Bringing Computer Science Concepts into the Language Classroom: A Case Study on Teachers' and Students' Perception of Modeling to Teach Computational Thinking

Marina Rottenhofer, Mag.[a1], Sandra Leitner, Mag.[a1], Lisa Kuka, Mag.[a1], Barbara Sabitzer, Univ.-Prof.[in] MMag.[a] Dr.[in1]
[1]*Johannes Kepler University Linz, Austria, marina.rottenhofer, sandra.leitner_1, lisa.kuka, barbara.sabitzer@jku.at*

Abstract*– Computational thinking for everyone! Since Jeanette Wing's proposal in 2006 of computational thinking (CT) as a fundamental skill such as reading or arithmetics, CT has gained popularity all over the world. CT is a strategy that is needed to tackle problems in the field of computer science and includes elements such as pattern recognition, decomposition, abstraction, generalization, and algorithmic thinking. To be able to systematically tackle linguistic tasks, the language learner needs a set of problem-solving skills, too. In foreign language acquisition, the learner faces different linguistic learning problems, and thus, learning and mastering different problem-solving skills could reduce linguistic complexity and facilitate the learning process. To benefit from CT skills in language teaching and learning, the authors use an innovative method: modeling. In computer science, models, such as the entity-relationship diagram or UML (Unified Modeling Language) like activity diagrams are used to analyze and visualize complex tasks. Due to the many implementation options that these diagrams offer, they also prove to be useful in other areas. The authors especially focus on the field of language learning and investigate the use of computer science models as a teaching and learning strategy for students of all age groups. Employing a mixed- methods approach, this case study explores teachers' and students' views on the integration of modeling in language learning activities. Results demonstrated that there was a remarkable difference in students' learning performance as well as a positive attitude towards modeling as a teaching and learning strategy.

Keywords—modeling, computer science, UML, computational thinking, learning strategy, digital education

I. Introduction

Since Jeanette Wing introduced computational thinking (CT) as a "fundamental skill for everyone" in 2006 [1], much research has been done and many different definitions have emerged. Also in the educational sector, CT has gained increasing importance and found its way into many national curricula, including Austria.

Generally, CT is seen as a problem-solving strategy, but programming is mostly suggested as a tool to foster CT from an early age. However, even though programming is an essential skill in the 21st century, the intense focus on programming in CT education can lead teachers to believe that programming is required to be able to teach CT [2]. Other than that, CT should be introduced long before programming [3]. The authors' view coincides with Lu & Fletcher's statement. Since they see CT as much more than the ability to program, they aim to spread CT as an unplugged (without the use of technical devices) problem-solving technique for every subject. As a basis for an unplugged instruction of CT, the authors use models from the field of computer science (CS) and implement them as a teaching and learning strategy in different subjects.

Modeling in computer science can be defined as "an abstract description of a real or planned system, which contains the essential properties of the system for a specific objective." [4, p.4]. In language learning, as well as in other subjects, modeling triggers deep problem-solving processes, and thus, as Hubwieser states, modeling has great importance for general education and should also not be overseen in IT lessons [5]. Our objective of using models as a teaching and learning tool is threefold: firstly, we aim to provide a hands-on tool to teach CT easily in every subject and school level. Secondly, we want to support students' learning, and thirdly, we aspire to spark students' interest in computer science.

Since technology is changing the world around us rapidly, more and more skilled workers are needed. As the Digital Economy and Society Index 2021 (DESI) reports, 55% of companies in Europe have difficulties in finding ICT specialists. The lack of individuals having sufficient digital skills indicates that more training offers are needed [6]. In Austria, computer science is a compulsory subject that starts relatively late in the 9th grade. Before that, only school-independent concepts are being implemented. However, digital skills need to be fostered long before that. So, also students' interest in CS and other STEM fields (Science, Technology, Engineering, and Mathematics) can be sparked much earlier. Several studies also claim that early exposure to STEM can spark interest in these topics and influence future job choices [7-9]. Besides increasing students' interest, the early training in CT skills helps them to tackle a life strongly influenced by computing. As suggested by Barr and Stephenson, students "must begin work with algorithmic problem solving and computational methods and tools in K-12" [10].

This paper presents a case study focusing on language learning which is part of a longitudinal study that aims to familiarize students and teachers with CS modeling and computational thinking in all subjects and school levels. The study was conducted within a 4-month-period in 2020/21 and the subjects of this study had no prior experience with modeling. This study aimed to find out more about teachers' and students' perceptions of modeling and to investigate the

Digital Object Identifier (DOI):
http://dx.doi.org/10.18687/LACCEI2022.1.1.89
ISBN: 978-628-95207-0-5 **ISSN:** 2414-6390

potential and challenges of implementing these concepts in the language classroom.

II. THEORETICAL BACKGROUND

In our research, diagrams from the field of computer science are used as graphic organizers in all subjects and school levels. A graphic organizer is "a visual and graphic display that depicts the relationships between facts, terms, and or ideas within a learning task. Graphic organizers are also sometimes referred to as knowledge maps, concept maps, story maps, cognitive organizers, advance organizers, or concept diagrams" [11]. Originally, graphic organizers have their roots in Ausubel's theory on advance organizers (AO). According to his theory, AOs are introduced at the beginning and support learners in connecting prior knowledge to new information. He claims that AOs "explicitly draw upon and mobilize whatever relevant subsuming concepts are already established in the learner's cognitive structure and make them part of the subsuming entity" [12]. In Hall et al. [11] several studies are presented that confirm the positive effects of graphic organizers on learning outcomes. Especially reading comprehension and vocabulary knowledge has often been the subjects of investigation, which underlines the usefulness of this method in foreign language learning.

In this study, UML models [13], as well as the entity-relationship-model [14] are implemented as teaching and learning strategies in the language classroom. The process of modeling has various definitions. Generally, a model is a simplified and reduced description of a real or planned system with the essential elements of it and the creation of such models is called modeling [4]. In the field of CS, modeling is a fundamental discipline for describing, presenting, and communicating data [15]. Similarly, Pilone and Pidman describe modeling as "a means to capture ideas, relationships, decisions, and requirements in a well-defined notation that can be applied to many different domains" [13, p.5]. To create different models, many skills are required, which are well served by CT (e.g. pattern recognition, decomposition, abstraction, generalization, or algorithmic thinking). Previous studies on the use of modeling as a teaching and learning strategy have shown promising results [16]. In second language acquisition, in particular, experience has shown that modeling is a very useful tool [17-18]. These findings led to further and more intensive investigations in this specific area.

III. COMPUTATIONAL THINKING AND COMPUTER SCIENCE EDUCATION IN AUSTRIA

In Austria, computer science as a compulsory subject is limited to the 9th grade with two 50-minutes-teaching hours per week [19]. Earlier, in lower secondary schools (grades 5 to 8), computer science can be implemented school-independently. In other words, there is no uniform curriculum for the implementation of CS before 9th grade and so, there is often a big heterogeneity among the students [20]. To meet the demands of digitalization, the Austrian government released the "Master Plan for Digitalization" in 2018 [21], which includes expanding the technical infrastructure, setting a stronger focus on digitalization in teacher training, and introducing the curriculum *Basic Digital Education* [22]. Until the end of the school year 2021/22 this curriculum can be implemented as an elective subject or integrated into other subjects, such as foreign language learning. From the coming school year 2022/23, the curriculum will be implemented as a compulsory subject in lower secondary schools. The curriculum of *Basic Digital Education* consists of eight areas, whereas CT is one of them. However, according to our experience, this is the part of the curriculum that teachers with no digital background find particularly difficult to put into practice. In the Austrian curriculum, CT is described as (1) the ability to work with algorithms and (2) the creative use of programming languages. Especially schools that are not technically well equipped or teachers with no previous programming experience see a major hurdle when implementing different programming languages in their subjects. Thus, the authors aim to make CT more tangible for teachers by introducing them to the main pillars of CT with modeling as a teaching and learning tool.

IV. COMPUTATIONAL THINKING AND FOREIGN LANGUAGE LEARNING

In the language classroom, computational thinking skills can be used to support students' learning processes and add to their repertoire of learning strategies. Complex grammar explanations with dense information and numerous exceptions can lead to a cognitive overload and, consequently, to discouragement. To foster the notion of keeping a clear head and developing a strategic mind, CT skills, as well as its tools and diagrams, are proving to be useful.

CT can be described as a thinking process that is usually subdivided into several skills such as decomposition, abstraction, pattern recognition, and generalization, as well as algorithmic thinking and debugging [23]. To solve a problem, it can be helpful to subdivide it into several smaller problems that are easier to handle, which is usually referred to as decomposition. Focusing on important information by omitting unnecessary details (abstraction) to find similarities, differences, and patterns (pattern recognition) helps to define problems more clearly and to find solutions that are transferable from one problem to another (generalization). Formulating a step-by-step guide, so that a computer or other persons can apply it to other problems as well is called algorithmic thinking. Lastly, the algorithm or solution has to be tested, evaluated, and, if necessary, readjusted to the results of this evaluation process (debugging). This process is done by many people in different fields on a daily basis. For example, the teacher who uses an inductive grammar teaching method by giving students several grammatical items and encouraging them to find the grammatical rule behind it is strengthening CT skills in their students. First, they have to find a pattern, which is done by focusing on similar and different words in the sentence. Then, they have to find generalized terms for what they have observed, e.g. -ed is added after the regular verb to form the past tense. Afterward, they try to come up with a step-by-step guide, on how to formulate the grammatical item and which questions have to be asked to form it right, e.g. is the verb irregular then check the list with the irregular past participle words, otherwise add -ed to the verb. There are numerous tools and diagrams that facilitate and visualize the computational thinking process, for example, activity diagrams, ER diagrams, class diagrams, etc. In the end, the outcome of the developed algorithm can be tested on several examples and adjusted if needed, in the example the students may come across situations with a negation and now have to include this in their algorithm as well.

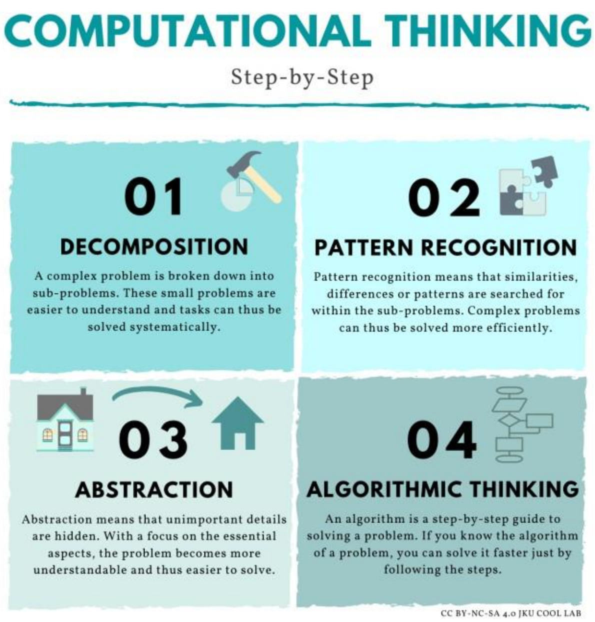

Fig. 1 Areas of Computational Thinking

A. Modeling as a Tool to teach Computational Thinking in the Language Classroom

Modeling with CS diagrams offers a powerful opportunity to introduce these computational thinking concepts in an easy and unplugged way that allows students to immediately dip into the learning process. Furthermore, this approach represents an opportunity for cross-curricular cooperation between language and computer science instruction. To facilitate the implementation of modeling, a reference framework for modeling (ReMo) has been developed. The framework is an assessment tool and guideline for students and teachers based on the Common European Framework of References (CEFR) that describes modeling proficiency and describes the computational thinking processes that occur [24]. Teachers can provide models as visualization of complex learning content or let students actively engage in the modeling process. The latter requires students to apply various CT skills to be able to turn the main information of a text into a model. This chapter presents three different CS diagrams that have proven to be very suitable for the first encounter with modeling. Especially in foreign language learning, these models offer a wide range of possible uses. In computer science, static diagrams and dynamic diagrams are used to visualize different aspects of a real system. When students develop their models, both static and dynamic diagrams foster decomposition, abstraction, generalization, and pattern recognition. Added to this, dynamic diagrams allow training algorithmic thinking skills. For this case study, we used class and object diagrams and the entity-relationship diagram as static-, and the activity diagram as a dynamic model.

Class and object diagrams are used to visualize various elements of systems and their relations. Class diagrams represent the abstract model of a system, whereas object diagrams depict the concrete objects in a certain moment [25]. In the language classroom, these models are ideal to elaborate vocabulary, visualize relations and hierarchies, and categorize terms. As presented in figure 2, the name of the class are nouns, attributes are seen as adjectives, and methods as verbs. These classes can then be concretized in objects, where the animals are e.g. a cat with its specific attributes and methods.

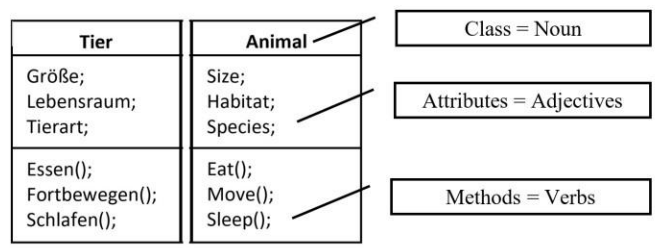

Fig. 2 Class Diagram

The entity-relationship (ER) is easy to acquire and implement since it only consists of three elements. The rectangles are “entity-types” that are used as nouns, the diamond shapes are “relationship types” and the ellipses are “attributes” that describe the relationships and characteristics of the nouns. This diagram has proven very useful for text work, especially as an intermediate step when writing summaries, as it resulted in more coherent texts. In computer science, the ER diagram does only use generic instead of specific terms, since it represents a type of a system and not an instance [26]. However, in our approach, the subject and not the diagram syntax lies in the foreground. So, one can use this diagram with concrete terms of a text and/or abstract terms. The diagram in figure three had been developed with a reverse-engineering approach, where, as a first step, parts of the text (source code) were highlighted with colored shapes. Subsequently, these elements were turned into an ER diagram with generalized terms.

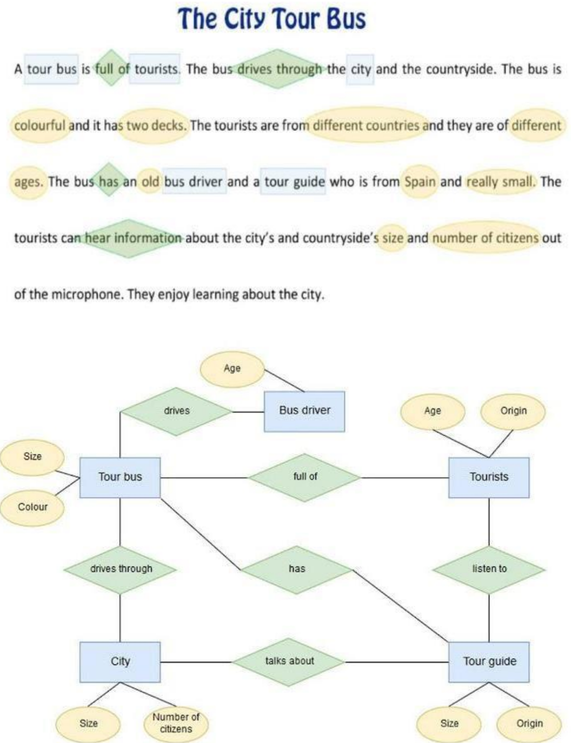

Fig. 3 Entity-Relationship Diagram

The activity diagram visualizes a series of activities of a certain process and is very suitable to introduce algorithms to the students. In foreign language learning, grammar rules, as well as a series of events of a story or instructions can be elaborated easily. With the activity diagram, the complexity of e.g. long grammar explanations is reduced by extracting the most important information. By creating the model, a bigger problem is decomposed into smaller, more tangible pieces and turned into a step-by-step instruction. For a teaching and learning approach, the activity diagram is reduced to elements such as initial and final nodes, rectangles for the activities, rhombuses for decisions, and rules of loops and branching. Figure four illustrates step-by-step instructions on how to form the simple past in English.

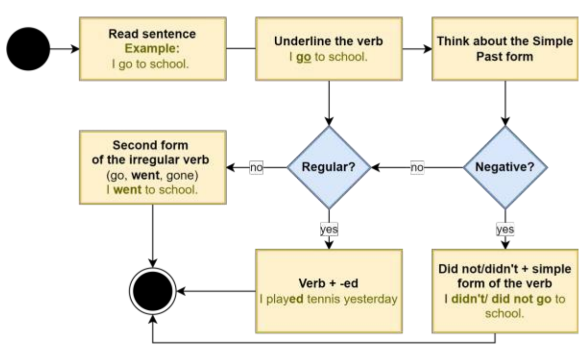

Fig. 4 Activity Diagram

V. The Study

This paper presents a single case of a multiple case study based on Yin [27] focusing on the usefulness and practicability of modeling as a teaching and learning strategy. The study as a whole covers several schools collaborating with the COOL Lab of the Johannes Kepler University Linz, where modeling has been implemented and investigated. The JKU COOL Lab is an innovative teaching and learning lab for teachers, students, and university students and deals with topics related to digital education. Besides events, workshops, and weekly clubs, the COOL Lab regularly offers in-service training for teachers across Austria and more intensive training and support for several partner schools.

To provide high-quality training that leads to changes in teaching behavior and influences students' learning outcomes positively, a training model, the Educational Pyramid Scheme (EPS), has been developed in the course of a 3-year Erasmus+ project entitled "Modeling at School" [28]. With the EPS, different target groups are involved in the training process (students, teachers, and prospective teachers), take on various roles (multiplier, mentor, tutor), and convey the knowledge on several levels. The multiplier is the conduit between the school and the lab and spreads the innovation within his or her institution by recruiting other teachers to participate. These teachers function as mentors who, together with the tutors, implement the knowledge in the classroom. Tutors are students who collaborate with and support their peers in the learning process.

This case study on modeling was conducted in one of the partner schools and occurred over 4 months from March to June 2021. Before the collaboration was established, none of the students and teachers had attended the COOL Lab's offerings and so the topic of modeling was still unknown. The subjects of the investigation were one Spanish teacher and her 12th-grade Spanish class (age 16-18 years). This teacher was the multiplier of the school, who was the COOL Lab's contact person and promoted the innovation within her institution. Furthermore, she also functioned as a mentor, who acquired the new content herself and implemented it in one of her classes. The students were in their third year of learning Spanish and had three instructional periods per week. To find out more about the usefulness of modeling as a teaching and learning strategy, the study addressed the following research questions:

1. How do teachers and students perceive modeling as a teaching and learning strategy?
2. What are the chances and challenges of using modeling as a tool to foster computational thinking?
3. How does modeling facilitate language learning?

A. Implementation

The implementation of the case study can be divided into four phases:

1) Teacher Training: At the beginning of this study, the Spanish teacher with no previous modeling experience received initial training from the JKU COOL Lab. The training included an online session of 90 minutes where she got introduced to the three different models mentioned in section 4a: activity diagram, class diagram, and entity-relationship diagram. Furthermore, best practice examples for language instruction were shown and implementation options in her context were discussed. To facilitate the implementation of CS models in her Spanish class, the Reference Framework of Modeling (ReMo) was also presented to the teacher. After this initial session, video tutorials on modeling were made available to the teacher, and support was offered when needed during the implementation. The teacher was very inquisitive and motivated to use the new strategy in class right away.

2) Guided Classroom Implementation: After the training phase, the teacher implemented the knowledge gained in her 12th grade Spanish lessons. Due to the COVID-19 pandemic, students attended school in different shifts, and thus, the teacher used a hybrid model that combined face-to-face and online teaching simultaneously. After explaining the project for the next few weeks to the whole class (hybrid), the teacher divided the students into a treatment group TG working with models, and a control group CG using the traditional approach. The circumstances caused by the pandemic did not allow the groups to be randomly assigned. Both, TG and CG consisted of students working at school (A) and online from home (B). In other words, students also collaborated in a hybrid form (see table 1) but changed weekly between working at school and online from home.

TABLE I
OVERVIEW STUDENTS

	Group A (face-to-face)	Group B (online)
Treatment Group TG	4	3
Control Group CG	4	4

For three weeks, the TG and CG were divided into two separate classrooms and worked online with their peers at home. In this phase, the TG students had the task of independently acquiring three models, training each other in a hybrid form, and applying the models to the Spanish learning content. To facilitate the implementation, students were given information sheets and video tutorials for each of the three models. Simultaneously, the CG continued to work traditionally on their Spanish tasks and shared the elaborated content in an online document for the TG. This content was then visualized with suitable diagrams by the TG. To record the modeling progress and make the diagrams available to everyone, students created a *GeoGebra* e-book that subsequently served as learning material for the upcoming exam.

3) Peer Learning: After the three weeks, students of the TG acted as tutors and passed on their experiences to the CG students. At this time, there was no longer remote learning and thus, all students were present at school. The TG showed them how the learning material was prepared with diagrams and supported their peers of the CG in their learning progress.

4) Observation Phase: The last phase aimed to find out whether students actively use modeling as a learning strategy without being directly asked for it. In other words, modeling was no longer given as a direct task in class. The teacher slipped into the role of the observer and supported the students only when help was requested from their side.

B. Methods

This case study occurred over 4 months from March to June 2021. To establish the trustworthiness of the case study, the following sources of evidence proposed by Yin [27] were used:

1) Documentation: In the course of the study, members of the JKU COOL Lab held regular consultations with the teacher, where the implementation in the class was discussed and notes were taken. Furthermore, students' results on tests were collected.

2) Observation: In the school year 2020/21, where the investigation took place, the teacher had three instructional periods for 50 minutes per week. During these Spanish lessons, the teacher was a participant observer and collected field notes during and after the observation period.

3) Interviews: At the end of the study a focus group interview [29] was conducted where the implementation of modeling including its advantages and disadvantages were reflected and discussed intensively. The teacher asked for students' feedback regarding the usefulness and practicability of modeling.

4) Physical Artifacts: Physical artifacts created by the students were another source to ensure triangulation. During the weeks of guided implementation, students had the task to create a *GeoGebra* e-book, where they collected all the material they had elaborated with modeling. Although the process of developing the models could be directly observed by the teacher, the *GeoGebra* e-book with the final results allowed the researchers to gain a broader perspective concerning the implementation of modeling.

C. Outcomes

1) Teacher Perception of Modeling: According to the teacher, the main motivation for changing the learning strategy in foreign language lessons at her school was the two-year search for a suitable method for vocabulary training, text comprehension, and oral and written production. After the initial training by the JKU COOL Lab, the teacher felt prepared to implement it right away in her Spanish lessons. According to her, since it is a hands-on approach and no technical skills are needed, this is an ideal way to teach CT. Especially in her language lessons, she saw great potential. In the implementation phase, the Spanish teacher noticed that especially students with difficulties in foreign language learning had advantages from the benefits of collaborative learning supported by the tutor system, as well as visualizing difficult learning content. Surprisingly, even speaking in a foreign language was facilitated by the visualization of a text displayed as a diagram. During the guided implementation, one group (TG) had the task to acquire modeling skills and apply them to the current learning content, while the teacher was working traditionally with the CG in another classroom. At first, the teacher was unsure whether the students could complete the task on their own. But then it turned out that with the help of the video tutorials and the information sheets, the students mastered and implemented the diagrams without any problems. In addition, they worked independently, shared tasks within the group, and supported each other. Then, in phase three, the teacher also did not have to intervene and stayed in the background. It was interesting to see how seriously the tutors took their role and taught the other classmates from the CG. She described these phases as follows:

> "Altogether, the combination of modeling as teaching and learning strategy and the training method showed that students can help their fellow students in a very effective way to learn a foreign language."

In summary, the implementation in the 12th grade was very successful. Due to the independence of the students, no additional time was needed to learn the new concepts. On the contrary, the tutoring system relieved her of the burden of hybrid teaching and she was able to concentrate on a small group and support the weaker students more intensively.

Because of the positive experience, she immediately introduced modeling to another Spanish class. This group consisted of 7th-grade students (age 12-13 years) in their first year of learning Spanish. This time she has experienced that introducing the new concepts took more time. On the one hand because of the language learning year and on the other hand because of the age and the cognitive level of the students. In addition to a completely new language, the students also had the task of learning the modeling syntax. To avoid cognitive overload, one model after the other should be introduced slowly and with intensive support from the teacher. At first, they only worked with class diagrams and vocabulary. However, clustering helped them enormously in language acquisition. Subsequently, further models and concepts such as algorithms and generalization will have been introduced. In summary, in lower grades, the teacher needs to be aware that more intensive support and more time are needed to successfully teach the concept.

2) Student Perceptions of Modeling: The students did respond very positively to the project and saw the benefit of modeling as a teaching and learning strategy for language classes. They particularly liked the clustering of vocabulary with class diagrams, text summaries with ER diagrams, and the concept of algorithms. Through the introduction of modeling and computational thinking, they became aware of the connection to computer science and learned new skills that they can apply in different situations. Concerning learning the content without the guidance of the teacher, they had no problems. On the contrary, they liked that the teacher trusted them to work independently. The fact that the students took the task seriously and were able to do it on their own is probably due to the age of the students and because self-determined learning was an important topic at school. The tutoring system allowed the students to slip into the role of teachers and pass on their knowledge to the CG in phase three. They liked this active role very much. The following is a personal reflection of one of the students:

> "I am a very good student, but I have great difficulties learning a foreign language. I have already tried many methods of how I can best memorize the vocabulary. By designing a class diagram, I can now remember the words much better because they are categorized and assigned in relation. When we presented our E-book with the modeled content to our classmates, everyone was enthusiastic and saw immediately how supportive the diagrams will be when learning. Also, the transformation from a text to an Entity Relationship Diagram helps to learn the most important vocabs of the content of a text. In addition, it helps a lot to remember the content of a topic."

However, there are also hurdles in implementing the modeling in everyday school life. Although modeling is considered a great learning strategy, implementation is not always possible. The big hurdle is the lack of time. Especially in times of the Covid-19 pandemic, the students were covered with work and the additional modeling of the content was time-consuming.

3) Sustainability & Learning Outcomes: Overall, it can be concluded that the implementation of modeling and CT supported students in learning a foreign language. During the observation phase, the teacher noticed that the students independently applied the models to the learning content. More specifically, when they worked on a text in their books, several students made sketches of ER diagrams next to the texts to summarize the main content. Very often, a domino effect could be observed. When some students started to model, other students copied the sketches or created their own. Students reported that these sketches not only helped to summarize the main points of the text but also as a basis for oral tasks and discussions. Furthermore, students continued to cluster the vocabulary with class diagrams. However, this became less and less due to the lack of time during the pandemic.

Fig. 5 Chapter of the GeoGebra E-Book

In addition, the students reported that the *GeoGebra* e-book was very helpful in preparing for the test and that is why they used it as a template for developing new learning content as well. Analyzing the *GeoGebra* book, it can be concluded that the students understood the main concepts of modeling and CT and successfully acquired the learning strategy independently. Furthermore, the teacher was able to determine an improvement in the summaries concerning coherence on the one hand and an increase in performance on the other. After the new learning strategy was introduced, many students' grades (1 = highest grade, 5 = lowest grade) improved in the second written exam (see Figure 6). Results showed students had a higher score in exam 2 (mean = 2.00, SD = .926) than in exam 1 (mean = 3.00, SD = 1.254). A paired samples t-test found this difference to be significant, $t(14) = 5.12$, $p < 0.001$. Together this suggests that modeling may have positive effects on students' learning outcomes, supporting our hypothesis. However, further studies are needed to determine which other factors have influenced performance improvement.

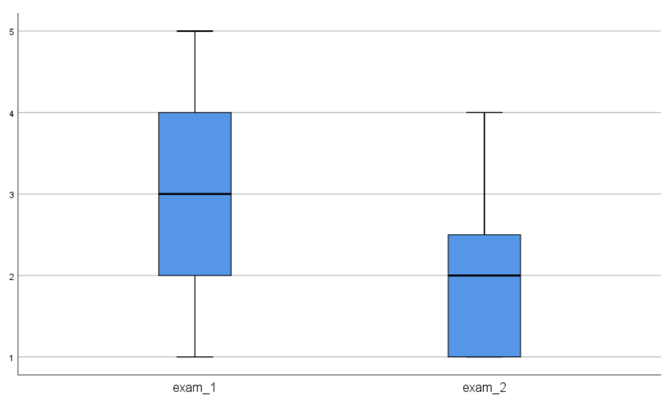

Fig. 6 Boxplot of Students' Grades in Exam 1 and Exam 2

TABLE 2
PAIRED SAMPLES T TEST RESULTS

				95% Confidence Interval of the Difference				
	M	SD	S.E. mean	Low.	Upp.	t	df	Sig. (2- t.)
Pair 1 e1- e2	1,000	,756	,195	,581	1,419	5,123	14	,000

VI. DISCUSSION & CONCLUSION

Computational Thinking (CT) has gained considerable attention in the last years and is already part of many school curricula. To facilitate the implementation of CT in different school levels and subject areas, an innovative method is used: models from the field of computer science as a teaching and learning tool. This case study aimed to get a better understanding of teachers' and students' perceptions of modeling and to find out more about chances, challenges, and language learning outcomes. From the analysis and triangulation of the collected data, it could be seen that modeling is a very useful tool to teach and learn a foreign language as well as to train CT skills. The students involved in the case study have benefited greatly from the project and continued to use the learning strategy. Some students were so enthusiastic about the modeling that the project spread throughout the school and other classes also wanted to learn this strategy. The teacher also saw the great potential in introducing CT with modeling and started to implement it in other classes as well. Despite the benefit of modeling, also challenges could be noticed. Students reported that modeling is time-consuming and thus, often it was not possible to integrate it, whereas the teacher also experienced that it required more effort to introduce modeling in lower grades. However, it is worth investing this time, as the students benefit greatly from this method and new content is elaborated more easily and anchored in the long term. Since other classes have already expressed the desire to learn modeling as well, an intensive all-day workshop was planned for January 2022. Unfortunately, this had to be postponed due to the rapidly spreading COVID-19 Omicron variant. As soon as the situation has returned to normal, the partner school will put a strong focus on computational thinking and modeling and further studies will be conducted.

REFERENCES

[1] Wing, J. M. (2006). "Computational thinking." *Communications of the ACM*, 49(3):33–35.

[2] Voogt, J., Fisser, P., Good, J., Mishra, P., Yadav, A. (2015). "Computational thinking in compulsory education: Towards an agenda for research and practice." *Education and Information Technologies*, *20*(4), 715-728.

[3] Lu, J. J., Fletcher, G. H. (2009). "Thinking about computational thinking." *Proceedings of the 40th ACM technical symposium on Computer science education* (pp. 260-264).

[4] Hubwieser, P., Mühling, A., & Aiglstorfer, G. (2015). *Fundamente der Informatik. Funktionale, imperative und objektorientierte Sicht, Algorithmen und Datenstrukturen.* Oldenbourg, GE:Wissenschaftsverlag.

[5] Hubwieser, P., Broy, M. (1996). *Der informationszentrierte Ansatz: Ein Vorschlag für eine zeitgemäße Form des Informatikunterrichtes am Gymnasium.* Institute for Computer Science Technical University Munich, Technical Report, Munich

[6] European Comission (2021). *Human capital and digital skills in the digital economy and society index.* Retrieved from: https://digital-strategy.ec.europa.eu/en/policies/desi-human-capital

[7] Swift, T. M., Watkins, S. E. "An Engineering Primer for Outreach to K-4 Education," *Journal of STEM Education*, Institute for STEM Education and Research, Jan 2004.

[8] Maltese, A. V., Tai, R. H. (2010). "Eyeballs in the fridge: Sources of early interest in science." *International Journal of Science Education*, 32(5):669–685.

[9] Russell, S. H., Hancock, M. P., McCullough, J. (2007). "Benefits of undergraduate research experiences." *Science*, 316(5824), 548-549.

[10]Barr, V., & Stephenson, C. (2011). "Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?" *Acm Inroads*, 2(1), 48-54.

[11]Hall, T., & Strangman, N. (2002). "Graphic organizers." *National Center on Accessing the General Curriculum*, 1-8.

[12]Ausubel, D. P. (1960). "The use of advance organizers in the learning and retention of meaningful verbal material." *Journal of educational psychology*, 51(5), 267.

[13]Pilone, D., Pitman, N. (2005). *UML 2.0 in a Nutshell*. Sebastopol, CA: O'Reilly Media, Inc.

[14]Chen, P. P. S. (1977). The entity-relationship model: a basis for the enterprise view of data. *Proceedings of the June 13-16, 1977,* National Computer Conference, New York, NY: 77-84.

[15]Diethelm, I. (2007). *Strictly models and objects first: Unterrichtskonzept und-methodik für objektorientierte Modellierung im Informatikunterricht* (Doctoral thesis, University of Kassel, Kassel, Germany). Retrieved from https://d-nb.info/98668760x/34

[16] Sabitzer, B., Demarle-Meusel, H., & Rottenhofer, M. (2020). Modeling as Computational Thinking Language: Developing a Reference Framework. *Proceedings of the 2020 9th International Conference on Educational and Information Technology*. Association for Computing Machinery, New York, NY: 211-214.

[17]Rottenhofer, M., Rankin, T., & Sabitzer, B. (2020). Grammar Instruction with UML. *2020 IEEE Frontiers in Education Conference (FIE),* Uppsala, SWE: 1-5.

[18] Rottenhofer, M., Sabitzer, B. & Rankin, T. (2021). Developing Computational Thinking Skills Through Modeling in Language Lessons. *Open Education Studies*, 3(1), 17-25. https://doi.org/10.1515/edu-2020-0138

[19] Siller, H. S., & Fuchs, K. (2010). „Bemerkungen zur Fachdidaktik Informatik." *25 Jahre Schulinformatik in Österreich-Zukunft mit Herkunft* (pp. 121-125).

[20]Jarz, T. (2010)."Aktuelle Probleme der Schulinformatik." In Brandhofer, G., Futschek, G., Micheuz, P., Reiter,A., Schoder, K., editors, *25 Jahre Schulinformatik.Zukunft mit Herkunft,* pages 116–120.

[21]BMBWF Bundesministerium für Bildung, Wissenschaft und Forschung. (2018a). *Masterplan für die Digitalisierung im Bildungswesen.* Retrieved from: https://www.bmbwf.gv.at/Themen/schule/zrp/dibi/mp.html

[22]BMBWF Bundesministerium für Bildung, Wissenschaft und Forschung. (2018b). *Digitale Grundbildung. Änderung der Verordnung über die Lehrpläne der Neuen Mittelschulen sowie der Verordnung über die Lehrpläne der allgemein- bildenden höheren Schulen.* Retrieved from:https://www.ris.bka.gv.at/Dokumente/BgblAuth/BGBLA_2018_II_71/BGBLA_2018_II_71.html-

[23]S. Bocconi, A. Chioccariello, G. Dettori, A. Ferrari, K. Engelhardt, P. Kampylis, Y. Punie (2016). "Developing computational thinking in compulsory education." *European Commission, JRC Science for Policy Report*

[24] Sabitzer, B., Demarle-Meusel, H., Rottenhofer, M. (2020). Modeling as Computational Thinking Language: Developing a Reference Framework. *Proceedings of the 2020 9th International Conference on Educational and Information Technology* (pp. 211-214).

[25]Seidl, M., Brandsteidl, M., Huemer, C. & Kappel, G. (2012). *Uml@ classroom. An Introduction to Object-Oriented Modeling.* Switzerland: Springer.

[26]Bagui, S. & Earp, R. (2011). *Database design using entity-relationship diagrams*, second edition. New York: CRC Press.

[27]Yazan, B. (2015). "Three approaches to case study methods" in education: Yin, Merriam, and Stake. *The qualitative report*, 20(2), 134-152.

[28]Demarle-Meusel, H., Rottenhofer, M., Albaner, B., & Sabitzer, B. (2020). Educational Pyramid Scheme–A Sustainable Way of Bringing Innovations to School. *2020 IEEE Frontiers in Education Conference (FIE),* Uppsala, SWE: 1-7.

[29]Powell, R. A., Single, H. M. (1996). "Focus groups." *International journal for quality in health care,* 8(5), 499-504.