference to it facilitates the use of the modeling framework.

In each reference level (A1, A2, B1, B2, C1, C2) detailed descriptions of receptive and productive modeling skills are represented in form of "knows.." and "can do..." descriptors. Moreover, the modeling framework takes into account not only the syntax accuracy from a computer science perspective, but also sets a deep focus on the mental processes that occur while working with models. As visible in figure 8, in the framework, there are several pie charts in the second column. These pie charts should help the user to understand to which extend the individual diagrams follow the standard of computer science.



Co-funded by the Erasmus+ Programme of the European Union

MODELING AT SCHOOL

Reference Framework FOR MODELING

Level Group	Level	Description
A Basic User	A1 Beginner	Knows different types of models from everyday life and knows what modeling is. Can read and understand simple computer science diagrams. Structured and algorithmic thought processes are recognizable in their beginnings. Can filter out essential information and/or elements from texts, situations, objects etc. and present them clearly. The method of representation can be chosen freely, e.g. word clouds, mind maps, tables etc.
	A2 Elementary	Can understand basic concepts and the purpose of individual computer science (CS) diagrams. Knows basic elements of individual CS diagrams and can apply them to describe simple CS diagrams, which deal with concrete objects, situations, processes, relationships, contexts etc. in different subjects and subject areas. The application follows the basic concept of the CS diagram, but there is still plenty of scope. Individual diagrams and/or diagram elements are implemented creatively. The focus lies on the subject-specific content and not on the diagram itself. The style of presentation should be similar to the computer-based diagram, but need not be completely correct.
B Independent User	B1 Intermediate	Knows various computer science (CS) diagrams, their basic elements as well as basic functions and purposes and can name them. Knows various technical terms such as branching, loops, algorithms, etc. Can read and understand more complex CS diagrams. Can independently select and create suitable diagrams for non-IT applications and different purposes. The diagrams serve primarily to learn, elaborate or present contents of different subjects. The respective subject is in the foreground.
	B2 Upper Intermediate	Knows why and how computer science (CS) diagrams are applied in the CS context. Is able to abstract, classify and generalize concrete content and terms from the diagrams. Can correctly use elements and shapes of CS diagrams for non-IT content. Subject-related content is modeled, but CS thought processes or computational thinking are in the foreground. Can meaningfully combine different diagrams with different purposes with a task of a certain topic area.
C Proficient User	C1 Advanced	Knows the most important computer science (CS) diagrams and their elements and can specify their functions. Can read and largely understand more complex CS diagrams. Can create correct diagrams in preparation for programming in a visual or text-based programming language. The application takes place in computer science lessons and is then ideally implemented in code. This level should ideally be achieved in the final year (graduation examination/ A levels).
	C2 Mastery	Knows the most popular computer science (CS) diagrams and their elements (especially UML diagrams). Can read and understand complex CS diagrams from different perspectives of a software project. Can independently visualize different perspectives and areas of a software project with suitable diagrams. Can also use the independently generated diagrams for software development in practice and convert them into code.

Pie chart: light blue = amount of computer science

Figure 8: Reference Framework for Modeling

In the following section, the six levels of proficiency are described in greater detail:

1. **A1 - Beginner:** at the first level, the visualization process lies in the foreground. The A1 learner is familiar with the term “modeling” itself and is also able to read and understand basic diagrams from the field of computer science (CS diagrams). As it can be seen in the pie chart, the syntax correctness is not subject of attention at the beginner level. However, even though there is much scope for the diagram realization and a computer science diagram is not apparent at the first glance, the practitioners still need to apply the basic thinking skills of computer scientists called computational thinking to be able to perform the tasks. Specifically, already at the initial stage, skills such as problem-solving, decomposition, pattern-recognition, abstract thinking or algorithmic thinking are required. For example, the learner is required to filter out essential information, which then needs to be structured and sometimes also be transformed into an algorithm. This is possible through modeling with diagrams as it is practiced in computer science. Using basic forms of modeling as graphic organizers or tool for structuring and summarizing information, can be an effective learning strategy. [4] It is creative, innovative and trains computational thinking as well as problem solving – all of them competences defined in the framework for 21st century learning [5].

2. **A2 - Elementary:** at the second level, the practitioner can understand basic concepts and the purpose of individual computer science (CS) diagrams. Furthermore, he or she is also familiar with basic elements of these diagrams and is able to create simple CS diagrams which deal with concrete objects, situations, processes, relationships or contexts in different school subjects and subject areas. In other words, the application follows already the basic concepts of the CS diagrams, but there is still plenty of scope. The learner creatively applies individual CS diagrams and/or diagram elements but the focus still lies on the content and not on the diagram itself. Hence, the style of representation should be similar to the CS diagrams, but need not be completely correct.

3. **B1 - Intermediate:** as it can be seen in the pie chart of B1, the share of computer science and subject content is evenly distributed. This implies that the practitioner already knows various computer science diagrams, their basic elements as well as its basic functions and purposes and knows the exact designations. Furthermore, he/she is familiar with technical terms from the field of computer science, such as branching, loops or algorithms. At this stage, the practitioner is able to independently select and create suitable diagrams for a non- IT application and various purposes and can represent algorithms from everyday life or other subjects in form of a computer science diagram. Even though computer science is already strongly represented, the diagrams at the B1 level still primarily serve as a tool for learning,

elaborating or representing contents of different subjects. In short, the focus still lies on the respective subject.

4. **B2 - Upper Intermediate:** from this level on, the amount of computer science is predominant. Even though subject-related content is modeled, computer science thought processes or computational thinking are in the foreground. At a B1 level, the practitioner is familiar with many computer science diagrams, knows why and how they are applied in a computer science context and is able to abstract, classify and generalize concrete contents and terms out of the diagrams. Moreover, he or she can also correctly use elements and shapes of computer science diagrams for non-IT content. In contrary to the previous stages, the B2 level requires the practitioner to combine various computer science diagrams to pursue different learning purposes and to represent different views of a topic or subject area.

5. **C1 - Advanced:** in the advanced level, we move from an interdisciplinary approach to a technical/subject-specific implementation of modeling. In an educational setting this means, that modeling is applied in computer science lessons in preparation for programming in a visual or text-based programming language. At this stage, the practitioner knows the most important computer science diagrams and their elements and can specify their functions. Added to this, he or she can largely understand more complex computer science diagrams and is able to correctly create diagrams which are then transformed into code. Speaking of the Austrian educational system, these skills should ideally be gained at the stage of the A-levels.

6. **C2 - Mastery:** at the final stage of the reference framework for modeling the diagrams are used in software development projects and converted into code. The C2 practitioners know the common computer science diagrams and their elements, especially UML (unified modeling language) diagrams. He or she can read and understand complex computer science diagrams from different perspectives of a software project and is able to independently visualize different perspectives and areas of a software project with suitable diagrams.

The presented Reference Framework for Modeling is mainly designed for teachers of all subjects. It supports the integrative implementation of modeling as learning strategy and mapping tool in different subjects for teaching computational thinking. It gives an orientation, which receptive and productive skills users should have at which level of proficiency. Thus, it presents information about the extent of computer science contents at each level. Even persons without computer-science background can rate outputs (diagrams) designed by students regarding the fulfillment of some requirements of the basic digital education. The evaluation of this framework comprises internal (peer review process) and external (expert interviews and feedback from different target groups) elements. Relevant parameters are usability, comprehensibility, usefulness and practicability of the framework. Currently, the main evaluation focus is on the levels A and B, this is due to the fact that the majority of partner schools are lower secondary schools.

4 Link Reference Framework & Key Competences

For the purpose of the curricular and pedagogical contextualization of Ready-to-use-guideline for Teachers (IO2), we analysed the *Key competences for lifelong learning* recommended by the European Council (see: European Council, 2018). Our goal was to identify the essential knowledge, skills, and attitudes to be developed through modeling, both on the subject-based and transversal (beyond-the-subjects) level. In addition to the policy documents provided by the European Union, several other sources have been involved in our study, such as OECD's definition and selection of competencies (see: OECD), research literature on assessing and teaching 21-century- and transportable competences and skills (Lai & Vierling, 2012; Griffin & Care, 2015; Pellegrino & Hilton, 2012; Voogt & Roblin, 2012), furthermore documents and studies concerning the competence-based renewal of Finnish education (e.g. Lavonen & Korhonen, 2017; Miettinen, 2019). The lessons learned from the international policy and research context have been complemented by the pedagogical consultancy and recommendations, which have been offered for the teachers by the European Commission in the form of handbooks and popularization brochures (see e.g. Key competences for lifelong learning, 2019) and in dissemination events, such as international conferences organized by the European Commission to update the stakeholders in the educational sector on the latest information of the policy recommendations (see e.g. European Commission, 2019).

Based on our analysis, the support in the pedagogical framing of key competence development has a critical role for several teachers being involved in our project, while they will search for ways to introduce the novel modeling-based approaches in their classes. In order to prepare for the change, we also reviewed the pedagogical approaches and methodologies suggested by several sources from the above-described field, in order to provide practical support for teachers and students in their pragmatic application of modeling both in subject- and competence-based context. These results are already pointing towards IO3 and IO4, however already influenced our contribution to the Ready-to-use-guideline for Teachers.

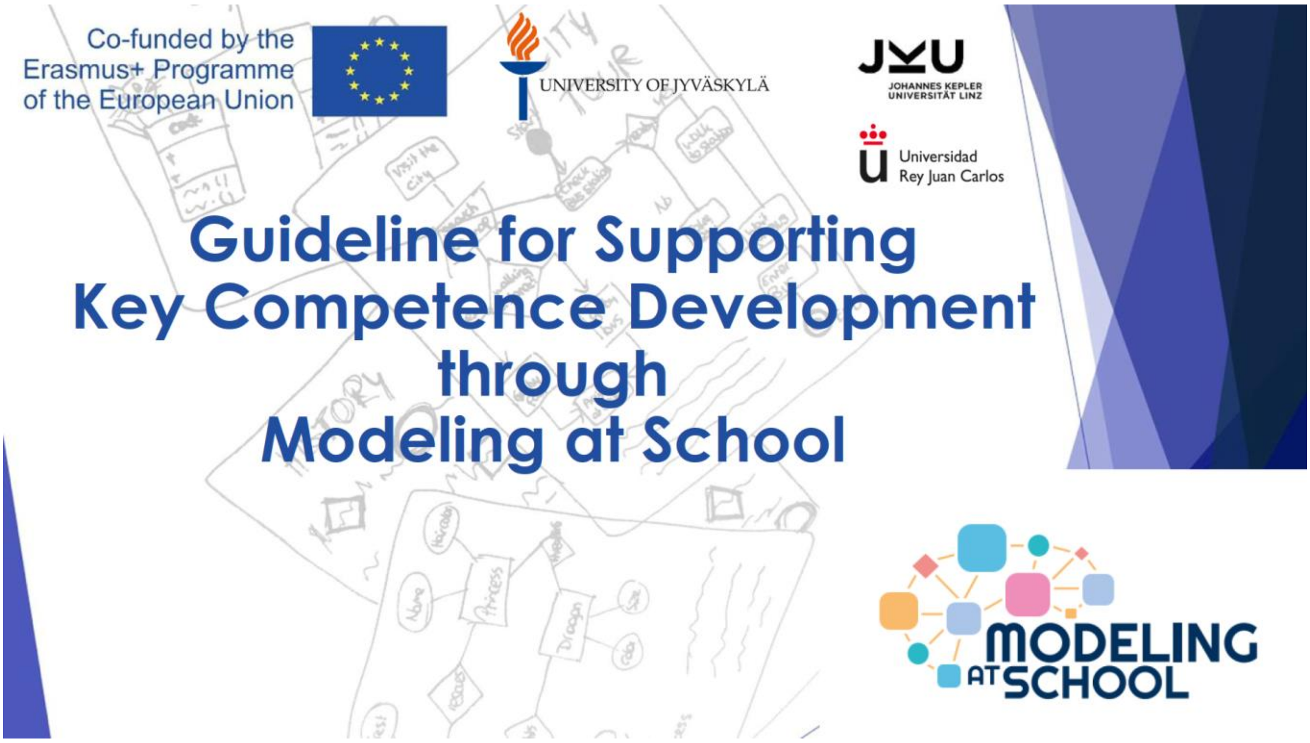


Figure 9: Opening slide from the Guideline for Supporting Key Competence Development through Modeling at School

The Table 1 below provides examples for those fields of knowledge, skills, and attitudes related to each key competences which could be effectively supported by modeling-based approaches. The Table 1 also refers to certain school subjects, where the modeling-based development of a given competence may be particularly relevant. As there are country-based differences in subjects, we have followed the subject structure of the Finnish National Core Curriculum.

Key competences for lifelong learning	**Essential knowledge, skills, and attitudes related to key competences to be developed through modeling (examples)**	**Particularly relevant subject-based examples for elementary school and for secondary school** (note: there are country-based differences in subjects)
Literacy competence	- Searching, identifying, understanding, distinguishing, processing, collecting, expressing, creating/formulating, assessing and interpreting information in an appropriate, *critical* and *creative* way.	Mother tongue Second language / foreign language
Multilingual competence	- Supporting language learning (listening, speaking, reading and writing) and understanding of grammar. - Understanding, expressing, interpreting concepts, thoughts, feelings, facts, and opinions. - Using different languages appropriately and effectively in different contexts.	Second language/ foreign language
Mathematical competence and	- Developing and applying mathematical thinking and insight for *problem solving*	Mathematics

	- Mastering processes, activities and knowledge related to numeracy (numbers, measures, operations, terms and concepts) - Ability and willingness to use and apply mathematical modes of thought, reasoning, argumentation, and presentations/communication (formulas, **models**, constructs, graphs, charts, statistical data) - Understanding mathematical aspects of digitalisation	
…competence in science, technology and engineering	- Knowledge of scientific concepts, theories, principles and methods, technology and technological products and processes - using logical and rational thought and scientific inquiry, and communicate and explain the evidence-based conclusions and reasoning behind them - understanding of science as a process for the investigation through specific methodologies, including observations and controlled experiments - verifying a hypothesis and the readiness to discard one's own convictions when they contradict new experimental findings - using and handle technological tools and machines - developing a concern for ethical issues and support for both safety and environmental sustainability, in particular as regards scientific and technological progress in relation to oneself, family, community, and global issues - understanding changes caused by human activity and responsibility as an individual citizen	Environmental studies (primary school) Physics Chemistry Biology (Geography)
Digital competence	- Using digital technologies in a confident and responsible way knowing their basic functions and use - Recognising and effectively engaging with software, devices, artificial intelligence and robots - Information, data, and media literacy - Using, accessing, filtering, evaluating, creating, programming and creating and sharing digital content - *Problem solving* and *critical thinking* - Supporting communication, collaboration, *creativity* and innovation - Understanding the general principles, mechanisms and logic underlying evolving digital technologies as well as safety (incl. digital well-being and competences related to cybersecurity), and intellectual property-related questions.	ICT as a subject Robotics
Personal, social and learning to learn competence	- Managing time and information - Managing and organising one's own learning and learning strategies - Dealing with complexity and having *problem-solving* attitude - reflecting critically and making decisions - learning and working both collaboratively and autonomously - organising and persevering with one's learning, evaluating and sharing it - seeking support when appropriate and effectively managing one's career and social interactions	Health education (in Finland) Home economics (Finland) Religion Ethics
Citizenship competence	- Understanding of history and social, political, socioeconomic, demographic, climate, and cultural issues - Sustainable development and democracy - *Critical thinking*, *problem solving*, participation, argument development - Accessing and interacting with traditional and new forms of media	History Social studies Politics Law (Geography) Religion

	- Intercultural communication	Ethics
Entrepreneurship competence	- Acting upon opportunities and ideas and transforming them into values for others - *Creativity*, *critical thinking* and *problem solving* - Strategic thinking - Planning and managing projects including both processes and resources - Making informed decisions including financial decisions relating to cost and value	Economics Guidance counselling/tutoring
Cultural awareness and expression competence	- Expressing and interpreting figurative and abstract ideas, experiences, emotions and meaning *creatively* and with empathy - Knowledge of cultures and traditions	Visual arts Crafts History Social studies Music Physical education Religion Ethics

Table 1: Essential knowledge, skills and attitudes related to key competences can be studied through modeling.

Note: Skills such as critical thinking, problem-solving, teamwork, communication, creativity, negotiation, analytical and intercultural skills are embedded throughout the key competences.

Based on the summary above, we linked the introduction of the development of each key competences with basic modeling diagram types and the following Guideline for Supporting Key Competence Development through Modeling at School was created:

https://drive.google.com/file/d/1KxTTG4DhAYVOEJSjyN8IttuFuiSAjzhX/view?usp=sharing

For instance, entity-relation diagram could be used for structuring the storyline in a story-telling/-writing activity to develop Literacy competence (see Figure 10).

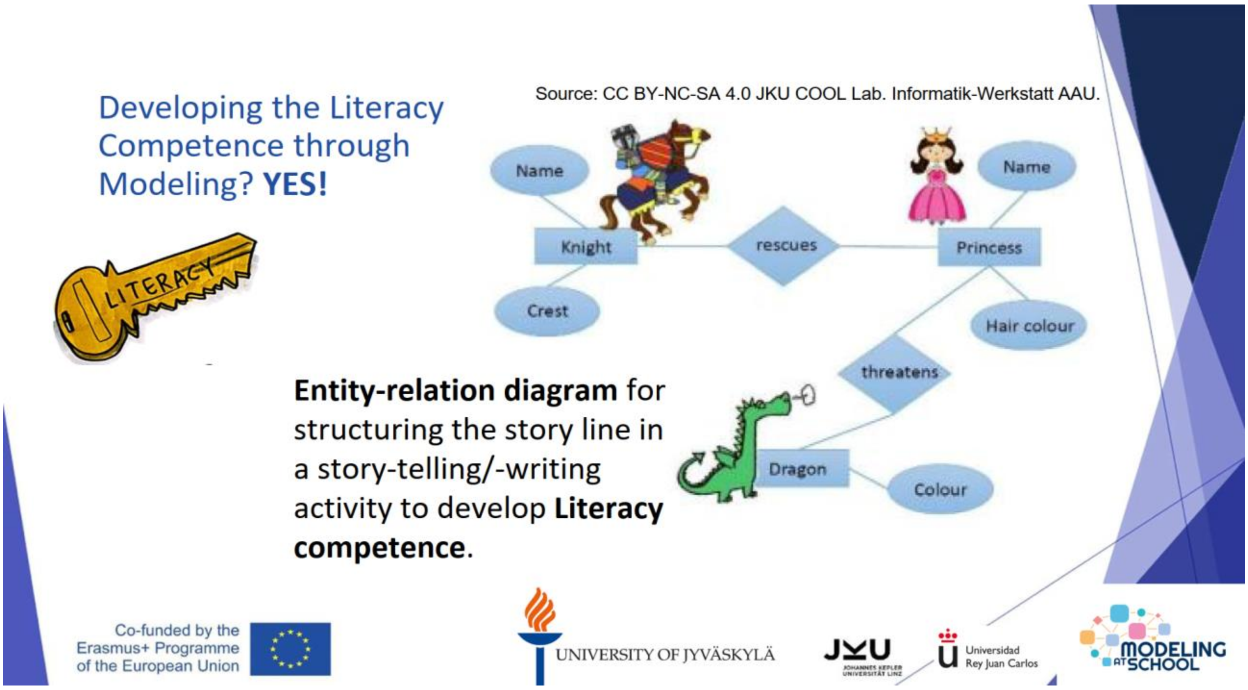


Figure 10: Use of entity-relationship diagram to support literacy competence. (Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU)

The following flowchart (Figure 11) demonstrates how modeling can be applied when developing Multilingual competence and particularly language learning and understanding of grammar, in this case, understanding rules for comparatives and superlatives.

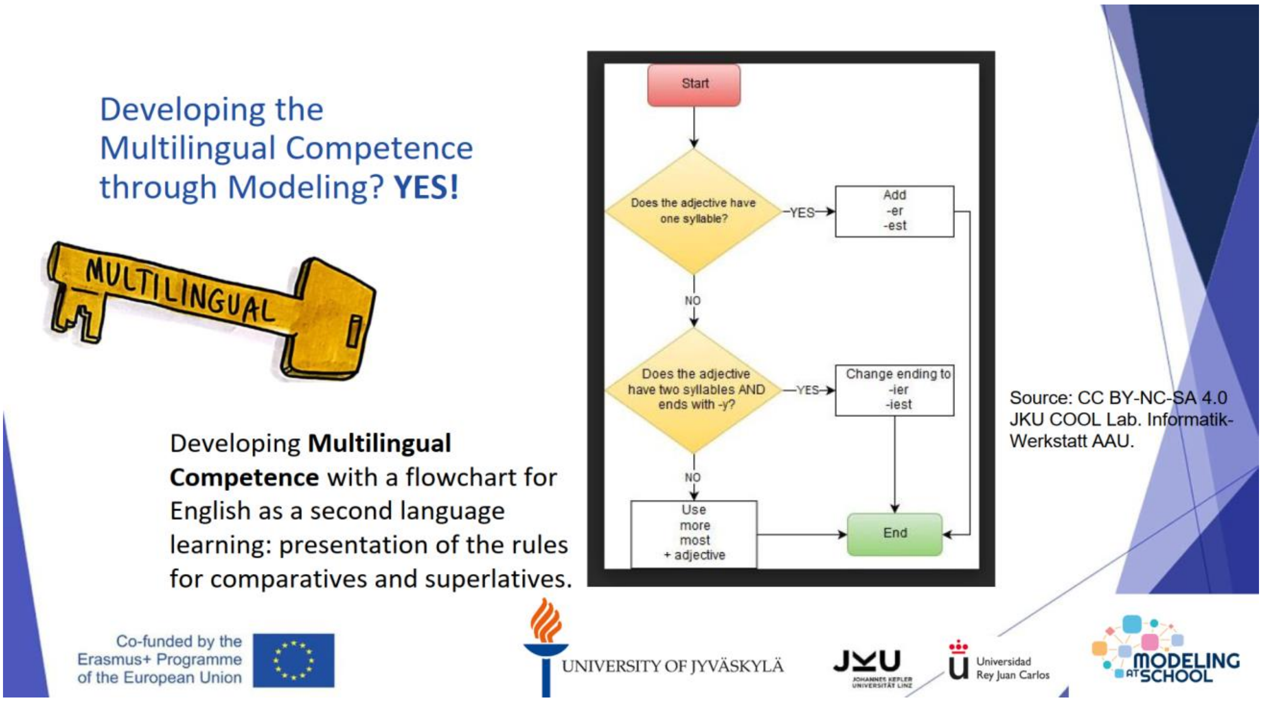


Figure 11: Use of flowchart to support multilingual competence. (Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU)

In the example below (Figure 12), scientific competence is developed by categorizing with class diagram-based modeling method the sediments, which are described in an Austrian Biology textbook.

Figure 12: Use of class diagram to support scientific competence. (Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU).

Digital competence development can offer many opportunities to build on the already familiar concepts of creating mind maps, and start to experiment with programming flowcharts to represent organized thinking about processes (Figure 13).

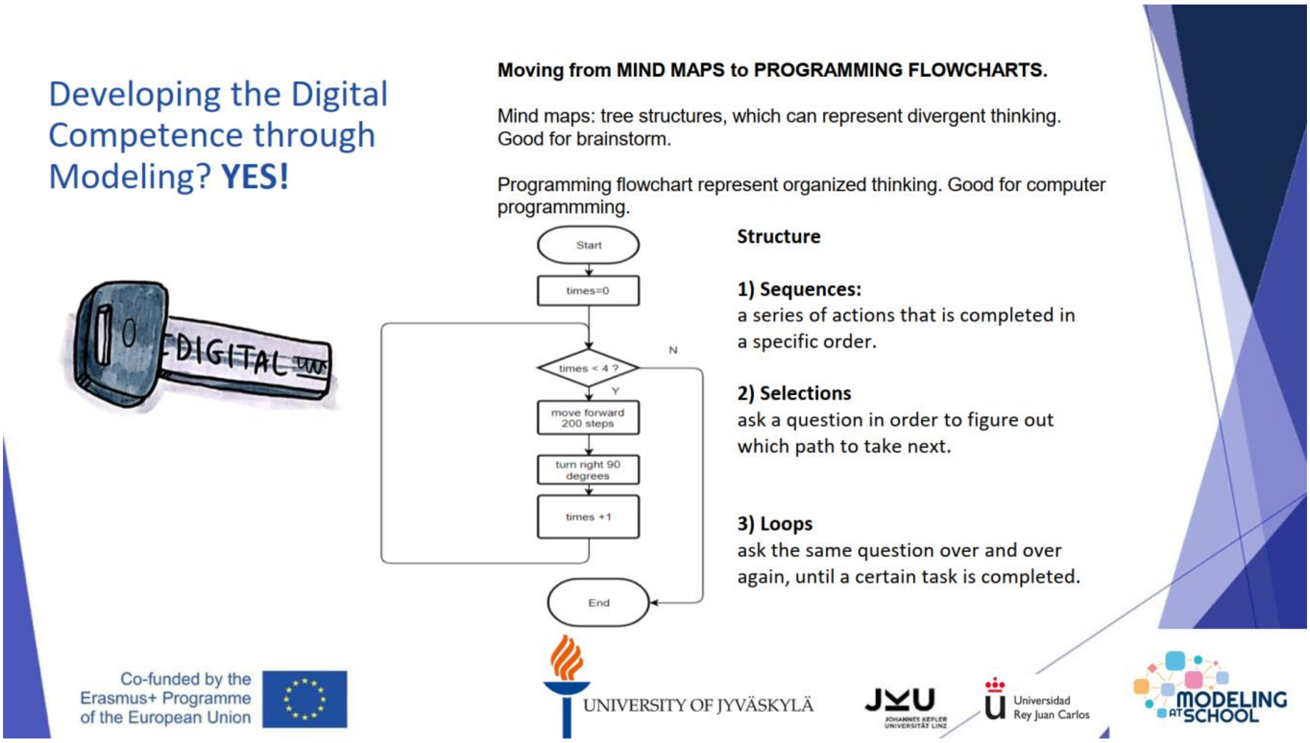


Figure 13: Use of programming flowchart to develop digital competence.
(Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU).

Complex competences, like personal, social and learning to learn competence can be challenging to develop in a clearly subject-based context. However everyday necessary knowledge, like the protocol for providing first-aid can be efficiently modeled by a flowchart (Figure 14).

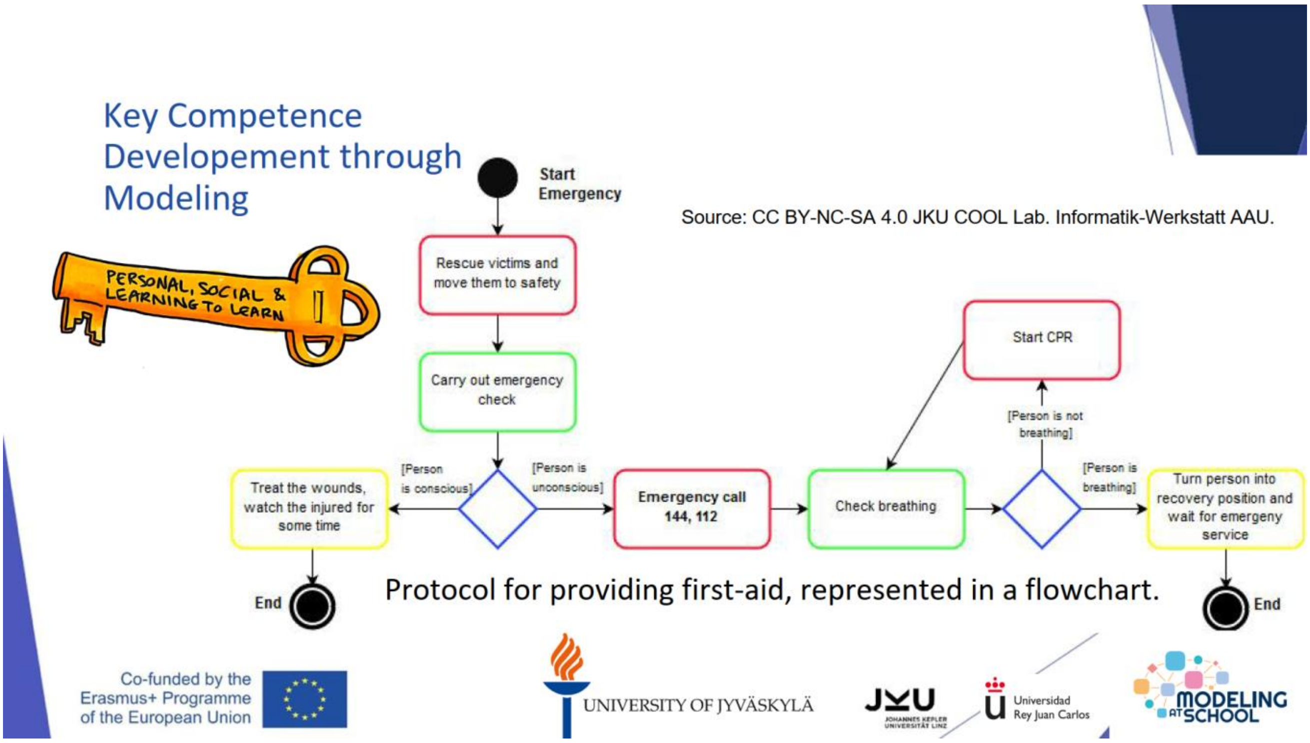


Figure 14: Use of flowchart to develop personal, social and learning to learn competence. (Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU).

Decision-making processes, especially if these are connected to complex social organizations like a whole country, are not easy to comprehend. On the other hand, modeling can be an appropriate tool to create a visual summary of the comprehensive mechanism (Figure 15).

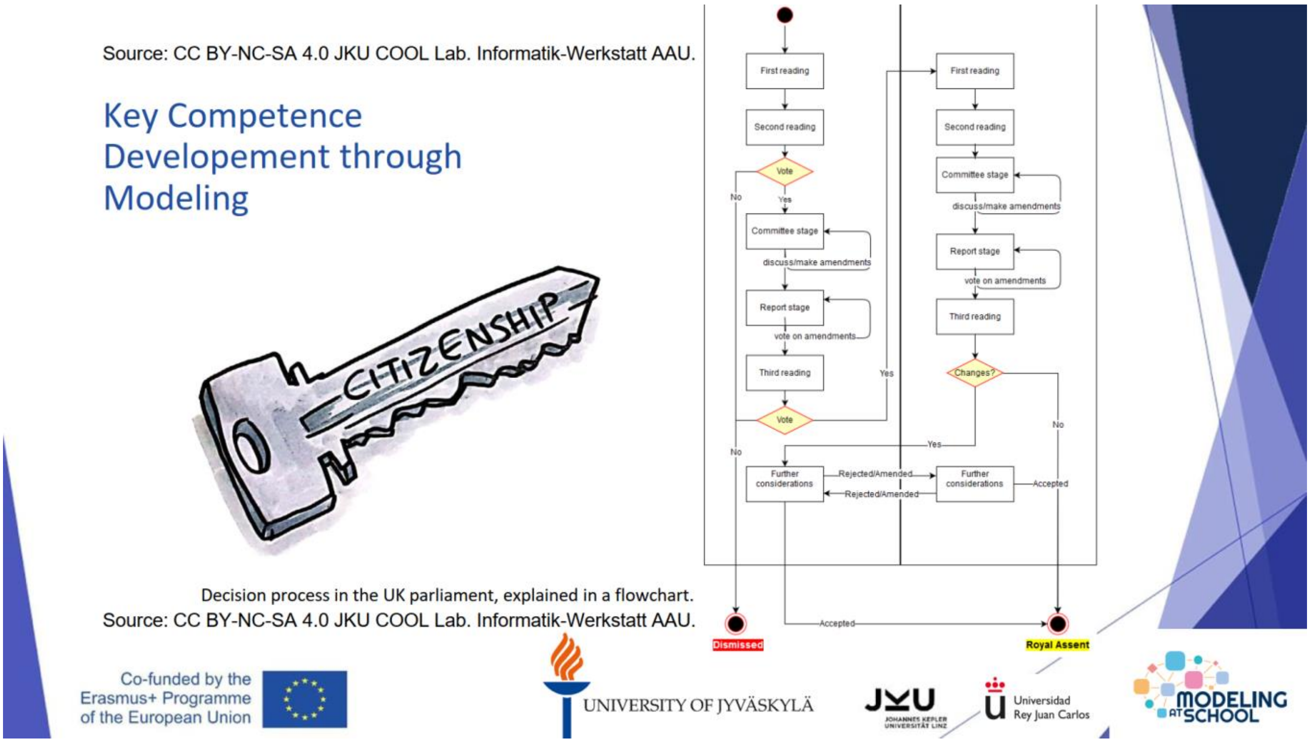


Figure 15: Use of flowchart to develop citizenship competence.
(Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU).

The development of entrepreneurship competence might offer several opportunities to implement effectively various modeling tools. The use of a simple “anchor diagram” as illustrated in the example below (Figure 16), summarize the results of a classroom discussion on what is generally money is used for in people’s life.

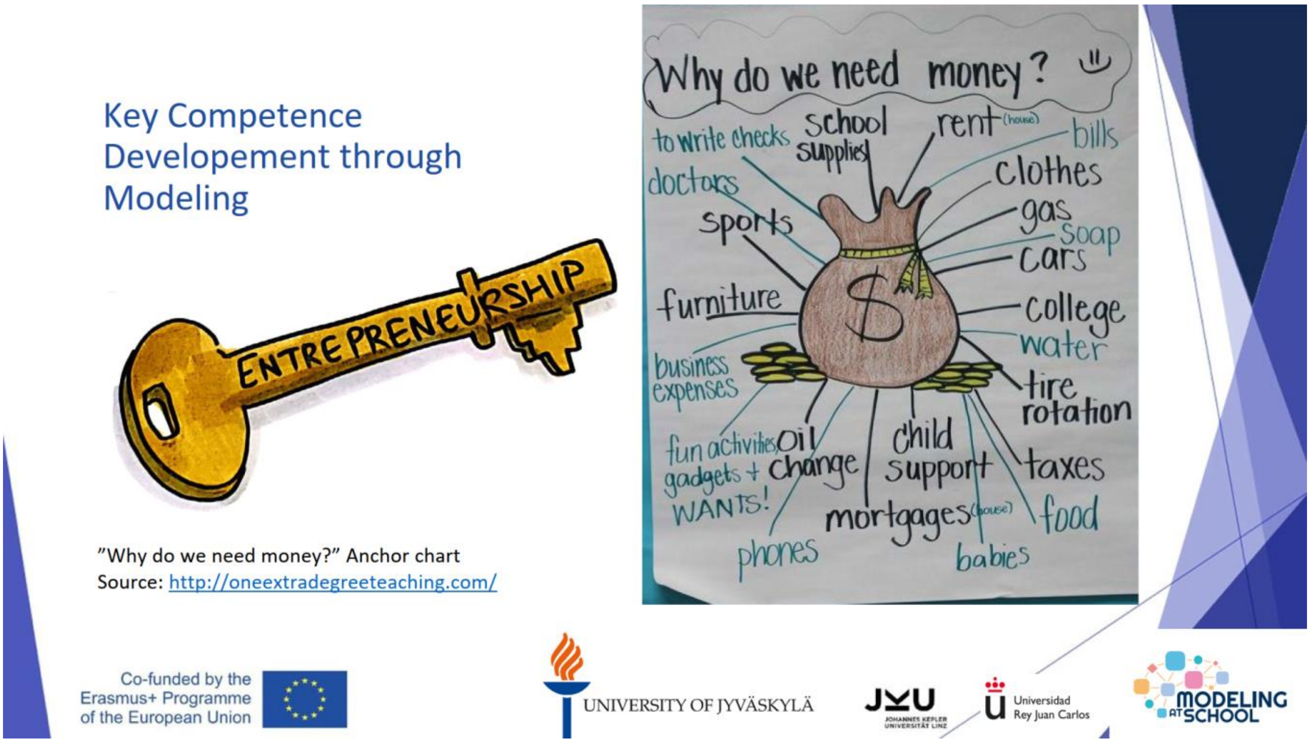


Figure 16: Use of an "anchor chart" to develop entrepreneurial competence. (Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU).

Family trees are well-known and widely used diagrams to visualize family relationships. At the same time, these can be implemented to develop knowledge, skills and attitudes related to the competence of cultural awareness and expression (Figure 17).

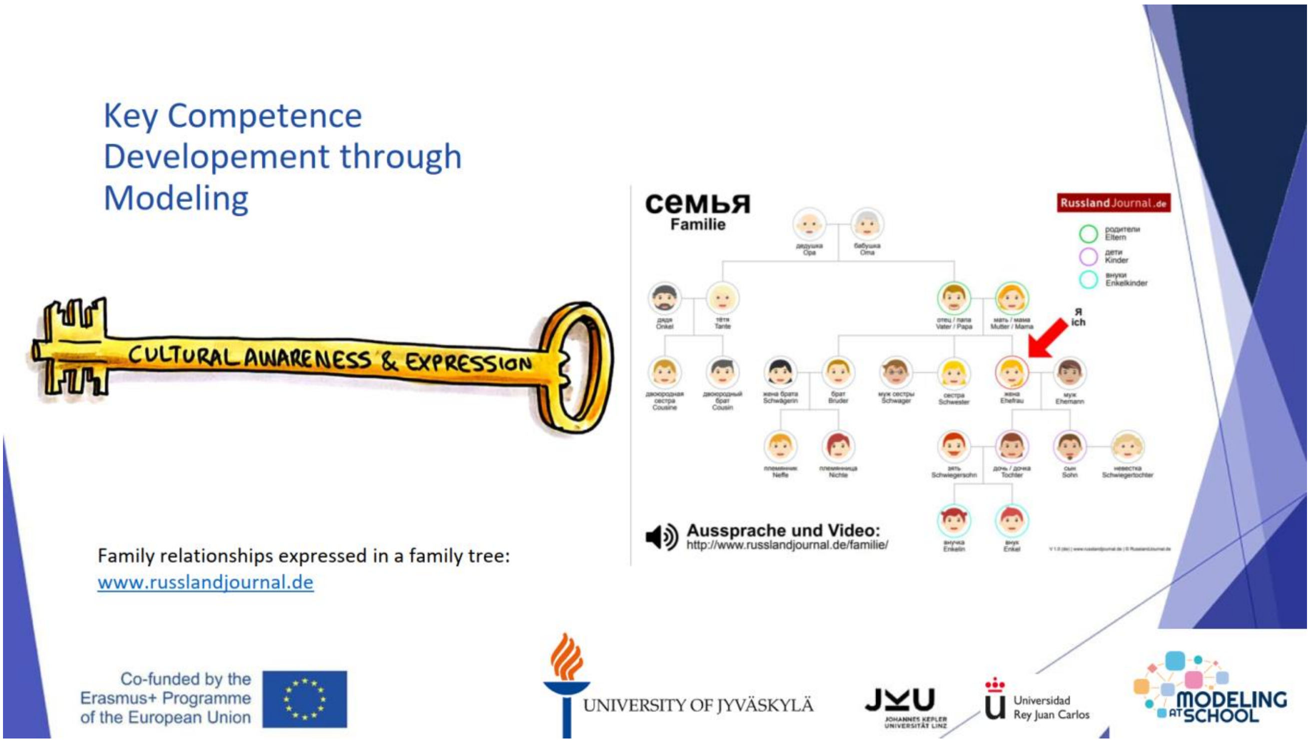


Figure 17: Using family tree to develop cultural awareness and expression as part of learning second language learning. (Source: CC BY-NC-SA 4.0 JKU COOL Lab. Informatik-Werkstatt AAU).

According to the pedagogical recommendations of key competence development (see: European Commission, 2019), there are several ways to support the introduction of the recommended competency areas in the classroom both in subject-based and multidisciplinary projects. These approaches can be underlined or extended by various modeling-based activities as well, in order to provide a deeper understanding, or engagement and motivation. For example, if formative assessment and self-assessment, self-reflection and quality feedback can be brought to the next level by modeling-related activities, the active role and a sense of autonomy of the learner in the creation of their learning journey may be increased through the participation in the modeling activity. A combination of individual and collaborative learning opportunities, understanding of different ideas, perspectives, and diversity, differentiation as part of modeling-projects may support both individual and collaborative learning processes. Implementing Real World Scenarios and following a cyclical process of design, creation, reflection, and adaptation in the modeling activities can be very effective to bring forward inquiry and project-based learning in a cross-disciplinary manner. Learning, thinking, reflecting, problem-solving in action as the core of the modeling activities, or the aesthetic design of flowcharts, or the creative presentation of the modeling-based approaches as part of the learning projects, or developing imagination and inventing playful learning opportunities through modeling may bring forward experiential learning in the learning process. As modeling can be an excellent tool for considering the language dimension in literacy development, or in foreign language learning, just like as in subject teaching, or in acknowledgment of other languages brought in the learning process by pupils, modeling can contribute to the development of language-aware school environment in the spirit of developing multilingual competences. Modeling may offer very effective approaches to understand how the internet and social media works, modeling-based activities can also involve using various digital technologies in an optimal way, or link certain school activities to European initiatives, like the EU Code Week. Modeling may be also useful in finding new ways to reflect on and strengthen inclusive settings in the learning process. Group modeling activities may encourage peer coaching, support a wide range of extra-curricular activities, career guidance and strengthen personal skills and competences. This way modeling may be a useful tool for supporting wellbeing, including social and emotional development in the school.

5 Glossary

As stated in prior sections, this document aims to provide proper guidelines with essential information about the diagrams proposed, their purposes and elements, and above all, how to put into practice. To complement previous information and as cross-sectional reference, this section aims to cover the majority of concepts needed for the guidelines proposed in this document. This glossary of terms that will serve as base for facilitating the understanding of the guidelines target end-users (teachers, instructors, pupils) and is characterized by being general, supporting the comprehension of these guidelines with regardless of CS background. We have divided this glossary into four categories listed below. These categories summarize: (1) Computational Thinking/Computer Science concepts, (2) techniques associated with CS/CT, (3) CS diagrams and finally (4) CS/Modeling elements.

For instance, Table 2 sum ups the main concepts/skills related to CT and CS. These concepts are the pillar for developing the CT ability.

CT/CS concepts	Abstraction	Ability to create something simple from something complicated/complex by leaving out the irrelevant details and finding the relevant patterns.
	Generalization	Ability to transfer a problem-solving process to a wide variety of problems, i.e. solving new problems based on previous experience. It generally involves identifying and exploiting patterns.
	Decomposition	Ability to break down problems into smaller parts that may be more easily solved. These parts can be understood, solved, developed and evaluated separately.
	Algorithmic thinking	Ability to think in terms of sequences and rules as a way of solving problems.

	Creative thinking	Ability to generate lists of new and unique ideas through a fresh perspective and sometimes unconventional solution to solve a problem.
	Critical thinking	Ability to develop independent and critical thought, enabling the analysis of ideas, knowledge and process linked to our personal value system and judgment.
	Communication	Ability to articulate thoughts and ideas in a variety of forms, communicate for a range of purposes and in diverse environments.
	Collaboration	Ability to work effectively and respectfully with diverse teams, exercise flexibility and willingness to accomplish goals and assume shared responsibility.
	ICT literacy	Ability to use technology to develop 21st century content knowledge and skills, in the context of learning core subjects.

Table 2: Definition of concepts related to Computational Thinking and Computer Science.

Secondly, a collection of techniques used to develop CT/CS are shown in Table 3. These techniques associated with CT are tools/strategies by which CT is put into practice [6]. Note that these techniques are closely related to the concepts aforementioned.

Techniques to develop CT/CS	Coding	Coding is an essential element of the development of any computer system that translates the design into code form and evaluating it to ensure that it functions correctly under all conditions.
	Debugging	Debugging is the process of analysis and evaluation using skills such testing, tracing, and logical thinking to predict and verify outcomes. Broadly speaking, it can be defined as finding and fixing errors from a reasonable approach.
	Evaluation	Evaluation is the process of ensuring a solution is appropriate, covering the initial requirements imposed on the problem. This process involves the assessment of: (1) the solution fits for purpose and (2) the solution does the right thing.
	Modularity	Developing autonomous processes that encapsulate a set of often used commands performing a specific function.
	Data representation	Summarize and represent patterns using data structures such as array, linked list, stack, queue, graph, hash, etc.
	Testing and verification	Debug a program, writing tests and formal verification.
	Algorithm design	Creating clear, finite and ordered instructions (or recipes) for solving a problem.

	Parallelization	It is a processing control structure in which several paths of control can be active at the same time.
	Synchronization	Mechanism used to guarantee harmony and control paths to communicate.

Table 3: Techniques used to develop concepts/skills of CT/CS.

Continuing the analysis, definitions of terms associated with Diagrams are presented. Table 4 sum ups main diagrams used in the activities carried out by authors and which were chosen for this document.

CS diagrams	UML	Unified Modeling Language (UML). UML diagrams illustrate the quantifiable aspects of a system that can be described visually, such as relationships, behavior, structure, and functionality.
	Entity-relationship	An Entity Relationship Diagram (ERD) is a visual representation of different entities within a system and how they relate to each other
	Sequence	Sequence diagrams model the flow of logic within your system in a visual manner, enabling you both to document and validate your logic, and are commonly used for both analysis and design purposes.
	Use-Case	Use case diagrams give a graphic overview of the actors involved in a system, different functions needed by those actors and how these different functions interact.
	Activity	Activity diagrams represent workflows in a graphical way. They can be used to describe the business workflow or the operational workflow of any component in a system. Sometimes activity diagrams are used as an alternative to State machine diagrams.
	Class	Class diagrams are the main building block of any object-oriented solution. It shows the classes in a system, attributes, and operations of each class and the relationship between each class. In most modeling tools, a class has three parts. Name at the top, attributes in the middle and operations or methods at the bottom. In a large system with many related classes, classes are grouped together to create class diagrams. Different relationships between classes are shown by different types of arrows.

Table 4: Definitions of CS/Informatics diagrams

Computer Science/Modeling elements	Branch	A branch is an instruction in a computer program that can cause a computer to begin executing a different instruction sequence and thus deviate from its default behavior of executing instructions in order
	Branching	Branching statements allow the flow of execution to jump to a different part of the program. The common branching statements used within other control structures include: break, continue, return, and goto.

	Loops	A loop in a computer program is an instruction that repeats until a specified condition is reached.
	While Loop	It is an entry-controlled loop. In while loop, a condition is evaluated before processing a body of the loop. If a condition is true then and only then the body of a loop is executed. After the body of a loop is executed then control again goes back at the beginning, and the condition is checked if it is true, the same process is executed until the condition becomes false. Once the condition becomes false, the control goes out of the loop.
	Do-While Loop	A do-while loop is similar to the while loop except that the condition is always executed after the body of a loop. It is also called an exit-controlled loop.
	For Loop	The initial value of the for loop is performed only once. The condition is a Boolean expression that tests and compares the counter to a fixed value after each iteration, stopping the for loop when false is returned. The incrementation/decrementation increases (or decreases) the counter by a set value.
	Break statement	The break statement is used mainly in in the switch statement. It is also useful for immediately stopping a loop.
	Continue statement	When you want to skip to the next iteration but remain in the loop, you should use the continue statement.
	Decision control	Depending on whether a condition is True or False, the decision control structure may skip the execution of an entire block of instructions or even execute one block of instructions instead of another.
	Loop control	Control structure that allows the execution of a block of instructions multiple times until a condition statement is encountered.
	Sequence	Sequence is a set of instructions executed one after another.
	Subroutine/procedure/method	Grouping and naming a collection of instructions that do a well-defined (unique) task.
	Iteration	Iteration is the process of repeating a set of instructions until a certain condition is met.
	Sequence	Formulating instructions to be followed in a given order.
	Variable	It is a zone of memory that is assigned a name and has a concrete value that can be changed.
	Patterns	A reliable set of traits or observable characteristics extracted in high-level abstraction.

	Algorithms	Simple commands to follow for solving a problem.
	Data collection	Find a data source for a problem area
	Creativity	Trying new approaches to get things done equals innovation and invention.
	Attribute	Structural characteristics of a class.
	Operation	Behaviour of a class.
	Association	Assign attributes to the relationship between classes.
	Aggregation	Special form of association used to indicate that objects are assembled together to create a new object.

Table 5: Definitions of Computer Science/Modeling elements.

6 Ready-to-use guidelines (Country specific)

6.1 Austria

As already presented in chapter 2, the general information about the top five diagrams is available in a printed version as (1) a first introduction to diagrams that are proofed to be very suitable for an interdisciplinary use and as (2) hands-on material for the different stakeholders. However, besides information about the diagrams, there is a need to provide background information about modeling as a teaching and learning strategy, how to implement it, modeling tools, etc. For this information and more detailed descriptions of several diagrams, we decided to keep abreast of the times by using digital tools and so we created an online ready-to-use guideline in forms of YouTube tutorials.

The idea of the Erasmus + project Modeling at School is to bring modeling as a teaching and learning strategy in schools. From our many years of experience working with schools, we know that it is difficult for teachers to take part in further training. On the one hand, this is due to time resources, on the other hand, it is due to the distance of the conference location from the workplace. The Austrian project partner schools are spread all over Austria. Furthermore, we have in mind that the outputs of the project should be made available to as many schools as possible with little effort and in a sustainable manner. Another reason why an online guideline seemed to be very suitable was the idea of an online training based on the flipped classroom method. The concept consists of various parts such as the online-ready-to-use guideline, workshops, personal advice, etc.. A detailed description of this concept will follow in the report of O3.

The first feedbacks of our partner schools to the online ready-to-use-guideline was very positive. The participants rated both the type of presentation and the content as positive. In the opinion of the multipliers interviewed, all relevant topics were dealt with in the guideline, so that the participants feel able to implement modeling in their lessons. With regard to the type of presentation (as videos), the appealing and compact presentation was particularly emphasized.

Figure 18: YouTube Channel

The modeling video guideline can be accessed via YouTube and is also available in English as printable version.

Link to YouTube channel:

https://www.youtube.com/watch?v=Fl22elWaw3o&list=PL9XUYlvvknBzXlKu0QenNOzofUkRN4fW-

Link to printable English guideline:

https://drive.google.com/file/d/1b8u80u4Jghoje_9su4ndGJQhH-EeldgT/view?usp=sharing

Based on the W-questions, the six short introductory videos should provide all the essential background information that is needed to get started with modeling:

1. **W**hat is modeling?
2. **W**hy? Modeling as a teaching and learning strategy.
3. **W**hat do I need modeling for?
4. Ho**W** does modeling work? Part 1
5. Ho**W** does modeling work? Part 2
6. **W**hat can I use to model? Useful tools to put it into practice.

For a first introduction to modeling, we decided to choose three diagrams for the videos, one diagram for each category of the table, that had been developed in Intellectual Output 1:

Structures & Categories	Rules & Procedures	Situations & States
+ **class & object diagrams** + trees	+ **activity diagrams** + flow charts + graphs	+ use case diagrams + **entity relationship diagrams** + state diagrams

Table 6: Diagram Overview

6.1.1. What is modeling?

In the first video, we explain what is meant by modeling and show the difference between mental, graphic, physical and Computer Science models.

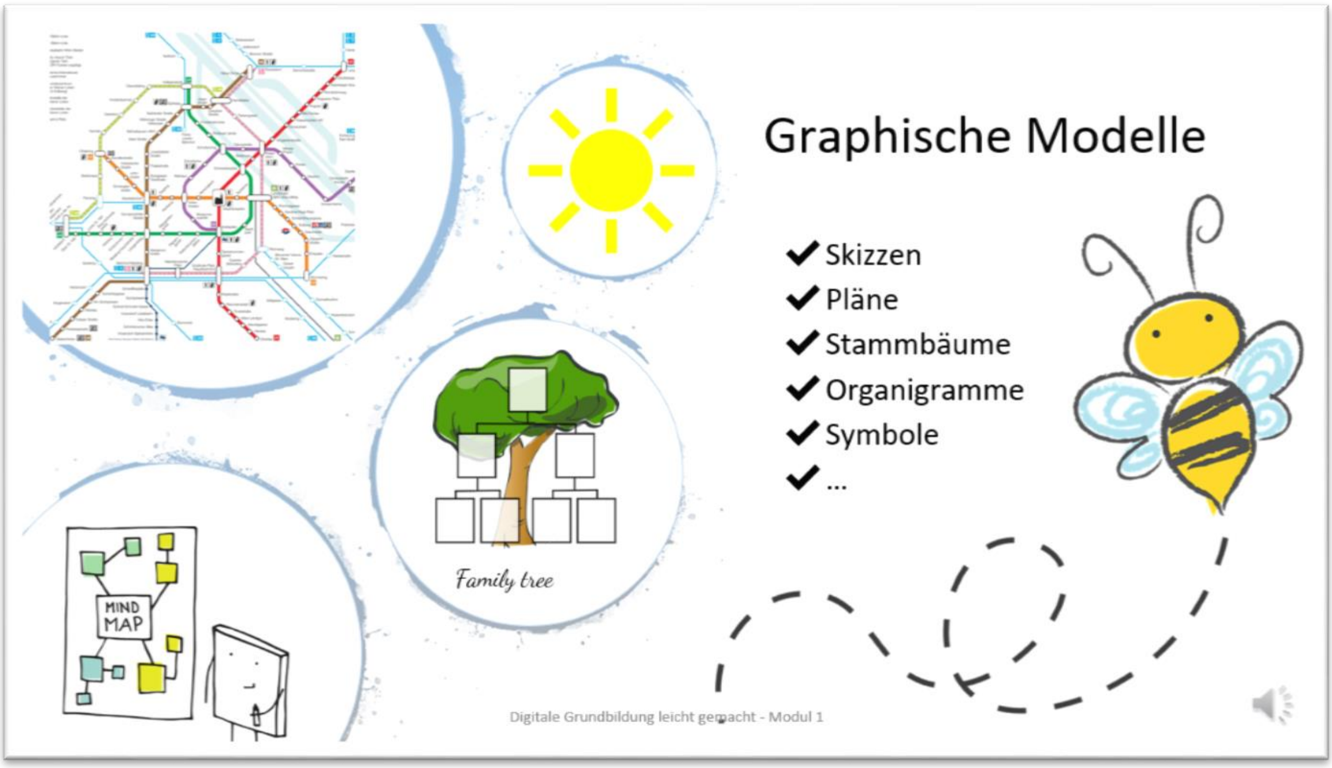


Figure 19: What is modeling?

6.1.2. Why? Modeling as a teaching and learning strategy

The second video shows why modeling should be used as a teaching and learning strategy. There are three good reasons for this: firstly, modeling is part of the Austrian curriculum for all subjects. Secondly, modeling is a brain-friendly teaching and learning strategy and thirdly, also promotes computational thinking skills, which is part of the new curriculum “basic digital education”. Furthermore, learning goals and arguments for modeling are presented.

Figure 20: Why modeling?

6.1.3. What do I need modeling for?

The third video further describes the three areas of application which have been developed in Intellectual Output one (see figure 21):

- Terms, Structures & Categories
- Activities, Rules & Procedures
- Situation, States and Relations

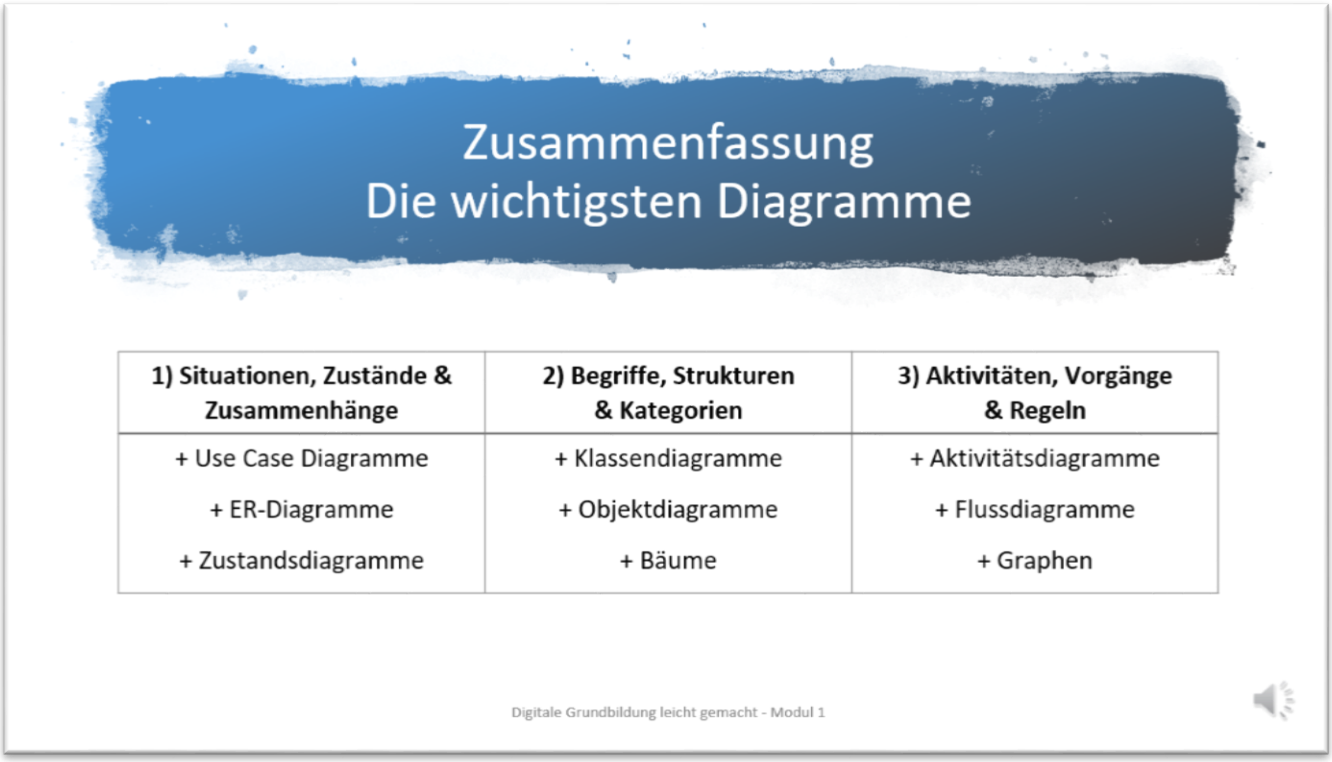


Figure 21: What do I need modeling for?

6.1.4. How does modeling work? Part 1

This video shows a step-by-step guide on how to use modeling in practice. The first step of modeling is the preparation phase. Here it is important to define the teaching and learning goals and the task of a certain subject:

- Do I want to present an overview and relations?
- Do I want to structure and expand knowledge?
- Do I want to describe situations or states?
- Do I want to represent and describe rules and procedures?

The second step is to get an overview of the task/text and capture the essential information by highlighting keywords, creating a table etc. The third step is to create a draft, a blueprint of a model. After that, the Computer Science diagram can be created by adding abstraction, generalization etc. In a more advanced stage, models can be further developed and combined.

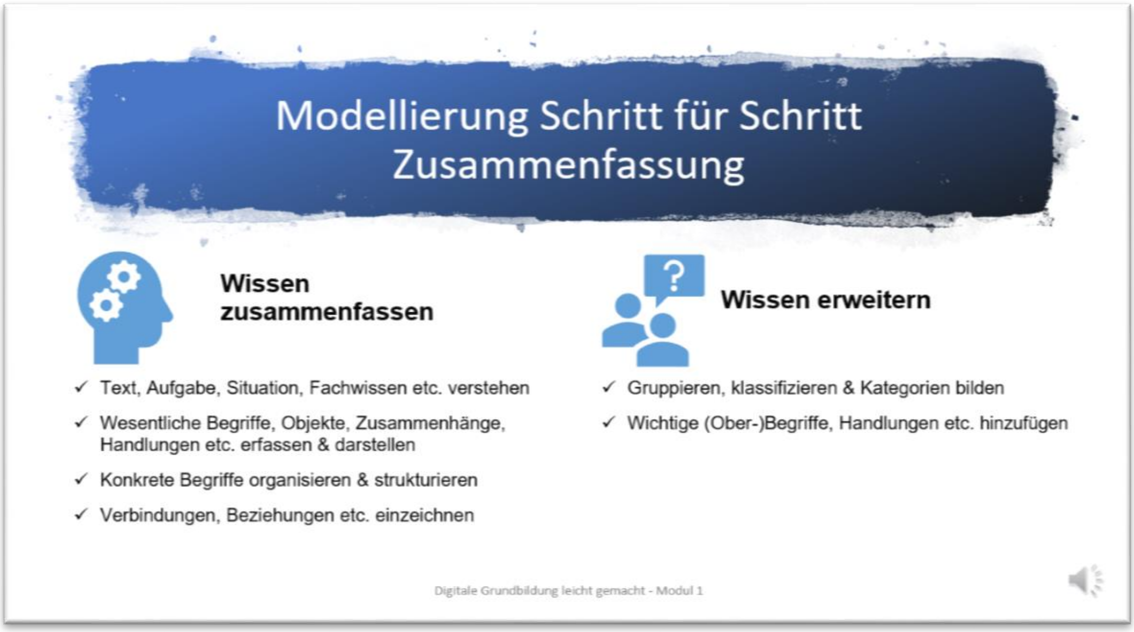


Figure 22: How does modeling work? Part I

6.1.5. How does modeling work? Part 2

The fifth video demonstrates a step-by-step instruction from modeling to computational thinking. Furthermore, it describes what is meant by computational thinking and where it can be found in the Austrian curriculum.

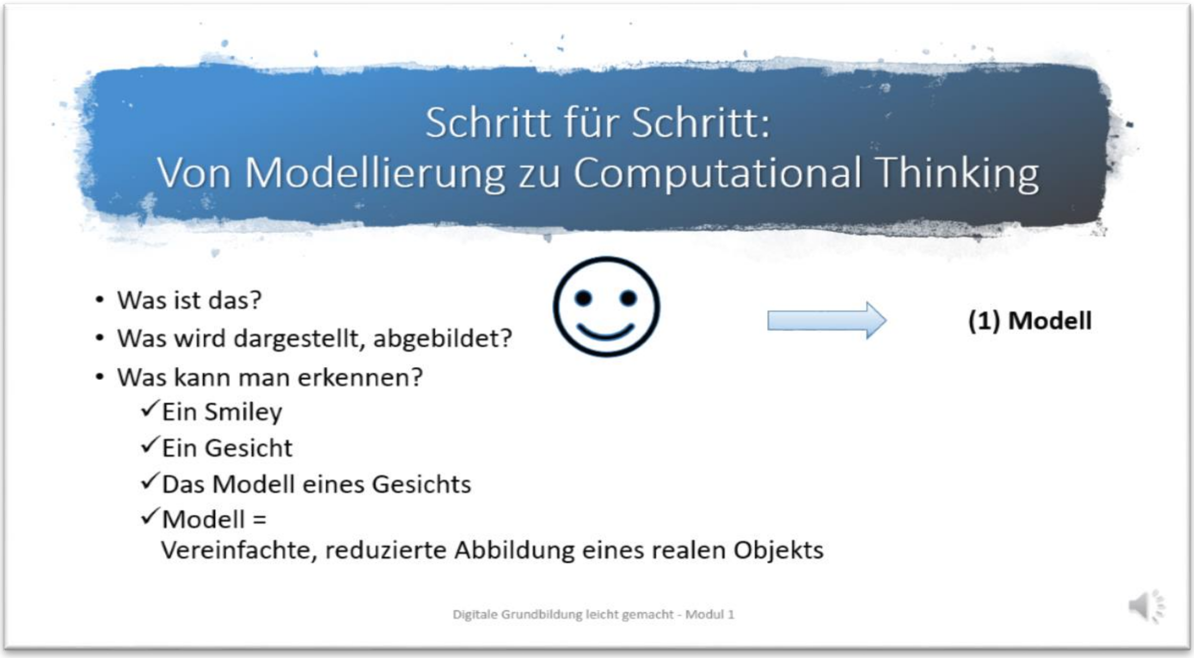


Figure 23: How does modeling work? Part II

6.1.6. What can I use to model? Useful tools to put it into practice.

In this video, we demonstrate several modeling tools. Modeling can be carried out in analogue or digital form. Besides using pen an paper, tools such as sticky notes, modelling clay, small cards or lego bricks can be very helpful, creative and an entertaining way to work with models. If you want to use digital tools for the implementation, there are various apps and programs for tablets, mobile phones and computer available as freeware. These tools are easy to use and do not require lengthy training.

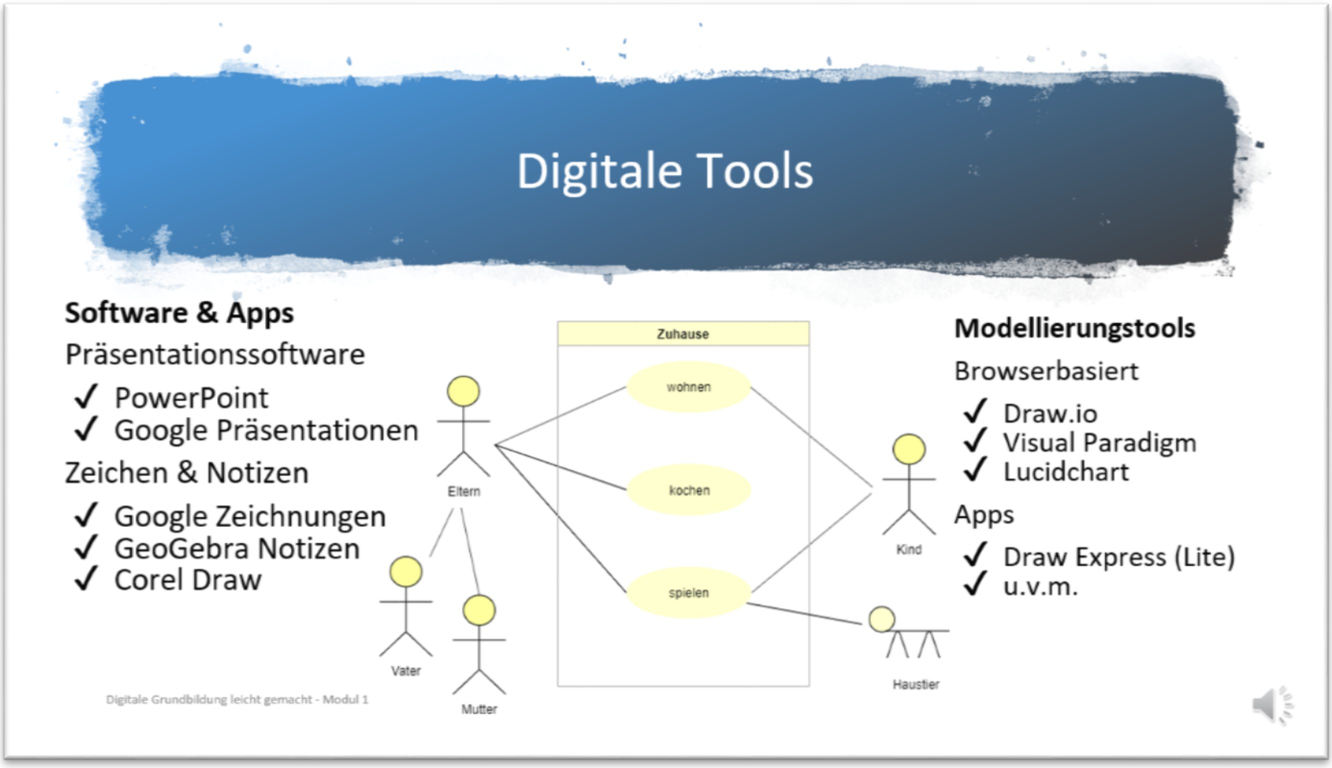


Figure 24: Useful modeling tools

6.1.7. Diagrams – step by step instruction

After the introductory videos, diagrams are presented in detail. The first video describes the entity-relationship diagram, the second deals with class- & object diagram and the third one presents the activity diagram. Besides a step-by-step instruction, the single elements and purpose of the diagrams are described in greater detail.

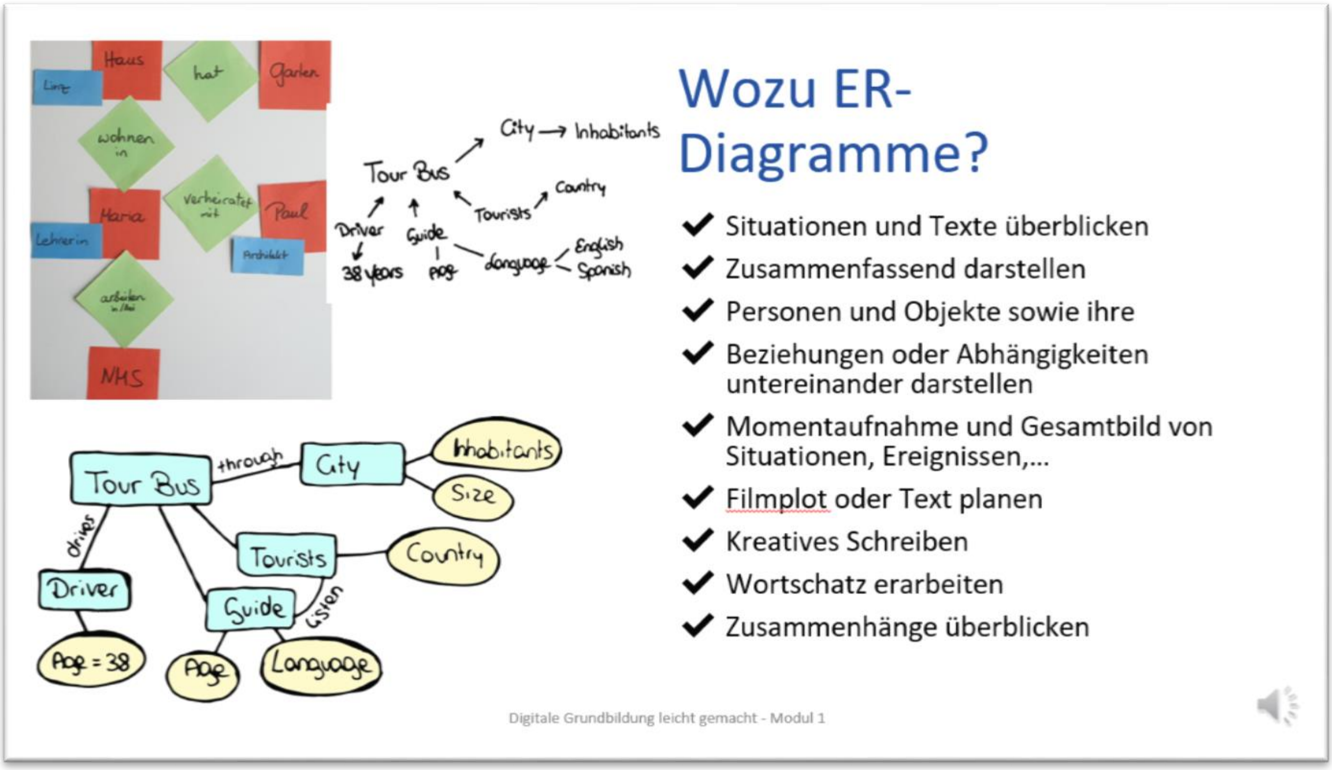


Figure 25: Entity-Relationship Diagram

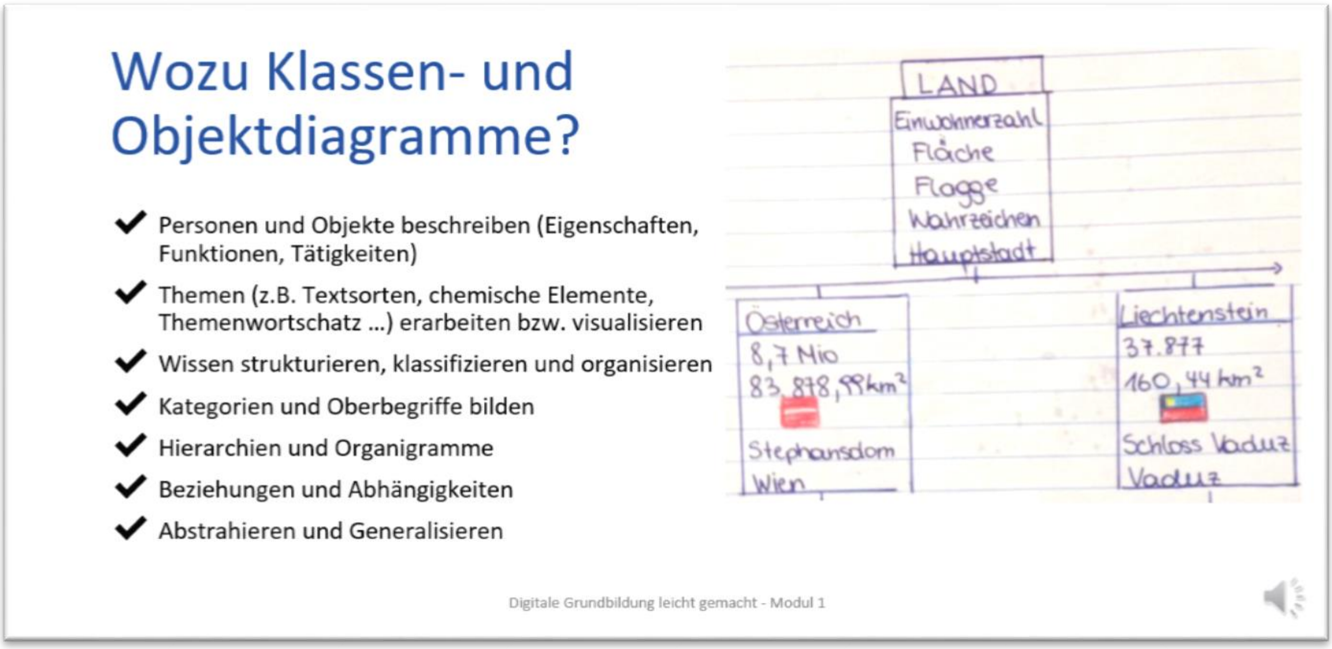


Figure 26: Class- & Object Diagram

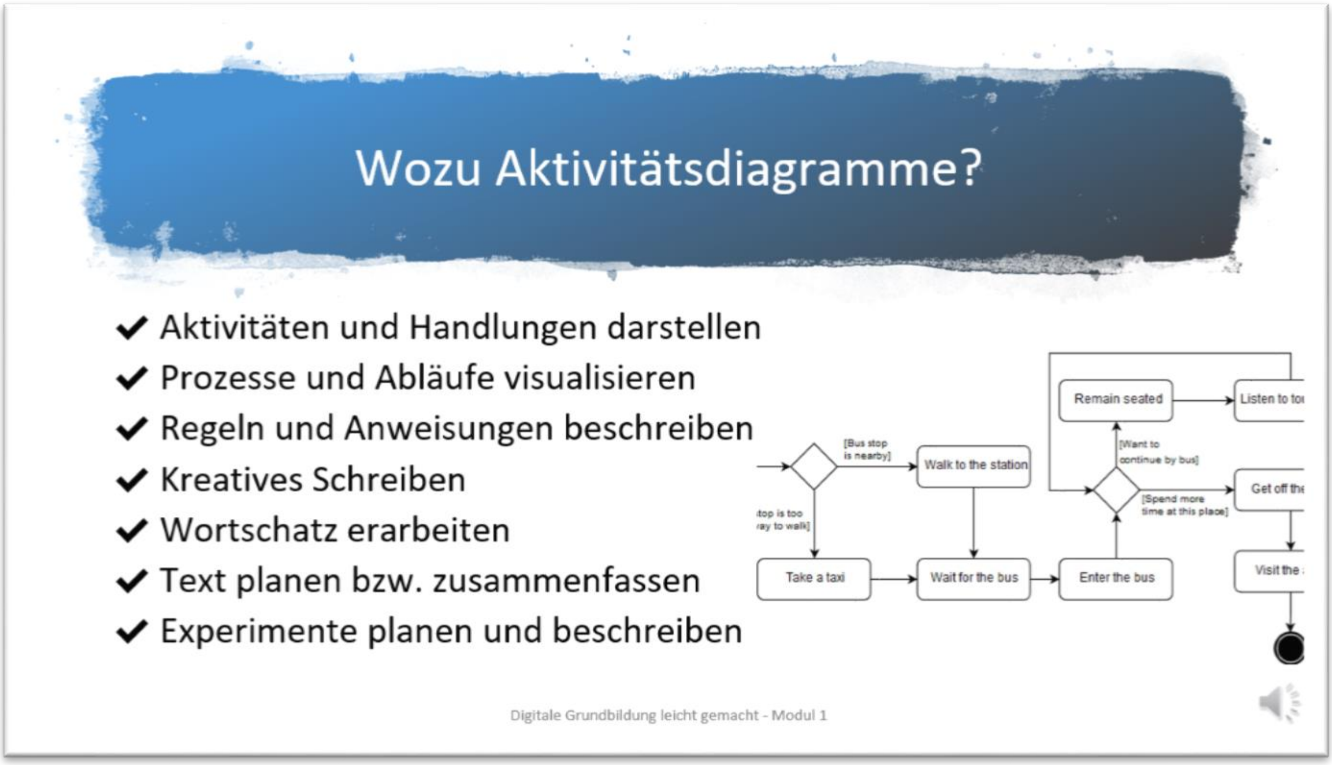


Figure 27: Activity Diagram

6.2 Finland

As a national workshop related to O2 and guidelines development, we presented the project at Finnish Educational Research Association conference at 21st and 22nd November 2019 at the University of Joensuu and received feedback from school directors, educators, and researchers, which we applied when adapting guidelines to Finnish context.

Because of teachers' time constraints and lack of incentives of participating in professional development, in Finland, it is challenging to involve teachers in a more extended online course (cf. Austrian model), if they are not previously interested in the topic. Unlike in many countries, teachers do not have obligatory in-service training hours, their professional development plan is very flexible and based on their own development needs. Finnish teachers tend also to be very self-directed and autonomous when searching information to serve their professional development. Furthermore, Finnish teachers generally master English language well and are used to read English texts. In Finland, teachers are also very used to read guidelines and are likely to find reading as a quicker way to get the information that watching videos, for instance.

For these reasons, at this point, we decided to keep the guidelines in English. Our method to motivate Finnish teachers to learn how to apply modeling in their teaching is to offer them a sample class (45-90 min) in which we work with their students and demonstrate together with the teachers, how modeling can be related to different subjects and key competences or transversal competences defined in Finnish National curriculum (FNCC, see also Intellectual output 1) in motivating way. Before the sessions, we will brief teachers about the objectives and provide them the guidelines which they can familiarise before and after the session. The version of the presentation of the guidelines to be used in Finland is also usable per se (as a teacher resource) if teachers wish to learn basics about modeling at schools. Currently, we are also considering options to provide teachers and students without the computer science background an easy-to-use game editors or analog game design methods. Teachers and students may be able to develop game-based examples for all the subjects, and implement modeling-based approaches as part of the game design.

6.3 Spain

In the Spanish context, Modeling is a new paradigm to exploit. Thanks to the insights gained by Programamos, INTEF and the research group KGBL3 in dissemination activities, our experience is that the activities must be on-site. Teachers have a series of well-defined content to develop and teach, being more difficult to innovate in new paradigms for teaching or learning, due to the scarce time. Prior activities carried out by KGBL3 demonstrated a good overall acceptance of ad-hoc activities within an adequate advance planning.

However, as other partners suggested before, we considered appropriate a prior phase of training for teachers, multipliers and mentors, which has to consist in ensuring the comprehension of the five main diagrams. To achieve this, an online folder located in Google Drive is provided, where the materials and guidelines are found.

Unlike other countries, in Spain, English proficiency is limited, becoming a concerning weakness within Educational System. Due to these arguments, all proposed activities of dissemination proposed in the project and the guidelines are translated to Spanish.

Spanish guidelines have been formatted and translated to Spanish based on the original version proposed by Austrian team. The document can be found in:

https://drive.google.com/file/d/1WGeVM2JCFRv-BYjG54HIjwbXeFYoEAjM/view?usp=sharing

7 References

[1] B. Sabitzer, H. Demarle-Meusel, and M. Rottenhofer, *“Modeling as Computational Thinking Language - Developing a Reference Framework,”* in 9th International Conference on Educational and Information Technology, 2020. (In press)

[2] C. of Europe. Council for Cultural Co-operation. Education Committee. Modern Languages Division. *Common European Framework of Reference for Languages: learning, teaching, assessment.* Cambridge University Press, 2001.

[3] D. M. Considine and G. E. Haley. Visual Messages: *Integrating Imagery into Instruction. A Media Literacy Resource for Teachers.* ERIC, 1999.

[4] D. A. Sousa*. How the brain learns.* Corwin Press, 2016.

[5] P21. 21st Century Skills - Battelle for Kids.

[6] Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). Computational thinking-A guide for teachers.

(European Commission) Supporting Key Competence Development: Learning approaches and environments in school education. Conference Input Paper. European Commission, 2019. Accessible online (30-1-2020): https://ec.europa.eu/education/file/6670/download_en?token=-fqDo2Gw

(European Council). Council Recommendation of 22 May 2018 on key competences for lifelong learning. The Official Journal 2018/C 189/01. Accessible online (30-1-2020): https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv%3AOJ.C_.2018.189.01.0001.01.ENG&toc=OJ%3AC%3A2018%3A189%3ATOC

(FNCC) Finnish National Board of Education: *National Core Curriculum for Basic Education 2014.* Helsinki , Finland : Finnish National Board of Education, 2016.

Key competences for lifelong learning. Luxembourg: Publications Office of the European Union, 2019. Accessible online (30-1-2020): https://op.europa.eu/en/publication-detail/-/publication/297a33c8-a1f3-11e9-9d01-01aa75ed71a1/language-en

(OECD) Definition and Selection of Key Competencies: Executive Summary. Definition and Selection of Key Competencies: Executive Summary. Accessible online (30-1-2020): https://www.oecd.org/pisa/35070367.pdf

Lai, E.R. & Viering, M.M. (2012). Assessing 21 st Century Skills : Integrating Research Findings. National Council on Measurement in Education. Vancouver, B.C.: Pearson.

Lavonen, J. & Korhonen, T. (2017). Towards twenty-first century education: Success factors, challenges, and the renewal of finnish education. Teoksessa S. Choo, D. Sawch, A. Willanueva & R. Vinz (toim.), Educating for the 21st Century: Perspectives, Policies and Practices from Around the World (s. 243–264). Singapore: Springer.

Miettinen, R. (2019). 21. vuosisadan kompetenssit – OECD kasvatuksen kielen uudistajana. Kasvatus 50(3), 203–215.

Griffin, P. & Care, E. (2015). Assessment and teaching of 21st century skills: Methods and approach. Dordrecht: Springer.

Pellegrino, J. W. & Hilton, M. (2012). Education for Life and Work: Developing Transferable Knowledge and Skills in the 21st Century. Washington, DC: The National Academies Press.

Voogt, J. & Roblin, N. P. (2012). A comparative analysis of international frameworks for 21st century competences: Implications for national curriculum policies. Journal of Curriculum Studies, 44(3), 299–321.

8 List of Tables

9 List of Figures