Atelier - An Online Platform for Programming Tutorials.

Ansgar Fehnker, Angelika Mader, Arthur Rump, Margot Rutgers, Lotte Steenmeijer, Chris Witteveen
University of Twente
7500 AE Enschede, The Netherlands

CCS CONCEPTS

• **Social and professional topics** → **Computational science and engineering education**; • **Software and its engineering** → **Collaboration in software development**; • **Applied computing** → **Interactive learning environments**; **Collaborative learning**.

ACM Reference Format:
Ansgar Fehnker, Angelika Mader, Arthur Rump, Margot Rutgers, Lotte Steenmeijer, Chris Witteveen. 2020. Atelier - An Online Platform for Programming Tutorials.. In *the 9th Computer Science Education Research Conference (CSERC '20), October 19–20, 2020, Leiden, Netherlands.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3442481.3442511

The aim of the *Atelier* project[1] is to develop an online platform that creates an atelier-like setting that emphasises collaboration and sharing of ideas. It is built for the *Community of Practice* [1] of students, student assistants, and lecturers involved in teaching programming in *Processing* in the first year of the bachelor programme Creative Technology (CreaTe) at the University of Twente. CreaTe is a design programme, with an engineering background in Computer Science and Electrical Engineering. It extols its own design philosophy, which emphasises autonomous design, creative thinking, multidisciplinary teams, tinkering and reflection. The motivation behind *Atelier* to help the creation of a Community of Practice where face-to-face tutoring is central. It is consciously not intended to replace face-to-face tutoring.

A distinguishing characteristic of education in CreaTe is the central role of tinkering as a means to master the material [3]. By tinkering, we understand a self-directed, playful exploration of material. This educational innovation emphasises ownership and motivation. The focal point of learning programming is the tutorials, where students work on their projects, supported by a team of student assistants and lecturers.

From the beginning, students define their own design projects, while they are required to use concepts that were covered in the course. The student fully owns the problem; there is no example solution that students can work towards or that tutors can refer to. Instead, the projects have to demonstrate understanding and correct application of given programming concepts (e.g., classes or arrays), proper software design (e.g., encapsulation and event handling) as well as sufficient complexity and interactivity. This requires tutors to discuss quality aspects and requirements in the context of the student-defined project.

The project develops an online platform called *Atelier* to support the educational innovation in the programming courses of CreaTe. To do this properly, tutors need to be able to comment on the program as a whole, as well as on individual lines of code. Since we are working jointly towards the assessment, it needs to support threaded discussions, i.e. students should be able to ask for clarifications, tutors should be able to discuss internally, and students should be able to involve each other and tutors. The tool is intended to be used in a classroom setting. Traditionally, the TA will give the student feedback on his code verbally. The change is that the student can easily share the program (see Figure 1) and that the TA can also document the feedback in Atelier. Students and TAs can follow up on any comment, and involve other students or TAs (see Figure 2). Therefore, Atelier does not only serve as a feedback tool for students, but creates transparency between TAs and students on coding practices.

Figure 1: One way of sharing programs is achieved by QR codes, such that teaching assistants can access the programs easily on a mobile device.

[1]The Atelier project is supported by SURF as part of its 2018 call on Open and Online Education.

CSERC '20, October 19–20, 2020, Leiden, Netherlands

ACM ISBN 978-1-4503-8872-6/20/10…$15.00
https://doi.org/10.1145/3442481.3442511

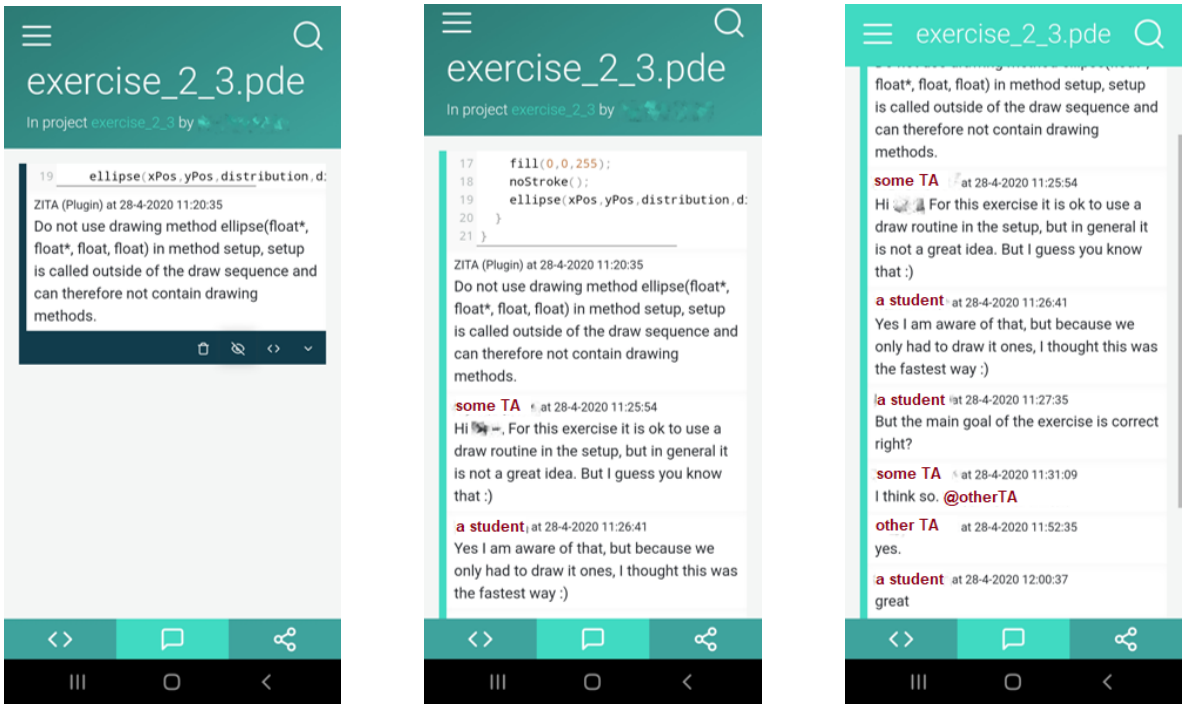

Figure 2: An initial automated comment was made visible by a teaching assistant and followed up by an explanation. The student replied, and another teaching assistant was involved in the discussion. Names have been replaced by generic labels "a `student`", "`some TA`", and "`other TA`".

To support the tutors *Atelier* will include the static analysis tool *Zita* as can be seen in the leftmost panel of Figure 2. It is a custom build tool to detect design smells in novice Processing code [2], which was presented at CSEDU 2018. Design smells are surface indicators for poor design and misunderstood concepts. Its use helps tutors to find problems systematically and consistently, which is a labour intensive and error-prone task for humans.

The research component of the *Atelier* project investigates the relationship between the following 3 core concepts:

- **CC1** Tinkering in programming education. This is defined by a self-directed mastery in self-defined projects.
- **CC2** Community of Practice. This is defined as a learning community related to a domain of practice, and mutual engagement.
- **CC3** Student assistant feedback. This is defined as the specific feedback students assistants provide during face-to-face tutorials on programming assignments.

The overarching research question for the *Atelier* project is how an online learning platform can support quality feedback to foster a Community of Practice in tinkering-based programming. This leads to the following specific research questions:

- **RQ1** Does the automated but moderated commenting improve the quality and consistency of feedback?
- **RQ2** Does the platform increase mutual engagement of students and student assistants on topics of software quality?
- **RQ3** Does the platform integrate with and improve practices of mutual engagement in a tutorial setting?
- **RQ4** Does mutual engagement on the online platform increase consistency in the program-specific practice of programming?
- **RQ5** Does the ability to document feedback online make the feedback more effective?

While the prototype of Atelier was developed as a mobile-first application, for use in a face-to-face tutorial, the focus changed to a desktop version and use in online tutorials, in response to Covid-19. The main challenges posed by the pandemic are its use in combination with other online tools, like queuing systems and video conferencing tools. The features include the ability to share programs, comment inline and on projects, tagging people, hashtags categorise feedback, an advisory tool that predicts to what extent a submission satisfies core learning outcomes, and a tool to evaluate the effectiveness of feedback. A preliminary finding is, for example, that automated feedback that is followed up by an assistant is more effective than sharing automated comments alone.

REFERENCES

[1] Marc Coenders. 2012. *De Canon van het Leren.* Vakmedianet Management B.V., Chapter Community of Practice - Etienne Wenger, 126–136.
[2] Remco de Man and Ansgar Fehnker. 2018. The Smell of Processing. In *Proceedings of the 10th International Conference on Computer Supported Education - Volume 2: CSEDU.* INSTICC, SciTePress, 420–431. https://doi.org/10.5220/0006701704200431
[3] Angelika Mader, Ansgar Fehnker, and Edwin Dertien. 2020. Tinkering in Informatics as Teaching Method. In *Proceedings of the 12th International Conference on Computer Supported Education.* INSTICC, SciTePress.