

Programming with a purpose

Lennart Rolandsson
Department of Education
Uppsala University
Uppsala, Sweden
lennart.rolandsson@edu.uu.se

This research is a full paper problematizing pair programming. The paper highlights how different purposes emerge in a programming activity as three students relate differently to the aesthetic words “ugly” and “efficient”. A problematic situation arises when the students pursue separate purposes during the activity, and it proves hard to overcome. This is something the paper discusses by focusing on various disturbances (emotional, intellectual and physical) to highlight the importance of designing collaborative educational activities with a clear-cut purpose, including or excluding the aesthetic value of the code.

Keywords: consummatory experience, habits, emotions, meaning making, pair-programming, practical epistemological analysis, situated artistic relations, artistic

I. INTRODUCTION

Pair programming is a method used by teachers for efficient learning. However, there are studies [1] eliciting the fact that students do not experience the need for collaboration, as they actually do not collaborate. In one study [2], the author suggests that people working on different levels of abstraction will find it challenging to communicate. In another study [3], they highlight how the differences in students’ skills influenced their compatibility and the outcome, and how working in pairs helped them move forward more efficiently when everyone took on the same responsibility. A reasonable expectation when one is committed.

However, all these studies expect that the individual constructs their knowledge of programming in a vacuum, not taking into account the impact of the specific situation and how the environment influences the students’ actions. It is a process this paper highlights within the transactional reality as students’ actions are analyzed over time. It could be argued that such a study is all about the interaction between the syntax and the programming environment (PE), as Dewey would prefer transactions, highlighting the fact that the subject and object emerge (and change) as a consequence of previous actions.

The study explores what is actually learned during the meaning-making process as students have to come up with new habits to cope. The paper approaches the meaning-making process as a continuous process, in line with the important *principal of continuity* [4], exploring the usefulness of disturbances and students’ emotions in relation to how students maintain equilibrium as the environment (including the PE) changes. Besides this, the study also offers a theoretical lens (data, theoretical framework and analysis) for a discussion of how aesthetic words can become the rationale behind a breakdown in collaboration. In fact, as the three sequences below show, there is a relation between the purpose of the activity and the aesthetic dimension when learning [5].

II. THE ENVIRONMENT

The environment concerns the circumstances the students relate to during the programming activity, i.e., the task, the code, the PE (the computer, the programming language as well as the debugger and the compiler) and each other’s actions. In the following, the task, the students, the PE and the code are described.

A. The task and the students

In an upper secondary school, three students (Olaf, Jan and Henrik) have been working on a digital board game (Swedish: luffarschack) during a seven-week period. The code of the game controls the number of markers in a row and the end of the game. It is a simple, straightforward task in a programming course for beginners. So far, five sections (methods) have been created, each responsible for something specific in the game. The division into sections is something they have achieved with the assistance of the teacher in order to work closely according to one of the standards of object-oriented programming (encapsulation). In fact, at the time when the students were video-recorded (with two cameras, one on the screen and one in profile), the majority of the code was already written. The study follows the students as they are engaged in working out two of the five sections: *rita* and *vinst* (see Fig. 1 and Fig. 2).

B. The programming environment (PE)

In a PE, one works in different computer windows in parallel: one with the code and one displaying the error messages. Under the hood, “behind” these windows, two services exist: a debugger that searches for syntactical errors and a compiler that draws from premade code libraries. In action, the programmer is in charge of how to improve the code with these resources as well as recompiling the new code to see if it solves the issue. In that process of rewriting and recompiling the code a number of times, syntactical errors can be solved. Logical errors can also be found, but not as easily as syntactical errors, which is why logical errors demand more iterations of the code to be discovered. The next chapter will go into the theory of how students’ predispositions (habits) are challenged in the activity as they have to express their “thinking” in code while working with it in the PE.

C. The code

In the video recording, the students started off in the first code section, *rita* (Fig. 1). However, their solution did not work out well, as the winner’s name did not display correctly, and one of the students (Olaf) persistently pushed for another solution in the second code section, *vinst* (Fig. 2). Consequently, after some issues and dissatisfaction with their work, the students re-focused on the second code section,

which made Olaf happy.¹ The sequences displayed in the paper (see chapter V) mainly concern what happened in the second code section but also how students relate to the code in the first section.

```
// The 1st section
private static void rita (SimpleWindow w){

    Square[] sq = new Square[49];

    // A marker is drawn as long as the return value
    // from the 2nd section is not true
    while (!vinst()){
        kryss(w, sq);
    }
    JOptionPane.showMessageDialog(null, "Vinst!");
}
```

Fig. 1. The first code section. The red square contains the loop and the call to the second section.

```
// The 2nd section
private static boolean vinst(){

    for (int i = 0; i < 47; i++){ // Markers in horizontal
        if (gb[i] + gb[i+1] + gb[i+2] == 3)
            JOptionPane.showMessageDialog(null, "The Blue is a winner");
        return true;
    }
    if (gb[i] + gb[i+1] + gb[i+2] == -3)
        JOptionPane.showMessageDialog(null, "The Red is a winner!");
    return true;
}

for (int i = 0; i < 35; i++){ // Markers in vertical...
}

for (int i = 0; i < 33; i++){ // Markers in diagonal...
}

for (int i = 0; i < 37; i++){ // Markers in diagonal...
}
}
```

Fig. 2. The second code section as it became after the third sequence.

III. THE THEORETICAL FRAMEWORK

This chapter describes some of the foundations of how humans deepen and create new habits as disturbances evolve in the activity. Much of the theory is from [6] and [7]. In Dewey's terms [4], learning is the outcome of pursuing an inquiry and gaining transformative experiences that change our habits. The following chapter *Result and Analysis* will go into how students decide to work within short and long learning loops to overcome these disturbances.

A. Learning concepts and creating new habits

According to Dewey [4], we commonly live our lives as our habits ordain, without any further reflection. This is a useful theory for studying how we both apply new concepts and learn about them as we experience their benefits. As a learner of programming, one draws from these and other concepts, which, in time, become valuable in the coding process and one begins to appreciate them, which leads to gaining new habits. In this paper, the students touch upon the concepts of *encapsulation* and the *return value* of a call. In fact, a *global variable* could be used to solve their inquiry. be useful for them in this lesson. However, the teacher's intention was to present the global variable during the next

lesson, where its value would become obvious to the students. In the process of finding a purpose through the concepts embedded in the code, the students move forward with the task.

B. The problematic situation and different learning loops

As the three sequences describe, various disturbances occur, which lead the students into a problematic situation; a habit is disturbed and the activity halts as they need to decide how to pursue the inquiry. This could be accomplished within a short learning loop [6], as this disturbance could easily be solved with only minor friction and would confirm their established habits. However, the problematic situation could also transform into an inquiry [4] with a long learning loop if it continues to challenge the students.

The long learning loop is initiated when the students' habits are not sufficient and they decide to persevere in order to solve the problematic situation. A situation in which the students have to reflect on and rearrange what resources they have in their environment [4]. As changes in the code are tested, to an observer, the students' working process might seem to be moving in useless and unexpected ways, but in that rearrangement of code, they also gain useful experience that will lead them toward a solution as the issue becomes clear. This way the students get to know the different aspects of the problematic situation and what they are expected to achieve. In that process of adjusting the code with the help of rearranging what they have, the students will come to realize their goals and how to move forward [6].

In short, problematic situations are solved in two ways: 1) in short learning loops that strengthen habits and 2) in long learning loops where habits are re-constructed or created. The meaning-making that happens in the long learning loop expands the students' capability to:

- discern what is relevant in the environment. The student gains attentiveness and ability to focus on the relevant objects (code and tools) to use in the activity.
- coordinate actions and thinking in relation to the learning environment, fulfilling a specific goal and purpose.

C. The importance of disturbances and emotions

The meaning-making process becomes crucial as we act in relation to what we experience as disturbances in the environment, e.g., peers' actions and changes in the code. This study scrutinizes these disturbances in order to gain a better understanding of what made the students deepen and transform their habits.

According to Dewey [4], in order to learn and reflect during the problem-solving process, one is in desperate need of disturbances and friction from others and the environment. For example, if the others are experiencing a disturbance or it is not clear how the code should be written, one could feel their habits are not adequate in that situation. Nevertheless, what triggers one in that situation is about one's emotions telling one something is out of balance. For instance, one becomes frustrated as the activity does not move forward [7]. In such a situation, one has to abandon one's old routines and

¹ Olaf made a victory pose with his right arm.

look for new practices in order to reflect on what happened and why there is a disturbance in the first place. One has to deal with the problematic situation in some way.

As a human being, one therefore tries to regain balance and find a solution where one's emotions (frustration, happiness, etc.) will communicate that something has changed, and one can establish a fulfillment of experience. Dewey's [8] theory lets us know that we can experience immanent emotions within our bodies [8, p. 119] as something is fulfilled, e.g., we do not utter "yeah!" after reflection, but instead it is said spontaneously and immediately when the solution appears in front of us. According to Dewey, when such emotions are expressed in action (body or speech), we become aware that the student we are focusing on is having an aesthetic experience, which is something that is empirically proven in the classroom [9] and framed as a consummatory experience to highlight the fact that these experiences are a part of the activity. However, the consummatory experience does not always happen, as the environment itself needs to offer some sort of conflict, friction, etc., to make one aware of a need for the re-evaluation of one's habits.

Without friction or disturbance, we would never know anything about the students' aesthetic consummatory experience [8, p. 14–15]. However, when a student is identified as having such an experience, we will approach it through an analytical window to touch upon the meaning-making process and the (bodily) expressed emotions. These emotions could be analytically separated upon reflection [8, p. 37–38], as the students express them through aesthetic speech, e.g., "nice", "ugly" and "this is sick".

IV. THE ANALYTICAL METHOD

According to Wittgenstein [10] we could approach a conversation as a language-game; how words and actions acquire meaning by their users in relation to the social practices of which they are a part. In this study, the sequences are approached in a first-person perspective as suggested by Wittgenstein [10] with implications for how the conceptualization of words and actions are unpacked where the work surrounding programming and the expressions carried out through code offer significant information about what is actually shared through action. This way of analyzing and understanding meaning-making is further described below [11].

In summary, the sequences chosen for this study are analyzed based on what the students' utterances can tell us, as well as the actions into which Wittgenstein's language-game is woven. In a similar manner to Wittgenstein [10], the study approaches the students' activity with the assumption that students embrace various experiences of how utterances and actions can be used. These utterances and actions are most likely difficult to unpack by themselves and, therefore, almost impossible to escape. However, it is possible to study them if they are expressed out loud and are visible to others as a public language [10]. In that process, we as educational researchers are able to capture emotions in action as soon as disturbances emerge in the activity.

In the next chapter, the students express a number of immanent emotions when something distracting or

intellectually demanding/interesting is introduced in the activity, e.g., when the process slows down (frustration) or restarts (happiness). In that spirit, specific sequences were searched for where students, from time to time, invest a lot of themselves.

A. The analytical tools, PEA and SAR

Practical epistemological analysis (PEA) is an analytical tool described by different researchers using transactional learning theory [11]. It includes four concepts that help us understand how the meaning-making process evolves in the activity and changes as the activity continuously transforms: *stand fast*, *encounter*, *relation* and *gap*.

In short, *stand fast* implies that the participants' actions are not questioned or in doubt. This is important as *stand fast* signifies the instances that can evolve into something more when the students become involved with the environment. An *encounter* can be what happens between the students (their actions and utterances), but it can also be what happens between a student and the code or a code structure, etc. As an encounter develops, a *gap* can emerge. However, the gap can be bridged with the *relation* it shares to what is already standing fast. In that process, learning occurs as previous experiences are transformed. However, that is not always the case, i.e., everything that stands fast can be questioned, and sometimes it is impossible to communicate the issue in words and thus actions have to be invented or the physical dimension has to be rearranged [9]. In such (problematic) situations, a lingering gap is experienced in the activity until the disturbance is solved.

Next, an extension of PEA will be described. The situated artistic relations (SARs) that occur when students transform an activity into something artistic. It is an analytical method [12] to extract aspects of meaning-making. Aspects that are mainly concerned with what is expressed as it relates to aesthetic actions. The analysis consists of the following four steps:

- **Step 1:** If any aesthetic expression is found, it is traced backwards to see if such an expression is a sign of fulfillment of experience. The expressions thus found are presumably connected to what is being used (the code) or to the expected end result of the task.
- **Step 2:** The trajectory found in step 1 is scrutinized to see if it stands for an aesthetic sensation, a repetitive reward or an artistic creation.
- **Step 3:** The disturbance is controlled in relation to the following criteria: a) the overall purpose of the lesson and b) the use of programming knowledge/skills and the programming environment.
- **Step 4:** If the students work in an artistic manner, the activity will lead them toward a fulfillment of experience, which is something the analysis will unravel as a positive and a negative normative statement, where the activity stops or continues in the inquiry process. If a positive normative statement appears in the transcript, the students will have what Dewey calls a consummatory experience.

V. THE RESULTS AND THE ANALYSIS

In this chapter, the connection between aesthetic actions and aesthetic experiences will be described. It is a connection with which other researchers have worked, as well [9]. After each sequence, a PEA follows, describing the relations the students create with each other and the environment to overcome the problematic situations. The word “ugly” will be scrutinized in relation to the purpose of the task, and SARs will be taken into account in order to understand how the artistic can become a challenge in collaborative work when all the students have to find the purpose of the activity.

The analysis is built on three carefully selected sequences to describe how the inquiry evolves over time according to the principal of continuity (Dewey), unravelling a qualitatively interesting case. The three sequences are described as: Table 1 (resisting control), Table 2 (giving it a try) and finally Table 3 (the code is demanding to write).

TABLE I. RESISTING THE CONTROL

172	J	Take the control and write the code, Making us all see what it can become!
173	O	But, you are in control of the keyboard!
174	J	No, no, no
175	O	You are the one!
176	J	I am not sure what code you would like me to write.
:	:	:
179	J	You do the code, we will see how it works
180	O	But, I will write ugly code
181	H	<u>Do you write ugly code?</u> (Jan bends forward, resting his cheek in his palms)
182	O	Yes
183	H	<u>Come on, we are talking about a computer!</u>
184	H	<u>What do you mean?</u>
185	O	(Olaf gets control)

In the first sequence, it is obvious that the students need tangible code, including a direction of how to continue (172 and 176). They also depend on what the PE brings to the inquiry (176, 179, 181, and 183), making them agree on how the board game should work. We can also see Jan prefers to run the code to “see how it works” (179). Some knowledge stands fast, e.g., how to use the PE and the function of the code sections. However, someone has to take the lead and reify their ideas into a working algorithm (172 and 173). In short, the first sequence describes a disturbance (a collective gap): “Who will write the code?” In that situation, for the first time, the students encounter the aesthetic word “ugly”. In the following, we will see how they relate to that word in different ways and succeed in overcoming that disturbance. A gap between what is standing fast and what they encounter is at hand.

Jan prefers for Olaf to be in control (172, 174, 176 and 179) of the keyboard and mouse. On the other hand, Olaf wants Jan to stay in control (173, 175, and 180). Accordingly, this is a situation where Olaf’s reluctance toward leadership and Jan’s trust in Olaf’s ability imply a problematic situation, as Jan “[is] not sure what code you would like me to write” (176). The situation evolves into an even worse one, as Henrik becomes astonished (or at least overwhelmed by surprise) as he relates to Olaf’s relation to “ugly code” with a question “Do you write ugly code?” (181), followed by a “Come on, we are talking about a computer” (184). A

disturbance and an aesthetic expression (SAR step 1) as well as a relation to the code is established with “ugly”. However, the expression “ugly code” could be an aesthetic sensation more than it is an artistic creation (SAR step 2), wherefore more information about their process need to be collected. In fact, the expression “ugly code” does not follow from the original purpose with the task (SAR step 3), and the students’ actions (172 to 185) do not tell us anything about a consummatory experience (SAR step 4). That expression tells us Olaf avoids, or at least resists, something he dislikes. He does that with an immanent emotion (concern) in his utterance “But, I will write ugly code” (180). It is something that is not easily shared between them, but it is still a disturbance they have to take into consideration if they want to proceed together with the inquiry.

That immanent emotion (concern) is as follows: Olaf compares the present code version with something from the future. However, the future is unknown, especially as the students are in “the process of transformation, the end of which they cannot foresee while experience is still in the making” [13, p. 292]. Therefore, approaching this sequence in a first-person perspective, Olaf’s statement becomes intriguing. However, the code on the screen can facilitate the next step, in a similar manner as ordinary written text on paper helps us sharpen the rationale of our message and reflect on what matters in the long run. Therefore, the comparison between the versions of code, one on the screen (in the present) and another (ugly) code version in the future (180), is not that bad after all. It is a comparison that tells us Olaf is working artistically, within a habit where he moves in time between the present and the future, which is something to be further explored in the following sequences.

Finally, they have solved the problematic situation – who will write the code – and next, “ugly code” is debated (181 to 184). A step with implications on the inquiry process, where Olaf has to establish some common grounds.

TABLE II. GIVE IT A TRY

192	O	Do you follow my thinking?
193	J	Yes, but we need to modify the code (Jan regains the control)
194	H	On <u>each instruction, then?</u> (5.0)
195	O	The question is, will it be more or [less efficient]
196	J	This one has to be at the end (of the code)
197	O	Or, will it be just too much code]
198	J	This one has to be at the end (of the code) (1.0)
199	O	Yes, I just made a draft, so you would follow me.
200	H	Yes indeed, it is not impossible at all
201		We try it out! (1.0)
202	O	<u>But, is it efficient to do the code that way?</u>
203		<u>It feels so ugly</u> , to work that way.
204		The code becomes <u>huge!</u> (39.0)
205	J	(Jan scrolls up and down in silence. He moves away in frustration)
206	O	(Olaf laughs)
207	O	The things are complex. For the moment we are (stuck)
208		We give it a try? (Olaf gets control)
209		<u>I am sick of this!</u> Check this out! We do it this way.
210		We give it a try! We preserve and then we write our message dialog box as it is, or we try it out to see how inefficient slash efficient it is (The draft is transformed into a comment, by Olaf)
211	H	Yes okey, okey, okey.
212	O	It is worth a try! (2.0)

213 H It is! We can actually give it a try, on one (instruction).
Honestly!
 214 O What?
 215 H Nothing.
 216 O This [the code] could also be preserved as it is
 217 That's what so nice with code
 218 You don't have to erase to restart every time
 219 In fact, we can put the code in (quarantine?)

The sequence above is the sequence where they encounter the draft (see Fig 3.) for the first time (192), and Olaf expresses “Do you follow my thinking?” in a manner that resembles an invitation or a way to check if they all agree or whether they will criticize the draft. Nevertheless, what interpretations to be found, Jan immediately relates to the draft with a suggestion for improvements (193, 196 and 198), but Olaf’s reply is somewhat strange when he emphasizes that “I just made a draft, so you would follow me” (199). A statement that makes us aware of something specific going on. In fact, Olaf initiates a long learning loop with the draft (192 to 201), but in that process, Olaf brings up the future (again), as he questions the efficiency if they “... do the code that way” (202). His expression is followed by an immanent emotion “It feels so ugly” (203) as “The code becomes huge” (204). This is obviously an emotional disturbance that none of them can relate to except for Olaf.

```
// The 2nd section
private static boolean vinst(){
  int color = 0;
  String winner = "";
  if (color ==3)
    winner = "The blue is a winner";
  else if (color == -3)
    winner = "The red is a winner";

  for (int i = 0; i < 47; i++){ // Ma
```

Fig. 3. The red square describes the draft

After 39 seconds of silence, Olaf laughs (206) and takes the floor as he underlines “I’m sick of this!” and “check this out” (209). On top, he intends to explore the “efficiency” of the code (210). These moments are interesting in that they tell us Olaf has decided to overcome the gap. He also repeatedly says that they have to give “it” a try (207, 209 and 211), which is something Henrik relates to (200, 201, 211 and 213). In comparison with the first sequence, Olaf has changed his attitude, as he “...give it a try” (210) and takes the chance of writing “ugly code”. Indeed, they all accept and relate to the draft and therefore stand fast. However, the focus on the draft vanish as Olaf transforms the draft into a comment (210). A comment that remains in the code throughout the lesson without any further function.

Nevertheless, another conversation of importance happens in parallel with the first one, expressing other disturbances, as well. In fact, Henrik relates early on to the draft with a question “On each instruction, then?” (194). A gap that is never addressed until later (213) where Henrik rephrases his question, suggesting a one-liner. This is a moment where Henrik relates to Olaf’s question about “efficiency” (202). Unfortunately, Henrik does not push for his idea (the one-liner), when Olaf replies with a “What?” (214). Instead, Olaf goes on (210, 216–219) to explain how

they can always rewind the code (in the future) if the code becomes “ugly”. The situation is complex (208), and upon closer inspection, it is obvious there are some emotions they have to deal with. However, before discussing the emotions (in the next chapter), the third sequence need to be presented, where Olaf fulfills his expectations (his purpose).

TABLE III. THE CODE IS DEMANDING TO WRITE

217 J You have to use (1.0) parenthesis
 218 O Oh no, that is demanding to do! (Olaf chuckles) (11.0)
 219 J Indeed, it is demanding to do!
 220 O (Olaf adds parenthesis) (17.0)
 221 O Oh, this became demanding (Olaf chuckles)
 222 H Yes (10.0)
 223 O Yes, the code becomes beautiful anyhow
 (Henrik and Jan play with their mobiles)
 224 O Wait a moment (he adds some more code) (66.0)
 ⋮
 233 H What have we done? (Henrik chuckles)
 234 We have destroyed my code!
 235 O No way! It is cool, it is chill! It is chill!
 236 The things that never been, will never be destroyed.
 237 H That was dam ... interesting. (6.0)
 238 O Okey.
 239 Hell it, the code became ugly anyhow! (Olaf chuckles)
 240 J (Jan regains control and Olaf moves back)

In comparison with the first and second sequences (Tables I and II), this is the sequence where there is an explosion of immanent emotions. Olaf mentions the code is “demanding to do” (218, 219 and 221. Swedish: “jobbigt att göra”), followed by chuckling (218, 221 and 239). Finally, in turn 223, Olaf makes his first positive normative statement with “beautiful anyhow” (Swedish: “ snyggt ändå”). In fact, this could be the consummatory experience (SAR step 4), where the code finally becomes “beautiful”. But in that exchange going from a negative to a positive statement, Henrik and Jan have lost interest in the activity (223). In summary, Olaf is on his own, finishing whatever there is still to do, and at the end of the sequence, Olaf goes back to a negative normative statement “the code became ugly anyhow” (239). Nobody confirms his experience! The breakdown in collaboration is a fact, and it is something that needs to be discussed in detail.

The third sequence ends in a remarkable way when Henrik utters in despair “We have destroyed my code” (234), which is something Olaf distinctly refutes with a “No way ...” (235) and “The things that never been, will never be destroyed” (236), almost like a philosopher as an earlier version of the code is preserved on the hard drive (210).

VI. DISCUSSION

In this chapter, the second and third sequences are at focus (Tables II and III). They are the sequences where Olaf is intrigued by a specific purpose, one that is not intended by the teacher, to explore what is “ugly” and “efficient” in code.

Between the first and the second sequences, a draft was created. However, the code draft created a gap, and the students were obviously not able to find the same purpose, as the draft is transformed into a comment (210). Therefore, it is of paramount interest to study how the students relate to the draft and encounter each other’s arguments and make meaning in that situation. In the beginning of the second sequence, it seems like Olaf relates to his peers and their ability to follow him (192), but it is more reasonable to

assume that Olaf has already focused on exploring and explaining how the code could become beautiful and efficient. This is something that becomes obvious in the third sequence, where Henrik returns to an earlier gap (see 194), and this time Olaf replies in a different manner (235 and 236) than previously (210, 218 and 219). The code does not need to be destroyed, and they can always rewind to an earlier version. In conclusion, Olaf is working artistically to explore and express “beauty” and “efficiency”. However, he has to find a way forward in order to fulfill his expectations of the activity (purpose) and communicate his values regarding the code, to Henrik and Jan.

To summarize, in the first sequence the students are excited by the task (the level of intensity is high), but someone has to take the lead to help them figure out “Where to put the code?”, and at first Olaf seems reluctant toward that mission. Next, in the second sequence, Olaf’s attitude switches from “I write ugly code” (203) to “We do it this way” (209). In this sequence the draft becomes something to focus on, and if Henrik had explained his perspective on efficiency instead of avoiding (215) the question “What?” (214), the outcome could have been different. Instead, they agreed to “give it a try” (212 and 213) in a long learning loop to explore what “it” could become (200, 201, 208 – 213).

It is obvious that Henrik relates to the draft (200 and 201), and Jan also suggests specific improvements to the code (196 and 198). Nevertheless, neither Jan nor Henrik relates explicitly to the word “efficiency” (195, 202 and 210). Henrik makes an attempt with a one-liner, but with minimal impact. In short, the situation is problematic where each individual experiences a gap: 1) Olaf has made a draft but is still not satisfied (203 and 204), 2) Jan compares the draft with what they did earlier in the first code section (205) and 3) Henrik relates to the “efficiency” with a one-liner (211 and 213).

In that situation, some emotional expressions are also found, e.g., Olaf gazes at Jan and relates to the situation with laughter (205). The laughter is unexpected but it is explained when Jan moves away from the keyboard and mouse and exhales loudly in despair. In conclusion, the sequences offer a number of turns where the students are heavily involved as emotions are expressed many times, e.g. “It feels so ugly” (203), “I am sick of this” (209), “Honestly” (213), “You don’t have to erase” (218) and “What have we done” (233). The problematic situation consists of emotional and intellectual disturbances that they need to solve before they can collaborate successfully. As it stands, Olaf is on a personal journey, performing an artistic exploration of the code (from 210 to 239).

In short, Olaf is in control of what will become of the code, and he is working artistically (SAR step 4), as he has focused on the goal of the activity to explain and explore the “ugliness” and “the efficiency” of the code. He has also preserved the code (210) in case the worst should happen. In fact, the draft is created after the ugliness-in-code debate started (180 to 185); a rapid discussion highlighting students’ differences in how they relate to aesthetics in code. According to Henrik, code could never become “ugly” – a computer never writes ugly code – and this is in stark contrast with how Olaf explores the aesthetic word “ugly”. It is also interesting to see how Henrik suggests (213) a one-liner when

talking about “efficiency”. A word that was never shared between them, but is still of relevance as they reach closure when Henrik experiences how his code is destroyed (234).

The activity reaches a situation where they hold different purposes regarding the activity and embrace different interpretations of the words “ugly” and “efficient”. It is obvious the students do not work with the same purpose. The activity has evolved into a complex situation leaving various emotions unsolved, which is something that can be considered problematic for their meaning-making process.

VII. CONCLUSION

The three sequences above describe a tension that could be the consequence of the fact that the students are working on different levels of abstraction [2], but as the analysis unravels, there is something else underpinning the activity. This paper elicits the fact that aesthetic words are interpreted in different ways, which is something that aligns with Wittgenstein’s allegory – the beetle in the box – where we can never know for certain what we have in common with others until we express our differences in action, elaborating on our knowledge *in situ*. In this case, the ugliness and efficiency of code is something the teacher does not intend in the activity. Nevertheless, one of the students preferred working artistically, which was something the other students never expressed in action during the whole lesson. This highlights a difference in habits that leads them into a problematic situation and finally a breakdown, as the aesthetic words were never exploited.

As this paper describes, code is a material that one can use to express oneself artistically, which is something Dewey [8] emphasized: “where material is employed as media is there expression and art”. However, in this specific case, it is challenging to expect students to work in collaboration, especially as one of the students is working artistically and the others are not. Therefore, the pedagogical takeaway concerns how aesthetic words could evolve into valuable resources for teaching and learning about programming. Otherwise, the students risk to learn mainly by navigating between differences in interests and values, with the additional risk of finding too many purposes in the activity.

ACKNOWLEDGMENT

The work is supported by grants from the Swedish Research Council, and Marcus and Amalia Wallenberg Foundation.

REFERENCES

- [1] S. Goel, and V. Kathuria, “A Novel Approach for Collaborative Pair Programming”, *Journal of Information Technology Education*, 9, 183, 2010.
- [2] S. Bryant, “Double trouble: Mixing qualitative and quantitative methods in the study of eXtreme programmers,” in *Proc. VL/HCC*, Rome, Italy, 55-61, 2004.
- [3] N. Katira, L. Williams, E. Wiebe, C. Miller, S. Balik, and E. Gehringer, “On understanding compatibility of student pair programmers”, *Conference paper SIGCSE’04*, March 3–7, 2004, Norfolk, Virginia, USA.
- [4] J. Dewey, *Logic - The theory of inquiry*. John Dewey: The later works 1925-1953, A. Boydston. Carbondale, Southern Illinois University Press. 1938/1986.
- [5] P.-O. Wickman, V. Prain, and R. Tytler, *Aesthetics, affect, and making meaning in science: An introduction*. (manuscript), 2020

- [6] J. Andersson, J. Garrison, and L. Östman, *Empirical Philosophical Investigations in Education and Embodied Experience*, Palgrave, 2019.
- [7] K. van Poeck, L. Östman, and J. Öhman, *Sustainable Development Teaching: Ethical and political challenges*, Routledge, 2019.
- [8] J. Dewey, *Art as Experience*, New York: Perigee books, 1934/1980/2005.
- [9] P.-O. Wickman, *Aesthetic Experience in Science Education: Learning and Meaning-Making as Situated Talk and Action*, Mahwah, NJ: Lawrence Erlbaum Associates, 2006.
- [10] L. Wittgenstein, *Philosophical Investigations*, New York: The Macmillan Company, 1953/1969.
- [11] P.-O. Wickman, and L. Östman, "Learning a discourse change: A sociocultural mechanism", *Science Education*, 86, 601-623, 2002.
- [12] M. Jonsson and L. Östman. (manuscript), 2020.
- [13] A. Jornet, W.-M. Roth and I. Krangle (2016) A Transactional Approach to Transfer Episodes, *Journal of the Learning Sciences*, 25:2, 285-330.