LINKING AUGMENTED REALITY WITH PEER TUTORING IN VOCATIONAL LEARNING ENVIRONMENTS: A MULTI-AGENT-BASED APPROACH

Research in Progress

Henning Gösling, DFKI GmbH, Osnabrück, Germany, henning.goesling@dfki.de

Tobias Dreesbach, DFKI GmbH, Osnabrück, Germany, tobias.dreesbach@dfki.de

Jannis Vogel, University of Osnabrück, Osnabrück, Germany, jannis.vogel@uos.de

Enrico Kochon, DFKI GmbH, Osnabrück, Germany, enrico.kochon@dfki.de

Abstract

During the elicitation of vocational training processes in a consortium research project, it became clear that augmented reality (AR) is a useful learning environment. However, using AR glasses may isolate trainees in training workshops and reduce the level of collaboration. In this study, we address these potential adverse side effects with peer tutoring. More exactly, we want to initiate tutor-tutee pairs among trainees during AR-based vocational training processes. We derive design features for a multi-agent-based approach to build tutor-tutee pairs based on concepts from the literature, a technical perspective, and the consortium within a design science research approach. To the best of our knowledge, we are the first to align AR with peer tutoring. With our first results, we provide a novel IT artifact in the realm of computer-supported collaborative learning in the information systems discipline. Future work can build upon these results even outside the scope of AR.

Keywords: Peer Tutoring, Augmented Reality, Multi-Agent System, Design Features.

1 Introduction and Motivation

Digital technologies usually undergo phases until they manifest as digital innovations in products and services, processes, or business models (Wiesböck and Hess, 2020). Augmented reality (AR) has received increasing attention as a digital technology in science and practice thanks to its continuous development and high market maturity. AR allows displaying virtual objects in the user's field of view and combines reality with visual information (Azuma, 1997). Skill training and service support systems are two of the well-known use cases of AR (Kohn and Harborth, 2018). Especially head-mounted displays (HMDs), such as the Microsoft HoloLens 2, are favorable AR devices in the professional environment as they allow hands-free interaction with AR and provide a rich 3D medium that is not separate from reality (Kammler et al., 2019).

In view of these positive characteristics of AR, in a three-year consortium research project, we strive to implement AR in the vocational training processes of two companies and two vocational schools as a digital innovation. The two companies involved in the project are a manufacturer of semi-finished copper products and a manufacturer of refrigeration, ventilation, and air-conditioning products. The two vocational schools are included to take the dual training in Germany into account. Given the increased complexity of products and machines and the different learning needs of trainees, these companies and schools aim for individual AR learning scenarios that integrate various didactic elements (for the methodology approach, see Dreesbach et al., 2021) without disturbing the processes of vocational learning itself.

The design science research (DSR) project was started with an initial requirement analysis. In a training workshop at one of the project's partners, the trainees work on the assembly and wiring of a changeover circuit at their workstations. Peer tutoring takes place between the trainees on a regular basis. Hence, if any trainee has a question or is at a loss, not only the instructor but also the other trainees at neighboring workstations are available for support. Peer tutoring (also known as peer teaching) is "a one-to-one teaching process in which the tutor is of the same general academic status as the tutee" (Cohen, 1986, p. 175). It is an established collaborative learning approach (Smith and Macgregor, 1992) that has positive effects thanks to the active involvement of both tutors and tutees in the learning situation (Goldschmid and Goldschmid, 1976; Cohen, 1986). Therefore, since AR is becoming an interesting medium for collaboration (Brockmann et al., 2013), we noticed an opportunity to align it with peer tutoring to improve the collaboration and learning scenario. However, despite the positive effects of AR as a supportive technology in learning and collaboration, we discover negative side effects on a social level as a result of the technology-use in the learning scenario itself. Drawing on the findings of Harborth (2019) in a qualitative study, AR users are concerned about a reduction in direct communication and mishandling. These insights are supported by Miller et al. (2019, p. 21), who figured out in a laboratory study that using AR glasses "hinder[s] communication as they prevent[s] eye-contact" and that the social presence changes between users wearing AR glasses and non-AR users. Moreover, Dunleavy et al. (2009) described the effect of students losing track of the real environment because they are very immersed in AR. Thus, we counteract these issues mentioned above and foster collaboration among trainees with a problem-centered DSR approach that is aligned with the following research question (RQ):

RQ: *How can a system linking AR with peer tutoring in vocational learning environments be designed to improve social presence and collaboration?*

To combine AR with peer tutoring and answer the RQ, we used a multi-agent-based approach on a technological level and called the artifact a multi-agent-based AR peer tutoring system (MA-AR-PTS). Software agents are responsible for the dynamic allocation of tutoring requests and, thus, for the building of tutor-tutee pairs. Having autonomous software agents as modular components that are reactive to changes in their environment and are specialized in specific functions allows our artifact to become context-sensitive, robust, and adaptable (Franklin and Graesser, 1997; Celaya et al., 2009). We suggest that the designed artifact reduces negative social effects due to the use of AR, promotes the use of peer tutoring and collaborative learning, reduces the workload of instructors, and adds more variety in selecting tutors.

This research-in-progress paper contributes to the body of knowledge of computer-supported collaborative learning (CSCL) by defining 15 design features (DFs) describing the technical implementation. Therefore, we provide prescriptive design knowledge that is scarce for CSCL artifacts focusing on peer tutoring (Magnisalis et al., 2011). We also provide design knowledge for AR-based collaborative systems. According to the taxonomy by Brockmann et al. (2013), our artifact can be classified as a co-located, synchronous, mobile, object-visualizing, multi-role, and HMD-based system.

Our research is grounded in the social presence theory (SPT), which was put forth by Short et al. (1976), who proposed that a communication medium has its own social presence to transmit social cues. Over time, the SPT has been applied in various computer-mediated communication (CMC) systems (cf. Biocca et al., 2003). In general, socio-psychological questions such as "How well did one person feel connected to another through an interface?" (Biocca et al., 2003, p. 459), fall within the scope of the SPT. In future work, we intend to evaluate the social presence and CMC aspects with our artifact and in vocational learning scenarios, following the call for research by Gunawardena and Zittle (1997).

In line with the work of Gregor and Hevner (2013) on DSR schemas, we introduce fundamentals in Section 2, followed by our research approach in Section 3. Then, we describe and demonstrate the MA-AR-PTS in Section 4. Finally, in Section 5, as research in progress, we outline future work in terms of evaluation and artifact development.

2 Fundamentals

Collaborative learning is defined as "a situation in which two or more people learn or attempt to learn something together" (Dillenbourg, 1999, p. 1). Hence, three elements (i.e., a group of people, available learning material, and particular forms of interaction among people that can trigger learning mechanisms) define collaborative learning situations (Dillenbourg, 1999). When applying collaborative learning, the teacher usually acts as the facilitator, while the students must take responsibility for learning. Consequently, students are more stimulated to reflect on their assumptions and develop social and team skills (Kreijns et al., 2003). Peer tutoring is a collaborative learning approach that ensures the tutee's active involvement in the learning situation while the tutor is rehearsing the learned material, which facilitates long-term retention and a more comprehensive and integrated understanding. Moreover, it involves the use of a peer as a role model of a student, demonstrating learning skills such as concentrating on the material, attending to another person, organizing work habits, and asking questions (Goldschmid and Goldschmid, 1976; Cohen, 1986). Implementation of peer tutoring varies depending on several aspects, such as the selection procedure of tutors and tutees, the way tutors and tutees are matching, or the choice of learning material. Generally, tutors and tutees are selected to improve their academic levels, to improve collaboration and to create social contacts. When the focus is on performance, tutors may be selected for their communication and tutoring skills, their ability to create trust, motivation, or high and influential social status (Cohen, 1986). However, in the case of unstructured implementation in a classroom setting, the most accessible tutors are the students' immediate neighbors who might not be qualified to provide help (Dong and Hwang, 2012).

CSCL is a research field that focuses on how IT artifacts can enhance those interactions among students that should trigger learning mechanisms. A CSCL artifact may support learning group composition, community ethos, teacher-student interaction, peer interaction, task structuring, scaffolding, meaning-making, and knowledge building (Resta and Laferrière, 2007). AR devices are regularly integrated into CSCL artifacts (Phon et al., 2014).

3 Research Approach

Building upon the mentioned problem aspects with AR in vocational learning scenarios that motivated our research, we follow the well-established DSR method for developing problem-oriented IT artifacts proposed by Peffers et al. (2007). Figure 1 shows the research approach, clarifying both the completed and the ongoing research. This paper's scope includes the motivation of our research, objectives, artifact development, and demonstration of the MA-AR-PTS.

In the first step of our research project, we elicited four processes at two vocational schools, a production workshop, and a training workshop. During the process elicitation in the training workshop, the potential issue of trainees isolating themselves with AR glasses from the other trainees in the workshop was identified. Therefore, we brought up the idea of a peer tutoring functionality that allows trainees in an AR-based vocational learning environment to initiate peer tutoring during an exercise that motivated our research and formulated objectives (cf. Section 1). Next, to derive design knowledge from the literature, we conducted a systematic literature review on relevant CSCL artifacts following Cooper (1988), Webster and Watson (2002), and Dybå and Dingsøyr (2008). We delved into the SpringerLink, ScienceDirect, AISeL, Wiley, IEEE, ACM, and JSTOR research databases, followed by a backward reference search. The search term used to identify relevant CSCL artifacts was a combination of the words "computer supported collaborative learning" and ("peer tutoring" or "peer teaching"). We evaluated research papers in English, analyzed the results qualitatively, and identified two publications with relevant content, i.e., publications presenting IT artifacts for initializing peer tutoring in learning environments. The first was by Westera (2007), who presented an artifact for initiating peer tutoring among students in an online learning environment. The second was by Dong and Hwang (2012), whose IT artifact included a funtionality for building tutor-tutee pairs in a classroom. However, we did not come across any artifact for building tutor-tutee pairs in an AR-based learning environment.

On the basis of the process elicitation in the consortium research, the technical perspective of software agents, and the systematic literature review, we defined 15 DFs for a multi-agent system that links AR with peer tutoring (cf. Section 4.1). DFs are "specific ways to implement a design principle in an actual artifact" (Meth et al., 2015, p. 807). By providing these DFs, we deliver design knowledge with a concrete technical instantiation of the multi-agent system (cf. Section 4.2) by prototyping (Riege et al., 2009). We follow a bottom-up approach in analogy with Feine et al. (2020), who first proposed DFs and abstracted them into higher-order design principles (DPs). Future work comprises the latter step to aggregate our DFs into DPs for a multi-agent-based approach integrating peer tutoring in AR-based learning environments, following the schema for DPs by Gregor et al. (2020), and to work with the partners from the consortium to evaluate these DPs. We present the MA-AR-PTS based on a small-scale example and demonstrate the application with screenshots (cf. Section 4.3).

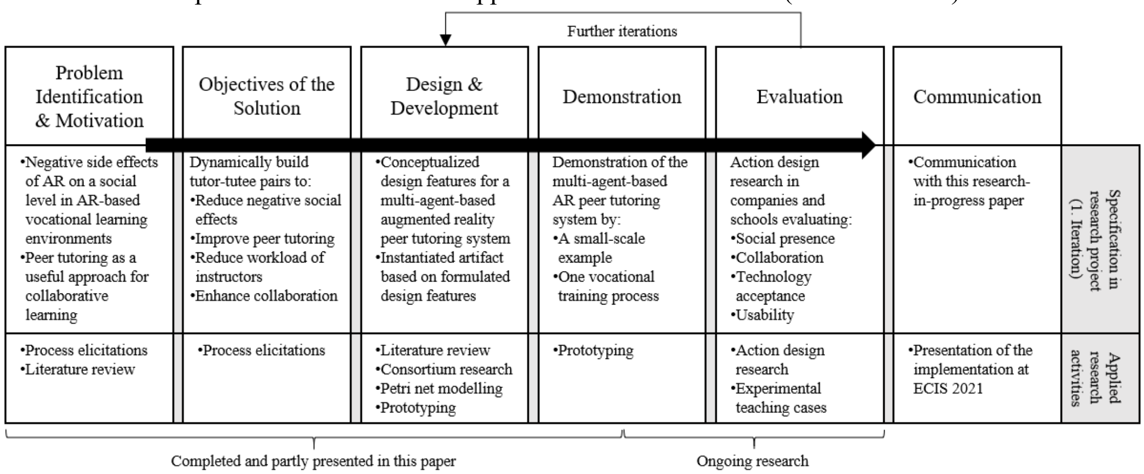


Figure 1. Problem-oriented Design Science Research in accordance with Peffers et al. (2007).

Ongoing research includes an evaluation to derive descriptive knowledge regarding the implemented MA-AR-PTS. Therefore, we apply action design research (ADR) (Sein et al., 2011) following the observation-centered ADR entry point (Mullarkey and Hevner, 2019), aiming to intervene in the vocational training processes of the companies and schools and to evaluate changes regarding social presence, collaboration, technology acceptance, and usability, which overall affect the learning scenario. We suggest a positive effect of our artifact to overcome barriers of AR use and foster collaboration and social contact. Hence, we make the following proposition.

Prop.: *The social presence resulting from the linking of AR and peer tutoring will have a positive impact on collaboration and learning.*

Future work will include evaluation studies examining the proposition and a reflection of the evaluation results on our instantiated artifact. Moreover, further iterations are needed to strengthen the system design considering the various stakeholders in the consortium.

4 Multi-Agent-Based Augmented Reality Peer Tutoring System

In this section, we conceptualize the MA-AR-PTS based on DFs (cf. Section 4.1) and then describe an initial implementation of our system (cf. Section 4.2). The basic workings of this implementation are demonstrated using a small-scale example with HoloLens 2 devices (cf. Section 4.3).

4.1 Underlying Design Features for Conceptualization

On the basis of the process elicitation and in line with the related work found in the systematic literature review (Westera, 2007; Dong and Hwang, 2012), we require our system to initiate tutor-tutee pairs among the trainees during ongoing exercises in an AR-based vocational learning environment. Figure 2 depicts the process of how a tutor is allocated after a trainee triggers a tutoring request in AR. We represent each trainee by a “trainee agent” and the instructor by an “instructor agent” because such a multi-agent-based approach ensures that the tutor allocation procedure is reactive to the current context of the trainees and instructor in the training workshop (*DF1*) (Franklin and Graesser, 1997). Having software agents as modular components also allows our artifact to become adaptable and robust (*DF2*) (Celaya et al., 2009).

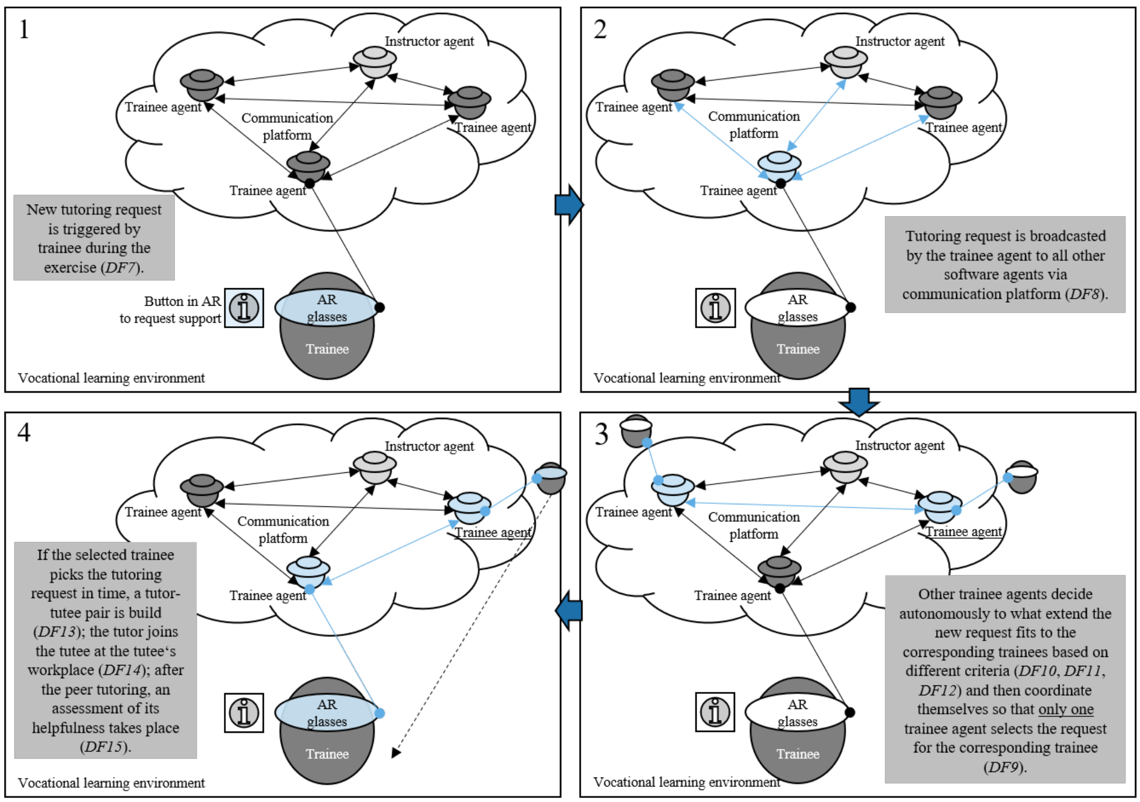
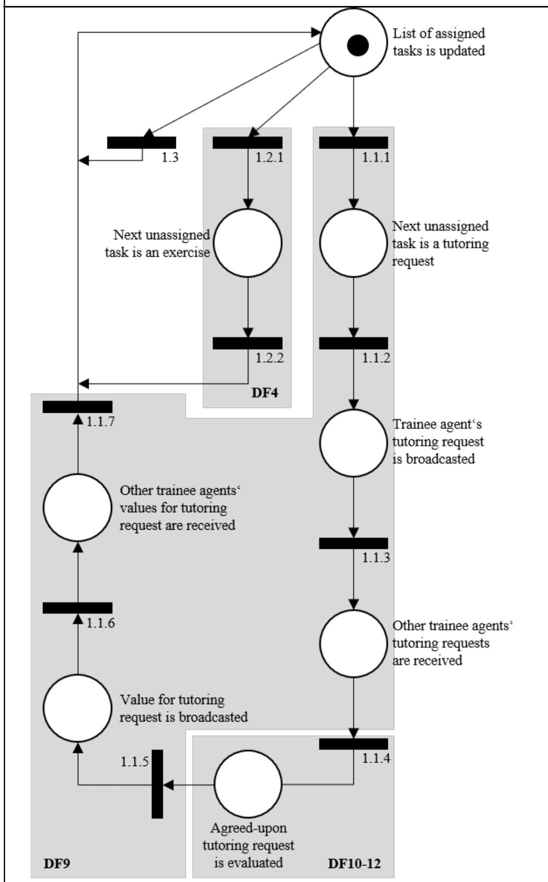
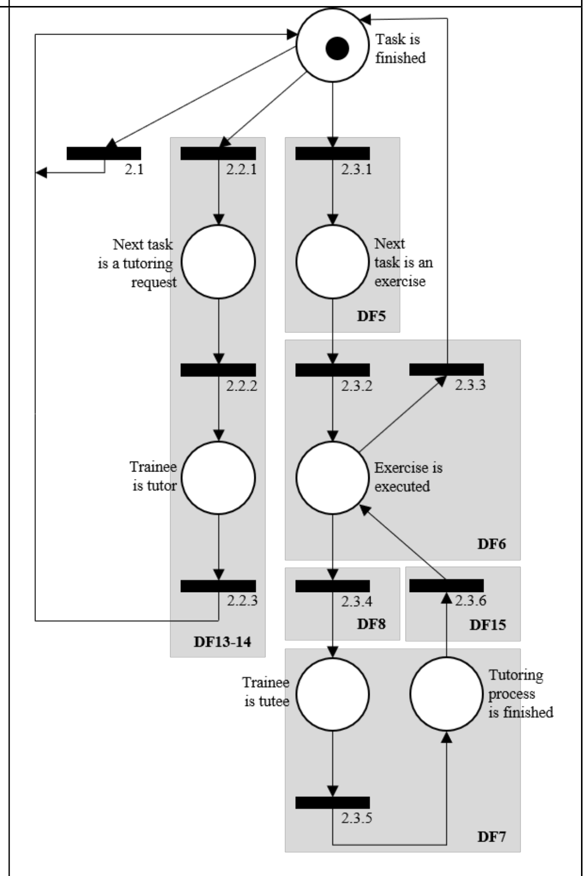


Figure 2. Multi-agent-based tutor allocation process in an AR-based vocational learning environment.

The instructor agent should support the instructor to broadcast exercises into the AR-based vocational learning environment (*DF3*). These exercises should be picked autonomously by the trainee agents for the corresponding trainee (*DF4*) and should be shown in the trainee’s field of view (*DF5*). The trainee should be able to select and end an exercise using the AR glasses (*DF6*). During the exercise, a trainee can become a tutee after she or he releases a tutoring request, also using the AR glasses (*DF7*). The corresponding trainee agent should broadcast the tutoring request via a communication platform to all the other active trainee agents (*DF8*). The other trainee agents should then decide autonomously which trainee becomes the tutor for this request (*DF9*). It is important that the trainee agents achieve a fair workload distribution among all potential tutors. Otherwise, the system will fail because it involves only a sub-group of highly qualified trainees as tutors (*DF10*) (Westera, 2007). Moreover, the MA-AR-PTS should select a competent tutor for a tutoring request according to the trainees’ accomplished learning paths and peer tutoring performances (*DF11*) (Westera, 2007). It should also take the current learning context (exercise, status, location, etc.) of all trainees into account (*DF12*). For example, a tutoring request should be inserted into the schedule of the selected tutor so that her or his current exercises are at least not disrupted and at best rehearsed in a meaningful way. The trainee agent representing the selected tutor should update the current schedule of tasks, which now consists of the regular, unfulfilled exercises that have already been released by the instructor agent and the new tutoring request. If the selected tutor does not pick the tutoring request within a certain time limit, the request should fail. However, if the selected trainee picks the tutoring request as the next task within the time limit, the tutor-tutee couple is finally built (*DF13*). The tutor should join the tutee at her or his workplace so that the peer tutoring can start (*DF14*) (Dong and Hwang, 2012). After the peer tutoring is finished, its helpfulness should be reviewed by the tutee (*DF15*) (Westera, 2007; Dong and Hwang, 2012).

Petri net for the assessment routine of a trainee agent	**Petri net for the execution routine of a trainee agent**





Transitions in the assessment routine	**Transitions in the execution routine**
1.1.1: Checking if the next unassigned task is a tutoring request	2.1: Checking whether there is no task to be executed
1.1.2: Broadcasting the agent's state	2.2.1: Checking whether there is a tutoring request to be executed
1.1.3: Receiving the other trainee agents' states	2.2.2: Answering the tutoring request
1.1.4: Evaluating the tutoring request	2.2.3: Stopping the tutoring request
1.1.5: Broadcasting the value for the tutoring request	2.3.1: Checking whether there is an exercise to be executed
1.1.6: Receiving the other trainee agents' results	2.3.2: Starting the exercise
1.1.7: Deciding on the tutoring request	2.3.3: Finishing the exercise
1.2.1: Checking whether the next unassigned task is an exercise	2.3.4: Releasing a tutoring request via broadcast
1.2.2: Assigning an exercise to the trainee's schedule	2.3.5: Waiting for the tutor to finish the peer tutoring process
1.3: Checking whether there is no task to be assigned	2.3.6: Rating of the tutoring performance

Figure 3. Petri nets describing the assessment and execution routines of a trainee agent.

On the basis of the specified DFs, we conceptualized the trainee agent, which is the core component of our MA-AR-PTS covering *DF4-15*. The instructor agent only covers *DF3,* whereas both types of software agents cover *DF1* and *DF2*. We modeled the trainee agent using Petri nets as suggested by Celaya et al. (2009). Generally, a Petri net is a directed graph consisting of two kinds of nodes, called places and transitions, connected by arcs (Murata, 1989). Each trainee agent is made up of two routines and each routine is modeled by a separate Petri net (Figure 3): one for the assessment of incoming exercises and tutoring requests and the other for the execution of activities when being a trainee (i.e., doing an exercise), a tutor (i.e., supporting an exercise), or a tutee (i.e., being supported during an exercise).

All trainee agents start the assessment routine (see Figure 3) by checking in transition 1.1.1 whether there are new tutoring requests to be assigned after being triggered by other trainees in the workshop. Assessing a newly arrived, unassigned tutoring request is performed by the collective intelligence of all active trainee agents. This collective intelligence synchronizes the assessment among all trainee agents. After a tutoring request is received, all trainee agents must make sure that they are in the same state. More specifically, all trainee agents should agree upon a tutoring request by selecting the tutoring request that is evaluated by the majority of the trainee agents, meaning that the tutoring request on evaluation can change during the assessment routine (in transition 1.1.4 to be precise). The trainee agents then calculate the utility value of the tutoring request according to the workload, competence, and learning context of the corresponding trainee. Afterward, the trainee agents wait until every active trainee agent has finished the calculation and has shared its value. According to its value and those of the other trainee agents, each trainee agent can make an autonomous assignment decision: only if it has the optimal value among all trainee agents will it pick the new tutoring request. If two or more trainee agents calculate the same value, the ID of a trainee agent is used as the final decision criterion. The path of the execution routine (see Figure 3) depends on the trainee's decision using the user interface (UI) of the AR glasses. The trainee decides either to answer a certain tutoring request or to perform a certain exercise out of the current task list. Consequently, the execution routine goes either into the path starting with transition 2.2.1 or into that starting with transition 2.3.1.

4.2 Implementation

We developed each software agent as a console application in C# with Microsoft Visual Studio 2019. The instructor agent is only used to create and broadcast new exercises (*DF3*). Each trainee agent is made up of two independent state machines, and each state machine is implemented using the Appccelerate library (Marbach et al., 2020). One state machine performs the assessment routine (*DF4*, *DF9-12*), and the other performs the execution routine (*DF5-8*, *DF13-15*). The trainee agents are loosely coupled with each other, with the instructor agent, and with the corresponding AR glasses by the MQTT protocol (Hunkeler et al., 2008). We run all trainee agents and the Mosquitto MQTT broker on a standard laptop. Trainee agents write their learning path in a Neo4j graph database that also runs on the same laptop. According to the completed learning path, each trainee agent evaluates the incoming tutoring requests. In the first implementation, this value simply represents the number of times a trainee has performed an exercise for which support is needed, plus the number of times a trainee has performed a tutoring request for this exercise. This simple decision rule increases the tutoring quality but does not distribute the tutoring workload evenly among all active trainee agents and does not take the current learning context of the trainees into account, which is why we will focus on enhancing this assessment routine in the further iterations of our research. We use HoloLens 2 devices as AR glasses. For the UI on the HoloLens 2, we developed a Unity application in C#.

4.3 Demonstration

In this section, we present a small-scale example to demonstrate the workings of the MA-AR-PTS (cf. Table 1). An instructor agent and three trainee agents are initialized. The trainee agents for Trainee 1 and Trainee 2 are each connected with a HoloLens 2 glasses, while the trainee agent for Trainee 3 is connected with a HoloLens 2 simulator running on a PC. At the beginning of the demonstration, there are no learning paths stored in the Neo4j database. After the four software agents' initializations, an instructor releases three exercises using the UI of the instructor agent. The trainee agents show these three exercises in the UI of their corresponding HoloLens 2 device. Trainee 2 finishes Exercise 1, and the Neo4j database is updated. Trainee 1 sees all three exercises in the UI and starts with Exercise 1 (screenshot A in Table 1), triggers a tutoring request during this exercise (B), and waits for a tutor. This tutoring request, after being analyzed by the other trainee agents representing Trainee 2 and Trainee 3, pops up in the task list of Trainee 2 (C). After selecting, answering and ending this tutoring request (D), Trainee 1 can rate the performance of Trainee 2 as the tutor for Exercise 1 (E). This rating is saved in the Neo4j database. The procedure of Trainee 1 receiving support from Trainee 2 is shown in Table 1 using screenshots of the UIs of the HoloLens 2 glasses for Trainee 1 and Trainee 2.

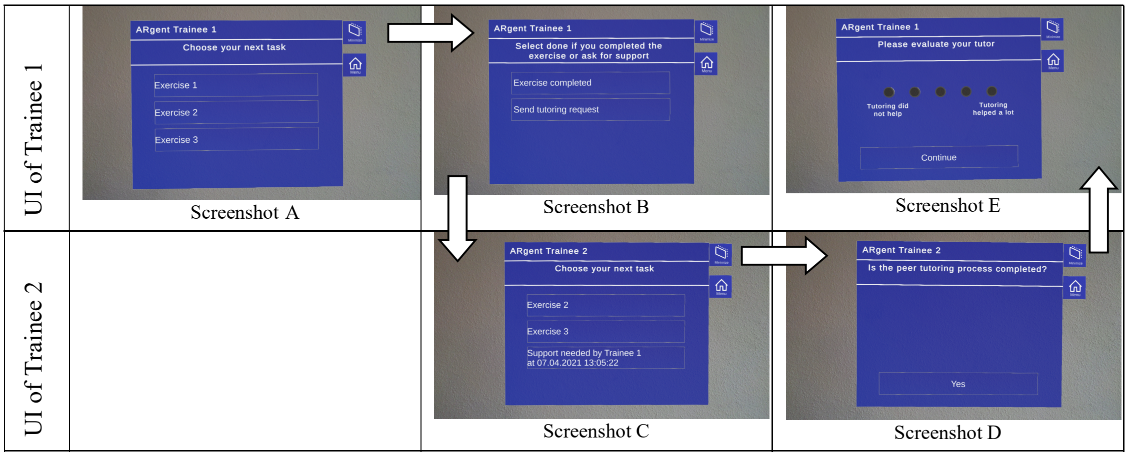


Table 1. Trainee 1 building a tutor-tutee couple with Trainee 2 using the MA-AR-PTS.

5 Conclusion and Outlook

In this research-in-progress paper, we present a multi-agent-based approach to link AR with peer tutoring in vocational training processes. We originally explored the use of AR in vocational learning environments in a consortium research project. We observed during the process elicitation that the use of AR glasses may negatively influence social cues, which affects the learning scenario. Therefore, we address these issues (i.e., isolation due to immersion, limited eye-contact, reduction in direct communication and collaboration) using a problem-oriented DSR approach. On the basis of the literature, vocational training processes, and technical concepts, we defined 15 DFs to inform researchers and practitioners of how such a system can be technically designed. Thereby, we provided prescriptive design knowledge in the shape of a concept, DFs, Petri nets, and a demonstration of the artifact. To the best of our knowledge, our concept is a novel one in the field of CSCL.

Future work will include testing our formulated proposition in evaluation studies and validating whether social presence changes as a result of our system use. Moreover, we aspire to derive descriptive knowledge regarding our instantiated artifact within ADR by integrating consortium partners (i.e., two companies and two vocational schools). These results will be reflected into higher-order DPs for multi-agent-based peer tutoring systems. We believe that these results can be transferred to other types of CMC, such as online classes. Our results are even relevant for firms offering collaborative e-learning solutions that do not use AR. From a scientific perspective, we further contribute, mainly in future work, to the SPT with insights into the impact of new types of CMC (i.e., AR) and how the medium can be designed in a useful way for vocational training processes.

Acknowledgements: This work is part of the research project AdEPT (FKZ: 01PV18008E). This project is funded by the German Federal Ministry of Education and Research (BMBF).

References

Azuma, R. T. (1997) "A Survey of Augmented Reality." *Presence: Teleoperators and Virtual Environments* 6 (4), 355–385.

Biocca, F., Harms, C. and Burgoon, J. K. (2003) "Toward a More Robust Theory and Measure of Social Presence: Review and Suggested Criteria." *Presence: Teleoperators and Virtual Environments* 12 (5), 456–480.

Brockmann, T., Krüger N., Stieglitz S. and Bohlsen I. 2013 A Framework for Collaborative Augmented Reality Applications. *In:* Shim, J., Hwang, Y. & Petter, S. (eds) *Americas Conference on Information Systems*. Chicago, USA.

Celaya, J. R., Desrochers, A. A. and Graves, R. J. 2009. Modeling and analysis of multi-agent systems using petri nets. *In: IEEE International Conference on Systems, Man and Cybernetics*. Montreal, QC, Canada.

Cohen, J. (1986) "Theoretical considerations of peer tutoring." *Psychology in the Schools* 23 (2), 175–186.

Cooper, H. M. (1988) "Organizing knowledge syntheses: A taxonomy of literature reviews." *Knowledge in Society* 1 (1), 104.

Dillenbourg, P. 1999. Introduction: What do you mean by "collaborative learning"? *In:* Dillenbourg, Pierre (ed.) *Collaborative Learning: Cognitive and Computational Approaches*. Oxford: Elsevier.

Dong, J.-J. and Hwang, W.-Y. (2012) "Study to minimize learning progress differences in software learning class using PLITAZ system." *Educational Technology Research and Development* 60 (3), 501–527.

Dreesbach, T., Berg M., Gösling H., Walter T., Thomas O. and Knopf J. 2021. A Methodology to enhance Learning Processes with Augmented Reality Glasses. *In:* Ahlemann, F., Schütte, R & Stieglitz, S. (eds) *International Conference on Wirtschaftsinformatik*. Virtual Conference.

Dunleavy, M., Dede, C. and Mitchell, R. (2009) "Affordances and Limitations of Immersive Participatory Augmented Reality Simulations for Teaching and Learning." *Journal of Science Education and Technology* 18 (1), 7–22.

Dybå, T. and Dingsøyr, T. (2008) "Empirical studies of agile software development: A systematic review." *Information and Software Technology* 50 (9), 833–859.

Feine, J., Adam M., Benke I., Maedche A. and Benlian A. 2020. Exploring Design Principles for Enterprise Chatbots: An Analytic Hierarchy Process Study. *In:* Hofmann, S., Müller, O. & Rossi, M. (eds) *Designing for Digital Transformation. Co-Creating Services with Citizens and Industry. DESRIST 2020.* Cham: Springer.

Franklin, S. and Graesser, A. 1997. Is It an agent, or just a program?: A taxonomy for autonomous agents. *In:* Müller, J. P., Wooldridge, M. J. & Jennings, N. R. (eds) *Intelligent Agents III Agent Theories, Architectures, and Languages*. Budapest, Hungary.

Goldschmid, B. and Goldschmid, M. L. (1976) "Peer teaching in higher education: A review." *Higher Education* 5 (1), 9–33.

Gregor, S. and Hevner, A. R. (2013) "Positioning and Presenting Design Science Research for Maximum Impact." *MIS Quarterly* 37 (2), 337–356.

Gregor, S., Kruse, L. C. and Seidel, S. (2020) "Research Perspectives: The Anatomy of a Design Principle." *Journal of the Association for Information Systems* 21 (6).

Gunawardena, C. N. and Zittle, F. J. (1997) "Social presence as a predictor of satisfaction within a computer-mediated conferencing environment." *American Journal of Distance Education* 11 (3), 8–26.

Harborth, D. 2019. A Framework for Collaborative Augmented Reality Applications." *In:* Ludwig, T. & Pipek, V. (eds) *International Conference on Wirtschaftsinformatik*. Siegen, Germany.

Hunkeler, U., Truong, H. L. and Stanford-Clark, A. (2008) "MQTT-S — A publish/subscribe protocol for Wireless Sensor Networks." *In: International Conference on Communication Systems Software and Middleware and Workshops (COMSWARE '08)*.

Kammler, F., Brinker J., Vogel J., Hmaid T. and Thomas O. 2019. How Do We Support Technical Tasks in the Age of Augmented Reality? *In:* Krcmar, H. & Fedorowicz, J. (eds) *International*

Conference on Information Systems. Munich, Germany.

Kohn, V. and Harborth, D. 2018. Augmented Reality - A Game Changing Technology for Manufacturing Processes? *In:* Bednar, P. M., Frank, U. & Kautz, K. (eds) *European Conference on Information Systems*. Portsmouth, UK.

Kreijns, K., Kirschner, P. A. and Jochems, W. (2003) "Identifying the pitfalls for social interaction in computer-supported collaborative learning environments: A review of the research." *Computers in Human Behavior* 19 (3), 335–353.

Magnisalis, I., Demetriadis, S. and Karakostas, A. (2011) "Adaptive and Intelligent Systems for Collaborative Learning Support: A Review of the Field." *IEEE Transactions on Learning Technologies* 4 (1), 5–20.

Marbach, D., Enzler, U. and Dolder, P. (2020) *Appccelerate*. URL: https://github.com/appccelerate (visited on 7 April 2021).

Meth, H., Mueller, B. and Maedche, A. (2015) "Designing a Requirement Mining System." *Journal of the Association for Information Systems* 16 (9).

Miller, M. R. *et al.* (2019) "Social interaction in augmented reality." *PLoS ONE* 14 (5).

Mullarkey, M. T. and Hevner, A. R. (2019) "An elaborated action design research process model." *European Journal of Information Systems* 28 (1) 6–20.

Murata, T. (1989) "Petri Nets: Properties, Analysis and Applications." *Proceedings of the IEEE* 77 (4), 541–580.

Peffers, K., Tuunanen T., Rothenberger M. and Chatterjee S. (2007) "A Design Science Research Methodology for Information Systems Research." *Journal of Management Information Systems* 24 (3), 45–77.

Phon, D. N. E., Ali, M. B. and Halim, N. D. A. 2014. Collaborative Augmented Reality in Education: A Review. *In: International Conference on Teaching and Learning in Computing and Engineering*. Kuching, Malaysia.

Resta, P. and Laferrière, T. (2007) "Technology in Support of Collaborative Learning." *Educational Psychology Review* 19 (1), 65–83.

Riege, C., Saat, J. and Bucher, T. 2009. Systematisierung von Evaluationsmethoden in der gestaltungsorientierten Wirtschaftsinformatik. *In:* Becker, J., Krcmar, H. & Niehaves, B. (eds) *Wissenschaftstheorie und gestaltungsorientierte Wirtschaftsinformatik*. Heidelberg: Physica-Verlag.

Sein, M.K., Henfridsson O., Purao S., Rossi M. and Lindgren R. (2011) "Action Design Research." *MIS Quarterly* 35 (1), 37–56.

Short, J., Williams, E. and Christie, B. (1976) *The Social Psychology of Telecommunications*. London, UK: John Wiley & Sons, Ltd.

Smith, L. and Macgregor, J. T. (1992) "What is Collaborative Learning?" *Assessment* 117 (5), 10–30.

Webster, J. and Watson, R. T. (2002) "Analyzing the Past to Prepare for the Future: Writing a Literature Review." *MIS Quarterly* 26 (2), xiii–xxiii.

Westera, W. (2007) "Peer-Allocated Instant Response (PAIR): Computational Allocation of Peer Tutors in Learning Communities." *Journal of Artificial Societies and Social Simulation* 10 (2), 1–5.

Wiesböck, F. and Hess, T. (2020) "Digital innovations.", *Electronic Markets* 30 (1), 75–86.