

Teachers, Reload Your Toolboxes: Using a Task Design Framework and ChatGPT to Generate Motivational Exercises for Female Learners in Computer Science

Sibylle Kunz and Claudia Hess

IT & Engineering

IU International University of Applied Sciences

Erfurt, Germany

e-mail: sibylle.kunz@iu.org; e-mail: claudia.hess@iu.org

Abstract—Challenging learners at school with interesting activities that foster competencies that are later required at university is an important part of a teacher’s job. But creating these tasks is time-consuming, and often these activities are traditionally more adapted to the interests of male than female learners, especially in Science, Technology, Engineering and Mathematics (STEM) courses. This, among other factors, contributes to the phenomenon of “the leaky STEM-pipeline” – losing the girls over the course of time before they chose their university career or vocational education. This paper uses an iterative approach to describe how a Large Language Model (LLM), such as ChatGPT can be used to generate interesting and female-oriented learner tasks based on a formerly developed framework of categories in a morphological analysis that helps to design computer science exercises in a structured and consistent way. The LLM is provided with the relevant features of the framework and iteratively asked to generate exercises with respect to girls’ interests and corresponding grading criteria, wrapping the task in a motivating story. The results show how a generative artificial intelligence can be used as a productive approach to the creation of teaching material targeted at a specific audience.

Keywords-STEM; ChatGPT; learners’ task; computer science lessons; morphological analysis; prompt engineering.

I. INTRODUCTION

The lack of girl students in Science, Technology, Engineering and Mathematics (STEM) majors has been a persistent issue that has garnered increasing attention from researchers. One contributing factor to this phenomenon is the absence of self-confidence and a sense of inadequacy in meeting the requirements of universities. This leads to a continuous decline in the number of girls interested in pursuing STEM majors during their school years, often referred to as the “leaky STEM pipeline” [1]. This also counts for Computer Science (CS). One adjustment lever is to systematically promote the competencies needed for studying computer science which can be done by a conscious lesson and task design. The quality of school lessons and tasks in STEM subjects has a major impact on learners regarding their career choices [2]. But learning activities in computer science often follow similar patterns like programming tasks as a kind of “finger exercise”, often designed by male authors or teachers: “The institutionalization of negative representations of women in CS demonstrates a severely unbalanced male influence in the

field. [...] Since CS has long faced a shortage of women, it is unsurprising that the materials reflect predominantly male-centered perspectives. This results in continuous generations of students learning from materials that are rooted in inequality. Subsequently, the continued use and advancement of the field using biased standardized materials, forms a vicious cycle.” [3].

It seems that these tasks sometimes do not appeal to girls who are interested in other exercise contexts [4]. However, adapting exercises and their “contextual stories” is a time-consuming process for teachers. But not only context matters, it is also the way the exercise is done: for example, many girls seem to prefer group work over solitary training – which impacts task planning, distribution of work among team members, documentation, grading, etc. [5]. The overall goal – building up the competencies for a university career – must not be left out of sight.

This paper discusses how to use a Large Language Model (LLM) to design tasks for computer science lessons that target a specific group of learners. Due to the urgent need to better engage girls and to motivate them to pursue careers in STEM, we focus on creating exercises that are responsive to girls’ interests. In the scope of a case study, concrete examples of the use of a LLM – in our case ChatGPT – are shown. We demonstrate current limitations of these technologies and discuss ethical considerations, especially with respect to reinforcing existing biases and gender stereotypes. Our goal is to provide teachers and educational staff with a blueprint for using Artificial Intelligence (AI) technology in a responsible and well-thought manner to create practical exercises in computer science lessons for a certain audience.

To this end, the paper is structured as follows: Section II provides a theoretical foundation on the morphological analysis containing the task dimensions. Section III outlines the necessary steps of planning, creating and testing female-oriented learner activities and shows how teachers can use the framework and/or AI Chatbots as new tools. Section IV presents an iterative approach using different prompting techniques to create new tasks on topics of interest to girls. A case study in Section V illustrates the implementation of this approach and the corresponding results, which can easily be adapted to teachers’ needs. Section VI addresses the problem of gender bias in AI Bots and how it can be used for good to eradicate a real-world bias. The article is summarized in Section VII and Section VIII outlines further research.

II. THEORETICAL FOUNDATION

Competency-based education has proven to be a promising approach in STEM education and teaching [6]-[8]. Therefore in [9], a framework for designing appropriate learners’ tasks for female-responsive computer science lessons was developed with regard to the competencies most needed when starting a university career in the field of computer science. After identifying the most important dimensions needed to describe a task, those dimensions were grouped in a morphological analysis, resulting in a Zwicky box where instructors can choose one characteristic of each dimension to construct an exercise. Table 1 shows the Zwicky Box.

TABLE I. FRAMEWORK FOR DESIGNING FEMALE-RESPONSIVE LEARNING ACTIVITIES IN COMPUTER SCIENCE BASED ON [9]

Learning Activity Dimension	Characteristics/Choices			
Competency	Inventive	Thinking concretely	Analytic/ thinking in an abstract way	(Others)
Class organization	Single learner	Partnership work, two learners	Group-/ teamwork 3-6 learners	-
Gender pairing	Single-gender	Mixed-gender	-	-
Duration	Asynchronous during several lessons based on work packages	Synchronous during one lesson	Synchronous in more than one lesson	-
Task type	Focus on programming an algorithm	Focus on Human Computer Interaction (HCI)-design	Focus on physical construction	(Others)
Tools/ Media	Software tools	Real-world tools, hardware-based	Combination of hardware and software tools	-
Number of possible solutions:	There’s only one correct sample solution	Many sample solutions can be correct	Unlimited number of correct solutions	-
Assessment (frequency)	No support, no interaction	Sporadic support during lessons	Regular support in and between lessons	-
Assessment (method)	Formative during creation	Summative when finished	-	-
Documentation required	Yes	No		
Grading	Yes	No		

Teachers can design tasks by choosing one item from each row and giving appropriate instructions for each

dimension. But they still have to do some “storytelling” to describe a task subject, e.g., the kind of artefact to be developed and a context that is interesting to learners and connects the task to a real-world problem. This is where AI Chatbots based on LLMs come in. They have enough contextual background to come up with interesting exercise concepts, as will be shown in the next sections.

III. DEVELOPING LEARNERS’ TASKS IN A CYCLE

To design and test exercises for computer science lessons, a cycle model can be used. It consists of activities for planning the tasks, letting the learners work on them, providing feedback, grading, and optimizing or updating the task, as shown in Figure 1.

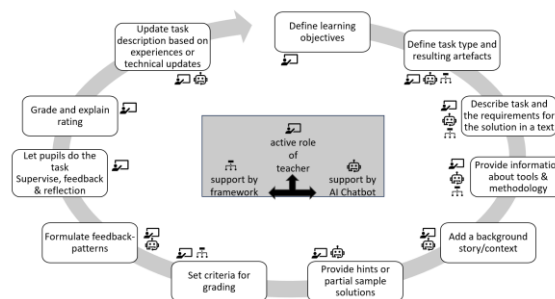


Figure 1. The cycle of exercise development

Many of the activities in this process can be supported by the framework and additionally by an AI Chatbot. Of course, the teachers themselves keep control over all activities and act as coaches for the pupils. (Note: The fact that the learners also will make more and more use of AI Chatbots for solving the tasks is not represented in the figure, since this is subject to negotiation processes in educational institutions on a meta level.) But it is important to state that one shouldn’t solely rely on AI for the whole process: “Using AI-generated teaching materials (e.g., lesson plans) and information (e.g., content for mini-lecture slides) without critical examination can create additional harms for students and negatively impact teaching and learning” [10]. To do so would mean sacrificing a significant amount of valuable knowledge and understanding in the field of education.

IV. AN ITERATIVE APPROACH TO CREATING SUCCESSFUL PROMPTS BASED ON THE FRAMEWORK

We ask ourselves the question: how can the framework described in the previous section be used for automated generation of tasks by a LLM, like ChatGPT? Since LLMs are trained on a broad variety of textual materials, they are able to create answers from different perspectives, change sentiment and tone for a certain audience, and come up with background stories and information. This can be useful when teachers want to tailor their exercises to different ages and genders and provide a textual “hook” for learners to get them interested and started.

To enable a LLM to “create” a unique output, as much context as possible must be provided – so the basic idea is to feed the framework into the LLM first and then ask it to make use of it. In this case, the framework dimensions, as well as the possible items for each dimension, are needed as input. Apart from these details, the quality and usability of the resulting task depends also on the length of output, since the task description has to be specific about the problem, the desired form of results and the criteria for assessment and grading the exercises.

This means that a lot of prompt engineering has to be done before the results become usable [11]. Since prompt engineering often means starting with a rough draft and then fine-tuning the input, as well as asking the system to elaborate on certain aspects, while keeping the context, an iterative approach is needed. The design framework can support the quality of the generated tasks by considering several dimensions at the same time.

Several approaches can be distinguished in prompt engineering: zero-shot prompting, one-shot prompting and few-shot prompting. zero-shot prompting does not provide the AI Bot with any individual examples or additional information. Due to the broad training data base, the bot delivers answers in the correct context. One-shot prompting and n-shot-prompting (where n is any number greater than one) enrich the prompt with one or more examples or demonstrations, allowing the bot to make use of similarities. [12] [13]. Examples for these approaches can be found in the case study in Section V.

In general, the procedure for successfully developing interesting and female-oriented tasks based on the framework and an AI Chatbot can be outlined like this:

1. Preparation: Provide ChatGPT with the problem (i.e., create tasks for computer science lessons that might interest girls) and the framework. This can be done by listing all dimensions and the choices for each dimension. One difficulty is recognizing if two options exclude or impede each other, like solving a large task in a single lesson. Other requirements are that all dimensions are being addressed in the answer which requires a longer output size.
2. Design: Run the prompt and check the resulting suggestions (i.e., the task for computer science lessons) for completeness and usability. If things seem to be missing or do not make sense, add more contextual information and re-run the request. This might lead to more complex task descriptions. Ask ChatGPT to elaborate on the “story” of the task to make more understandable what is expected of the learners and to motivate them.
3. Evaluation: Check if enough information is included so that the task can be solved by the learners. Otherwise, learners might be frustrated at an early stage, worsening the problem of lack of interest in computer science.
4. Creating AI-based sample solutions: Depending on the type of task, AI Chatbots can also be used to generate parts of sample solutions. This refers to programming or interface-design tasks that do not

require hardware to be manipulated (like in robot assembly or circuit design – but here the bot might be asked for part lists and assembly hints). Nevertheless, ChatGPT does (no longer) support the creation of large amounts of code. But it can outline basic structures, object classes or deliver code-snippets, as well as explain code segments in natural language. This can be used to draft the outline of a sample solution for teachers or to generate tutorials for learners.

5. Test: When finished, the resulting artefact is a task description that can be used in a computer science lesson. Before using it, the teacher should check it again for consistency and level of difficulty to ensure the learners can cope with it and have already sufficient theoretical background knowledge to get at least started.
6. Optimize: Since AI Chatbots can save conversations for a longer time, teachers can get back to their last state and re-work existing tasks or even provide feedback about the experiences made in the classroom and add more context.

The following sections demonstrate how exercises can be effectively created using this iterative approach. It presents AI Chatbot prompts that can easily be copied and evaluated or modified to obtain the desired results (e.g., for a certain age group or with respect to geographical or cultural aspects).

V. CASE STUDY: USING CHATGPT TO GENERATE LEARNING ACTIVITIES APPEALING TO GIRLS BASED ON A COMPETENCY-ORIENTED FRAMEWORK IN AN INTERACTIVE APPROACH

A. A one-shot-prompt for simple learner activities

To get a feeling for the “knowledge” of the LLM about potential subjects suitable for girls, the first prompt (a zero-shot prompt) only asked for a task related to user-interface design. This is a task type suitable for pairs or teams, as well as single learners, and can be adapted to a variety of scenarios. Table II shows the prompt and the corresponding result.

TABLE II. EXAMPLE OF ONE-SHOT PROMPT TO GET STARTED

Prompt: “Create a school task for a computer science lesson. The task should appeal to girls and the goal should be to create a user interface. Use up to 500 words.”
Result: Task Title: Design Your Dream Fashion App
Objective: The goal of this task is for students to practice basic computer science principles by creating a user interface for a fashion-app. This activity aims to appeal to girls with an interest in fashion design while also fostering essential computer programming skills.
Overview: [...]
In this task, students will take on the role of app designers, and their assignment is to create a unique fashion app that offers users an interactive and creative experience. The app should have features that girls would enjoy, such as:
1. Personal style profile creation

- 2. Outfit recommendations based on weather and events
- 3. A virtual wardrobe to mix and match outfits
- 4. Fashion news, trends, and [...]

ChatGPT not only came up with a suitable task description, but also with a list of features, but presented no further details or hints how to solve the task.

B. Using the competency-based framework

In the next step, ChatGPT was given a short, but complete description of the ten categories forming the framework plus a hint regarding categories where only one could be chosen. Table III shows the resulting prompt.

TABLE III. PROVIDING THE FRAMEWORK

Prompt: "I will give you a framework for designing interesting tasks for lessons in computer science for school girls aged 12-19. I will list the important properties and the possibilities for each category. You will create a task according to these categories. Make sure the properties you choose fit well together.

The properties are:

- Fostered Competency should be one or more of these: like "thinking concretely", "analytic", "thinking in an abstract way", "being able to visualize", "thinking holistically"
- Class organization should be either individual work, working in pairs or working in groups
- Gender pairing should either be girls-only or mixed groups
- Duration should be either 1 lesson or several lessons
- Task type should be either programming, HCI-design, constructing a switch board, building a robot or any other computer science related work
- Tools and Media should be used, list sample tools to fulfill the task
- Number of possible solutions: either only one solution is correct or the task can have different solution variants or even an infinite number of solutions
- Assessment: the task can be graded or not graded
- Documentation: the task could require a written documentation or just a self-explaining artefact
- Grading: describe if the task can be easily graded concerning certain quality properties

Chose topics that girls are generally interested in. The task description should contain of a maximum of 2000 words and contain suggestions for all ten properties."

The result provided by ChatGPT is shown in Table IV. It is noticeable that it makes use of all given dimensions and comes up with a lot of details concerning requirements, software tools and hints for grading. It even "explains" the goals to the learners in the end.

TABLE IV. A GENERATED TASK

Result: Task: Designing a Mobile App for a Sustainable Fashion Community

Fostered Competency: Thinking holistically, thinking in an abstract way, being able to visualize.

Girls will need to think about the overall concept of a sustainable fashion community, consider abstract ideas related to fashion and sustainability, and visualize the app's design and functionality.

Class Organization: Individual work.

This task can be completed individually, allowing each girl to explore her creativity and problem-solving skills independently.

Gender Pairing: Girls-only.

Girls can work on this task in a girls-only setting, creating a comfortable environment that encourages participation and collaboration among

female students.

Duration: Several lessons.

This task requires multiple lessons to give students enough time to research, plan, design, and prototype their mobile app concept.

Task Type: HCI-design (Human-Computer Interaction design) and programming.

Girls will design the user interface and user experience (HCI-design) of a mobile app for a sustainable fashion community. They will also have the option to implement a basic prototype using programming tools.

Tools and Media:

HCI-design tools: Figma, Adobe XD, Sketch, or any other design software the school has available.

Programming tools: Thinkable, App Inventor, or any other beginner-friendly app development platform.

Research materials: Access to the internet or school library for researching sustainable fashion practices, existing fashion communities, and mobile app design inspiration.

Number of Possible Solutions: Different solution variants.

Girls can come up with different designs and functionalities for their sustainable fashion community app. While there may be common elements, each student's app will reflect her unique perspective and creativity.

Assessment: Graded task.

This task can be graded based on specific criteria, such as the overall concept and functionality of the app, the effectiveness of the user interface and user experience design, and the quality of the prototype (if applicable).

Documentation: Written documentation required.

Girls will need to provide a written documentation explaining the concept, features, and design choices of their app. This documentation will serve as a guide for evaluating their work.

Grading: The task can be easily graded concerning certain quality properties.

The task can be graded based on the clarity and creativity of the app concept, the usability and visual appeal of the user interface, the functionality and effectiveness of the app features, and the completeness and coherence of the written documentation.

Task Description (within the maximum word limit):

In this project, you will have the exciting opportunity to design a mobile app for a sustainable fashion community. The app will provide a platform for fashion enthusiasts to connect, share ideas, and promote sustainable practices in the fashion industry.

You will be working individually on this project, allowing you to explore your creativity and problem-solving skills. Throughout several lessons, you will research, plan, design, and prototype your unique mobile app concept.

To foster your holistic thinking, you will need to consider various aspects of sustainable fashion and abstract ideas related to fashion communities. Visualizing your app's design and functionality will be crucial to creating an engaging and user-friendly experience.

You will utilize HCI-design tools such as Figma, Adobe XD, or Sketch to create the user interface and design the user experience of your app. These tools will enable you to bring your creative ideas to life visually. Additionally, you will have the option to implement a basic prototype of your app using beginner-friendly programming tools like Thinkable or App Inventor.

Your app design can take different forms, reflecting your unique

perspective and creativity. While there may be common elements, such as user profiles, discussion forums, and sustainability resources, you have the freedom to explore various features and functionalities

To find out more about the variety of topics, the prompt was repeated several times, resulting in the following tasks: “Design a Virtual Wellness Retreat”, “Plan an Environmentally-Conscious Community Garden App” and “Designing an Interactive Virtual Reality Experience for Exploring Marine Life”. All tasks contained a similar variety of detail and suggestions for the solution.

C. Adding a storyframe with role models

To get the learners “hooked” on to the task, the next prompt requested a background story – but since no additional information was given on the girls’ age (set to 12-19 in the first prompt), the result very much resembled the style of a children’s book and was not really convincing (see Table V):

TABLE V. CREATING A BACKGROUND STORY

Prompt: “Create a background story for the sustainable fashion app you suggested first. The story should raise the girls’ interest and contain a maximum of 250 words.
Result: Once upon a time, in a world where fashion reigned supreme, a group of passionate young girls embarked on a mission to change the industry for the better. Their names were Mia, Lily, and Ava. [...]

Since role-models are an important catalysator in fostering the girls’ interest in computer science [5], it makes sense to customize the background. Iterative prompting leads to better (but still quite flowery) results, like in the following example (see Table VI):

TABLE VI. ADAPTING THE BACKGROUND STORY TONE AND CONTENT

Prompt: Rewrite the background story using a famous female computer scientist as a role model. Give a few biographical details on the woman and make it sound less pathetic.
Result: In the realm of technology, where innovation thrived, a group of aspiring young girls found themselves captivated by the remarkable story of Dr. Carol Shaw, a pioneering computer scientist who left an indelible mark on the gaming industry. Inspired by her groundbreaking achievements, they set out on a mission to combine their passion for fashion with their love for technology. Dr. Shaw, an exceptional programmer and game designer, blazed trails in a male-dominated field. Her talent and perseverance led her to become one of the first female video game developers, shaping the landscape of interactive entertainment. The girls admired her technical brilliance, creative vision, and the barriers she shattered along her journey.

Depending on the context, the idea of female founders instead of female computer scientists might also help to raise the learners’ motivation (see Table VII):

TABLE VII. ADAPTING THE BACKGROUND STORY TONE AND CONTENT

Prompt: Create a background story for the sustainable fashion app you suggested first. The story should raise the girls’ interest and contain a maximum of 150 words. Use a female founder role model to inspire the girls.
Result: In the bustling city of creativity, a young girl named Maya found herself captivated by the vibrant world of fashion. But she couldn’t ignore the environmental toll the industry was taking. Determined to make a difference, Maya sought inspiration from a remarkable role model - Stella

McCartney.
Driven by Stella’s fearless pursuit of sustainable fashion, [...]

This is a good example how iterative prompting can lead to better results. Since ChatGPT allows reference to previously processed prompts and increasingly more memory is available for this, many iterations can be performed. Moreover, users can store pre-defined instructions. This can be used for prompts describing the framework instead of repeating the details every time. When the task creation is finished, the development of a sample solution or additional hints can be generated. Table VIII shows an example for an Android code snippet with explanations generated from a simple prompt in natural language:

TABLE VIII. CREATING PARTS OF A SAMPLE SOLUTION

Prompt: “Create a code snippet for a function in the sustainable fashion app registering a new user.”
Result: Certainly! Here’s a code snippet for a function that registers a new user in a sustainable fashion app: // Assuming you have a User class to represent the user data public class User { private String username; private String email; private String password; // Constructor, getters, and setters // ... } // RegistrationActivity.java import android.os.Bundle; import android.support.v7.app.AppCompatActivity; import android.view.View; import android.widget.Button; import android.widget.EditText; import android.widget.Toast; public class RegistrationActivity extends AppCompatActivity { private EditText usernameEditText; private EditText emailEditText; private EditText passwordEditText; private Button registerButton; @Override protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); } [...] Please note that this code snippet assumes you have an activity_registration.xml layout file that contains the necessary UI elements such as EditText and Button to capture user registration information. Additionally, you would need to implement the logic to save the user data to a database or an API, which is beyond the scope of this code snippet. Remember to adapt the code to fit your specific application structure and requirements.

Thus, sample solution code generation becomes less time-consuming.

VI. MAKING USE OF GENDER STEREOTYPES “FOR GOOD”?

Generative AI models are often accused of containing certain biases based on the data they have been trained with

[14]. Especially gender bias is a widely discussed topic [15] and teachers using LLMs to generate learning material need to be highly aware of it. On the other hand, the approach taken in this paper actively makes use of the fact that a LLM has information about “typical” interests of boys and girls. But in this case, the original problem stemmed from a bias in the real world, namely that the exercises developed by teachers were too often dealing with topics that were more attractive to boys than to girls. Our approach actively uses the AI models’ gender bias in the short run to fight the leaky STEM-pipeline in the long run, because it is important to get the girl students “hooked” on the technical topics. There’s evidence that girls solving gender-stereotypical mathematical tasks can outperform girls in a control group [16]. Therefore, if this leads to more interesting learners’ tasks and in result to more active participation of girls in computer science lessons, it is a legitimate use of such an AI chatbot.

VII. DISCUSSION AND CONCLUSION

An AI Chatbot, such as ChatGPT, can be a useful tool in creating learners’ tasks for computer science lessons, especially with a focus on content that female learners are interested in. But to successfully create exercises, a framework is needed to ensure that certain properties are considered and that the task descriptions are comparable. This can be done by providing categories of a framework to the AI Chatbot. Prompt engineering can be steered in an iterative way, providing more and more context or using storytelling to create more appealing exercises. The task descriptions can even be enriched by using role-models from computer science or the business world. It is also possible to generate sample solution or hints, depending on the type of task. Evaluation and grading should remain in the hands of the teachers, since these more complex tasks that require group work can – at least at the present time – not be automatically done. But if teachers give written feedback to learners’ artefacts, an AI Bot can be used to fine-tune the mode of address and to help (re-)formulate objective criticism.

Due to the current LLMs’ large scopes of texts and facts, creating interesting and innovative exercises by this approach can save teachers lots of time and effort and cover many stages of the learning-material lifecycle. Student teachers should therefore be familiarized with such possibilities of task creation already during their studies and encouraged to use them. In this respect, the active use of LLMs should be included in the teacher training curricula.

VIII. OUTLOOK

Teachers can use the case study to gather some experience in AI-supported task design and test the results with regard to a rise in motivation especially for female learners. The next steps could then be to use the approach described above to generate female-, as well as male-oriented lesson tasks and test whether girls and boys prefer

one over the other using quantitative research approaches. A suitable study design could use an A/B-Testing approach to let students decide which type of task they would prefer to work on. Here, the reasons could also be asked in a questionnaire. Parallel to this, the approach should be tested within a larger third-party funded research project by a group of female computer science professors from the IU International University of Applied Sciences (including the authors), which is currently in the application phase. It would consist of a series of interventions and a ring lecture for female students on IT tasks at a large number of German secondary schools.

As long as the factual and methodological knowledge is successfully transported equally in these exercises (which can also be evaluated quantitatively), nothing speaks against this idea of structured and automated exercise-design.

REFERENCES

- [1] J. D. Speer, “Bye Bye Ms. American Sci: Women and the Leaky STEM Pipeline,” *Economics of Education Review*, Volume 93, 102371, 2023, <https://doi.org/10.1016/j.econedurev.2023.102371>
- [2] C. Hess et al., “How does school lesson design contribute to girls’ perception of their future-readiness for STEM majors? – An empirical study using critical incidents technique”, *The 4th International Conference on Teaching and Education Management (ICTEM 2023)*, Berlin, Germany, June 2023 (in press).
- [3] P. Medel and V. Pournaghshband, “Eliminating gender bias in computer science education materials” [Paper presentation], *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, New York, NY: Association for Computing Machinery, pp. 411–416, March 2017.
- [4] A. Steffen et al., “An Exploration of Critical Incidents Impacting Female Students’ Attitude Towards STEM Subjects”, Vol. 6 No. 1: *Proceedings of the 6th International Conference on Gender Research*, 2023, pp. 215-223, <https://doi.org/10.34190/icgr.6.1.994>.
- [5] L. Happe, B. Buhnova, A. Koziolok, and I. Wagner, “Effective measures to foster girls’ interest in secondary computer science education”, *Education and Information Technologies*, Vol. 26(3), pp. 2811–2829, 2021, <https://doi.org/10.1007/s10639-020-10379-x>
- [6] CC2020 Task Force, “Computing Curricula 2020: Paradigms for Global Computing Education,” *ACM*, 2020, <https://doi.org/10.1145/3467967>.
- [7] V. Thurner, A. Bottcher, and A. Kamper, “Identifying Base Competencies as Prerequisites for Software Engineering Education,” *EDUCON – 2014 IEEE Global Engineering Education Conference*, pp. 1069–1076. <https://doi.org/10.1109/EDUCON.2014.6826240>.
- [8] D. Zehetmeier et al., “Self-assessment of freshmen students’ base competencies,” *EDUCON – 2014 IEEE Global Engineering Education Conference*, pp. 429–438. <https://doi.org/10.1109/EDUCON.2014.6826130>, 2014.
- [9] S. Kunz and C. Hess, “How to catch’em all: Designing attractive learning activities for girls in computer science. A systematic framework”, *The 4th International Conference on Teaching and Education Management (ICTEM 2023)*, Berlin, Germany, June 2023 (in press).
- [10] T. Trust, J. Whalen, and C. Mouza, “Editorial: ChatGPT: Challenges, opportunities, and implications for teacher education”, *Contemporary Issues in Technology and Teacher Education*, vol. 23(1), pp. 1-23, 2023.
- [11] T. Teubner, C. M. Flath, C. Weinhardt, W. van der Aalst, and O. Hinz, „Welcome to the Era of ChatGPT et al.,” *Business & Information Systems Engineering*, vol. 65(2), 2023, pp. 95–101, <https://doi.org/10.1007/s12599-023-00795-x>.
- [12] DAIR.AI. Prompt Engineering Guide: Prompting Techniques <https://www.promptingguide.ai/techniques>, 2023. [retrieved: 08, 2023]

- [13] J. White et al., "A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT," 2023, arXiv preprint available at <http://arxiv.org/pdf/2302.11382v1>. [retrieved: 08, 2023]
- [14] C. O'Neil, "Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy", Penguin Books, 2017.
- [15] A. Nadeem, B. Abedin and O. Marjanovic, "Gender Bias in AI: A Review of Contributing Factors and Mitigating Strategies", *ACIS 2020 Proceedings*. 27., 2020, <https://aisel.aisnet.org/acis2020/27>. [retrieved: 08, 2023]
- [16] J. M. Hermann and R. Vollmeyer, "Gender stereotypes: implicit threat to performance or boost for motivational aspects in primary school?", *Soc Psychol Educ* vol. 25, pp. 349–369, 2022, <https://doi.org/10.1007/s11218-022-09693-8>.