Studying the Phenomenon of Developing Interest in Learning How to Code

What Happens to the Interest of Brazilian Undergraduates Over an Introductory Experience

Pasqueline D. Scaico, Ruy José G. B. de Queiroz
Center of Informatics
Federal University of Pernambuco
Recife, Brazil
{pds,ruy}@cin.ufpe.br

José Jorge Lima Dias Jr., Alexandre Scaico
Department of Exact Sciences
Federal University of Paraiba
Rio Tinto, Brazil
{jorge,alexandre}@dcx.ufpb.br

***Abstract*— In some CS0 settings, innovation and use of technology are not always an option due to costs, poor internet infrastructure and cultural aspects of teaching. It also happens that students are not familiar with CS since they are not educated about it at schools. These circumstances pose a more challenging situation in terms of encouraging beginners to become interested in this subject during an introductory course. In this study, we were motivated to study interest for learning coding in naturalistic contexts and from a dynamic point of view. To investigate it in this fashion, the Four-Phase Model of Interest Development was used. Based on an in-depth and a qualitative design, two groups of Brazilian beginners were observed. In this paper, we present how their interests have unfolded and why changes happened. Contextual factors, such as level of structure around the teaching process and tasks, and the educator's attitudes, emerged as positive influences. By understanding why beginners felt interested in engaging with coding, the role of some human factors in these learning experiences came up. Some students were not skilled at self-regulating. Also, when their beliefs of being able to learn coding dropped, willingness to engage was negatively affected. This study brings a non-fragmented view of the process of interest development in learning coding and its complexity. The findings highlight a need for educators to become more informed about how this process takes place and what individual characteristics are involved with it. To some students a motivational learning environment must be thought of not only in terms of a technical point of view, but also in terms of adjusting educators' expectations and discussing how to better manage human factors that play an important role in introductory experiences.**

Keywords*— *Interest development; trajectories of interest; programming education; CS0 course; qualitative research.

I. INTRODUCTION

Acquiring coding skills can be a frustrating and time-consuming process that turns learning into something discouraging. Having beginners interested is essential to keep them engaged, persisting and making an effort to succeed. In this sense, it is important to promote an environment in what interests can be supported to grow. Over the last decade, substantial endeavors have been made in developing motivational resources to encourage beginners to become more interested in learning coding, such as games, robotics or intelligent assistants [1, 7, 12]. However, relying on one-off solutions does not seem to be effective for achieving this complex goal. Although some perceptions surround what beginners in programming consider interesting, as [13] pointed out, it remains unclear what really impacts students or promotes longer term changes in their interests. As a research topic, interest in coding was narrowly addressed in CS education, particularly in higher education settings [21]. While many efforts are employed on development and technical aspects, little research has explored the possible effects of some motivational resources on students. In this way, anecdotal evidence surrounds some of the claims [14]. Also, interest in learning to code has been overwhelmingly studied under a fragmented and decontextualized point of view which might be a result from a tradition built upon quantitative methods of research.

We perceive interest development as a phenomenon which cannot be understood through a diminished view of its complexity. Our research focused on reaching a deeper understanding of how beginners experience it. We were driven to bring more context to the process of inquiry. Understanding context is essential considering the existence of distinct realities when it comes to experiences with introductory programming, especially those that rely on underprivileged circumstances. To a significant number of CS0 courses, innovation in terms of pedagogy and educational technology does not reach the classroom. Sometimes, learning relies on content-driven approaches due to a shortage of infrastructure and human resources and also, cultural aspects of teaching. Additionally, in countries without a minimum curriculum in CS at schools, which is the case in Brazil, students' aloofness and their misinterpretations about this area challenge even more the process of getting them interested in coding. In [15], Pears and colleagues pointed that three decades of active research on the teaching of introductory programming has had limited effects on the classroom. This is still a worldwide real scenario.

In this sense, the research related to interest and programming education still have gaps. To help contribute to this literature, we have been observing the phenomenon of developing interest in learning how to code in a situated and naturalistic way. From a longitudinal design, we tracked the

978-1-5090-5920-1/17/$31.00 ©2017 IEEE

experience of Brazilian CS students attending CS0 courses. We performed in-depth qualitative studies to investigate how the interest of sixteen beginners has unfolded. The Four-Phase Model of Interest Development (4PM) was used to guide the process of observing expressions of interest and recreate them as trajectories. The way we operated the 4PM provides a new perspective to scrutinize interest in programming settings. This study reveals what happened to the interest of beginners in coding during an introductory experience. We describe their trajectories of interest and the main influential forces that shaped the way in how these participants felt interested (or not) in engaging with this subject. Understanding interest from this perspective might better support how educators can cultivate interests for programming and work with them.

This paper is structured as follows: in section 2, the theoretical framework that guided the study is presented. Section 3 shows some specifics from the research method adopted. The context where the study was carried out, including a description of participants, is provided in Section 4. The results obtained from the analysis process are reported and discussed in sections 5 and 6, respectively. Final remarks are noted in the last section.

II. THE FOUR-PHASE MODEL OF INTEREST DEVELOPMENT

Motivational factors play an important role in the learning experience and its outcomes. One of them is interest, an element that can "energize"[1] the way people think. According to [9], interest is a central component to cognitive processes due to its power to influence how people choose to engage and persistently stay involved with certain things instead of others. The level of interest will drive learners to set value for objects (tasks, physical objects, ideas or content), define goals and help them to realize the effort required to succeed [3, 16, 22].

Interest is a psychological state of being willing to (re)engage with certain objects built under emotional and cognitive components [17]. While emotions frame the engagement with the object, cognition brings meaning, value and knowledge about it [19]. People become interested in certain objects and the involvement with the object itself is more important than any other reward which might result from it. This relation *person-object* was central to building this research field. An interest manifests itself as an effective reaction to something that is interesting and capable of catching someone's attention, which puts the person in a "state of interest" [11]. For some scholars, it is not a trait of personality as from other variables like motivation [18]. Interest rises from interacting with the environment. Based on this view, interests can be developed. An interest evolves sequentially in stages, as Hidi and Renninger [10] explain. Under an educational perspective, they present a theoretical framework, the Four-Phase Model of Interest Development (4PM), to explain its process of development.

As an interest develops, its nature changes and the learner experiences different expectations and needs regarding the environment (Fig. 1). As much it is settled, steadier is the disposition of the learner who engages with the object. The source of a new interest is primarily situational since the stimulus that triggers it comes from the outside. However, it needs to be noted that the environment is crucial in earlier phases of the process. Triggering a situational interest represents the first stage of the development, when the connection with the object is established. At this point, interest is basically an emotion. As the cognitive side is not developed yet, it is possible that the learner does not have knowledge or meaning about the object. However, as long as proper conditions exist, learners will keep their attention on the object and his/her interest for it might evolve into the second phase. If this happens, it is said that the situational interest is maintained. At this stage, the learner is capable of investment and will make some effort to be engaged. But, since there will likely be little knowledge, involvement can be lost whether it requires autonomous behavior or tasks that are meaningless or unattractive.

PHASE 1	PHASE 2	PHASE 3	PHASE 4
LEARNERS:			
Attend to content, if only fleetingly Need support to engage (from others and through instructional design May experience either positive or negative feelings May or not be reflectively aware of the experience	Reengage content that previously triggered attention Are supported by others to find connections between their skills, knowledge and prior experience Have positive feelings Are developing knowledge of the content Are developing a sense of the content's value	Are likely to independently re-engage content Have curiosity questions that lead them to seek answers Have positive feelings Have stored knowledge and value Are focused on their own questions May have little value for the canon of the discipline and most feedback	Attend to content, if only fleetingly Have curiosity questions and positive feelings Self-regulate easily to reframe questions and seek answers Can persevere through frustration and challenge in order to meet goals Recognize others' contributions to the discipline Actively seek feedback

Fig. 1. Learner's features in each stage of the 4PM. Adapted from [18]

As the interest grows, learners increase their knowledge about the object, which shifts them to more independence from the environment. When it reaches the third phase, learners develop curiosity about the object, which helps them pursue "what if" questions and engage of their own accord. This is a hint of how much interest is developed and means that it is becoming more internalized. In phase 4, learners are capable of demonstrating self-regulated behavior and a self-directed attitude, persistence in the face of challenging situations, and adjustment of their own learning agenda. Individual interests take time to be developed but represent a permanent state of motivation that is more valued as a learning outcome, according to [11]. In this study, due to its explanatory potential, the characteristics described by the 4PM was used to guide the process of realizing participants' expressions of interest.

III. THE RESEARCH METHOD

The goal of our research was to understand what happened with interest of beginners learning to code. We were thrilled to investigate interest development under a dynamic perspective so we could observe the initial stages of it, possible changes in its nature and influential factors. In order to accomplish this goal, we set the following research question: "How and why do

[1] "Energize" was a term used by Jean Piaget in 1980s.

interests in learning programming change across an introductory course?" We were also interested in collecting evidence to better explain how beginners in programming make sense about their learning experiences and how all of it shapes their states of interest. The research question called for a qualitative approach of inquiry. Because of our philosophical stands, as interpretative researchers, we try to make sense of that phenomenon by looking at meanings that participants build into it and by taking their context into account. We based the research on a case study because in this way a deeper understanding could be realized. Studying a case involves observing a phenomenon in a real-life way, while setting aside control or manipulation of variables [8]. To reach richness in understanding our central phenomenon, we decided to study a case consisting of two units of analysis. We considered as a unit of analysis a group of beginners attending a CS0 course. Observing these groups, we could observe interest development in distinct situations. Next, more details are introduced.

A. Units of analysis

The case was studied in a public university, settled in a small town in Brazil, that offers two undergraduate programs in computing: a teaching degree in CS and a bachelor degree in management information systems. Dropout and failure rates have been an issue to the CS0 courses of these programs. As we mentioned before, two group of beginners, one from each CS0 course, were taken as the units of analysis.

The CS0 courses had the same syllabus (Python was used as a programming language and the course load was 60 hours per semester, organized in two meetings on a weekly-basis) and class size (over 50 students). Educational technology was not included as part of any instructional agendas. The instructional contexts demonstrated somewhat strong contrasts in regard to incentives and restraints such as course materials, systems of feedback, instructional design, pedagogical strategies and the instructor' experience with beginners in coding. Students and instructors were challenged by lack of infrastructure of Internet, access to computer labs and academic support. A mentoring system was available as an elective activity and those interested were assigned to mentoring sessions that ran twice per week. However, this system was unstable and sometimes took longer to be arranged. During the study, it started two months after the courses had begun. So, instructors were the central resource for students. We chose to study these settings for distinct reasons. First, because having wide access to the field is an important issue when conducting a qualitative study. Secondly, the units of analysis were representative of what happens typically in a CS0 course in many CS programs in Brazil. It is worth noting that none of the researchers acted as instructors.

The next step was to sample the participants. Freshman undergraduates attending a CS0 course for the first time were our subjects of interest. After a process of recruitment, sixteen students volunteered. Selecting beginners was based on a self-selection approach. They were aware that no reward would be granted. Due to the longitudinal nature of the study, we needed participants who were more likely to be committed and provide insights into the phenomenon. Extensive variation in participant sampling was not a key issue since we understood the richness of each experience. However, variation was met if we consider variance in terms of instructional contexts, participants' age, social and educational backgrounds, fluency in technology and previous involvement with programming.

B. Collecting data

No specific instruments exist to measure stages of interest development [19]. However, since 4PM is a descriptive model, we could build a set of measurement variables from it. The process of measuring interest requires assessing cognitive and emotional components. In our study, to assess the cognitive dimension, engagement became a central construct to interpret its expressions. We observed how participants were interested in learning to code by noticing their reasons for engaging with it, episodes of the engagement by choice, sense of effort when engaged in tasks, and participation (including their willingness to meet and exceed tasks). Knowledge acquisition and changes in learning goals in programming were also taken into account. Lastly, persistency, desire of pursuing exploratory questions, and emergence of new learning strategies were indicators of changes in their states of interest.

To make sense of the emotional component that constitutes interest, we considered which feelings emerged during some interactions within the learning setting. The Geneva Emotion Wheel was used so the participants could report them over time [20]. Interviews, field observations, diaries and reflective field notes were used as instruments of data collection. Three semi-structured interviews were also conducted with each participant. Through these mechanisms, it was possible to build an awareness of how participants were realizing the experience of learning programming. The same subset of questions was repeated to participants during all three interviews which included self-reports of interests, perceptions about disposition of engagement with programming by their own volition, and changes in knowledge and goals in programming. In both studies, data were collected during a period of four months. We have closely followed the classes, performing over 100 hours of field observation (total of both studies) and transcribing more than 1,300 minutes of audio from interviews. On a two-week basis, participants were asked to complete diaries so we could take a picture of their engagement in and out of classes. They expressed how they felt in classes and realized what was happening in the learning setting. Out of class, they were encouraged to report when engaged with course assignments, readings, study groups or mentoring sessions. In this way, the sense of their experiences over time was built from different datasets which constituted a rich data corpus.

C. Analysing data

The goal of the qualitative analysis is to reduce and interpret data to make sense of participant's experience. The analysis has unfolded in two steps. First, to answer "How does interest change over time?", the data corpus was structured into separate clusters based on three points in time: the beginning of the experience; two months later; and the end. We based 'this subject is missing' on the indicators that we mentioned in the previous section in order to measure interest and its changes. As we stated, understanding characteristics of engagement was the central strategy for building a map of expressions of interests, something that was achieved through different ways.

From the researcher's observation notes, aspects such as attendance and participation in class activities were observed. From the diaries, emotions were predominant. When participants decided to engage with programming, we observed the ways. From interviews, based on the vocabulary used, we examined if learning was progressing from domain to specific knowledge. From field observations, researchers tried to understand how they were developing technical skills. Changes in attitudes, study strategies and expectations were also observed. As we already mentioned, participants were asked to self-report changes in their interest in learning programming each time they were interviewed and what made them believe in that. These data were also used for triangulation purposes. After analyzing the experience of each participant, we represented their states of interest on individual trajectories.

The second step of analysis was targeted to answer the question: "Why does interest change?". Looking at data from interview transcripts, diaries and field notes, we examined why each participant wanted to be (or not) involved with programming over time. We adopted the thematic analysis which is a method created by Braun and Clark to identify, analyze and report themes within a data corpus [5]. A theme is something significant related to the research question and represents some level of patterned meaning. However, a theme is not necessarily a matter of frequency and, therefore, it is not dependent on quantifiable measures. The process of analysis followed the six-steps described in [6]. Using an interpretative and inductive approach, we identified the themes of participants' trajectories of interest. The first effort was to familiarize ourselves with the data, reading the interviews over and over. An initial set of codes was generated and iteratively re-examined until data were completely tagged. As Braun and Clark explain [6], "thematic analysis can be a method which works both to reflect reality, and to unpick or unravel the surface of reality". In this fashion, we looked for capturing not only what was explicit in data, but also the underlying ideas that were beyond the semantic content, which is called latent themes. Accordingly, we searched, defined and named the themes. MaxQDA software was used to support this process. Next, the results obtained from the analysis are described.

IV. Contextual Aspects

A. Overall beginners' profile

Both undergraduate programs are held in a campus created to bring opportunities to people who struggle to access the higher education system, mostly because of geographical and economic issues. Over the years, we followed why students have chosen those programs. Sometimes it was a matter of convenience. Computing is not always their first calling. They enjoy technology, but they are not familiar with it. After failing in other areas, not related to technology, some students decide to try these programs just because they feel connected to daily technologies. Sometimes they make the choice by looking at the IT market, due to influence of family and friends or just because of pleasant experiences with games. Most of the freshman students come from public schools[2] reporting problems related to their background in math.

B. The group majoring in management information systems

In this unit of analysis, we observed the experience of ten freshman students. Four females and six males volunteered. They had no previous experiences with programming. The average age was 20 years old. Some reasons reported for choosing this major were pleasant experiences with games, social media and mobile technology, and influence of relatives. Most of them were not sure about their professional choices. Regarding the instructional setting, the instructor was an experienced coder. He has lectured on advanced programming courses for the last ten years. He wanted students to take more control of their learning process. So, he did not adopt a structured system of tasks, nor did he keep an attendance record. Beginners had no assignments to accomplish outside class. The instructional context was based on a top-down approach from which students were provided an overall view of content without explanation of all components that make up the subject. Classes ran in a computer lab containing one machine per student. Students practiced new concepts in programming after a lecture, most of the time solving simple problems. Pedagogical practices like group work and challenge-based activities, or constant feedback were not observed. As a final project, students developed a small information system of their own choice.

C. The group majoring in a teaching degree in CS

For this unit of analysis, six males volunteered. They were between 18 to 23 years in age. This group of beginners was more heterogeneous in terms of educational background and experience with technology. Two of them had some technical knowledge regarding some CS areas, like networks and hardware. But they did not know much about coding. Before starting college, another beginner attempted to learn Python by himself. The others were just curious about technology and sometimes they used to read general information related to it.

The educator was involved with outreach programs related to CS education. He had been lecturing in CS0 courses for the last four years. Over the years, he tried to improve his teaching practice by seeking *forms* to incorporate motivational strategies, such as *gamification*. He struggled to sustain certain initiatives in a constant pace so he gave up on them. He decided to adopt a teaching process familiar to what beginners recognized from high school: a bottom-up model of instruction.

One by one, each subject was taught in a detailed way. At each class, the instructor went a little deeper, bringing more details of how some structures worked in Python. The instructor was aware of how challenging it could be to learn initial concepts in programming. Most of the classes ran in a regular classroom without computers which made students solve problems without assistance of a coding environment. The course had a structured system of tasks which were delivered in a constant pace. The instructor planned the tasks to be small and ascending in terms of complexity. Assessments were based on several small tests and a final project of a simple text-based

[2] In general, in Brazil, the public education system is known for producing poorer learning outcomes than the private one.

game for beginners. The instructor made an effort to help students promptly ascertain their learning outcomes.

V. RESULTS

In this section, we present a glance of how the interest of the two groups of beginners have changed. Using a storytelling narrative, we explain why their willingness of being engaged with coding was affected.

A. The beginners majoring in management information systems

How their interests changed

Participants' interest unfolded in six different ways that we named as trajectories (In Fig. 2, trajectories A to F are labeled by color. On the left side, codes to participants are noted). Trajectories A, C and F were observed in the experience of one participant; trajectories B and D in two participants, and trajectory E in three participants. Considering the four phases described by Hidi and Renninger framework [9], at the end of the course interest of five participants had evolved to phase 2 (P011, P012, P014, P010 and P015). Two evolved to phase 1 (P016 and P017). Three beginners lost their interest altogether. (P008, P009 and P013). The reasons that participants engaged were mostly founded on external aspects (such as earning good grades or doing well in order to apply for a scholarship). In none of these trajectories interest evolved to an individual level. Six participants did not demonstrate any signs of interest for learning programming before initiating the course. They never had any curiosity or tried to engage by themselves. Due to this factor, in trajectories B and D, the starting point of interest fell in phase 1 (triggered situational interest).

We describe one of these trajectories to illustrate how interest shaped a student's engagement with programming. We present a small number of phrases uttered by the participant identified as P012 (called Maria on this narrative): "Before Maria begins college, she mentioned some of her attempts of learning programming for her own sake, looking for online courses and friends that could help her to get started (researcher's field note)." This participant turned her attention to programming before starting college. When the course began, her situational interest in learning this content was already triggered. During the experience, she did not increase the amount of time studying programming nor diversify learning strategies. Her motivation to be engaged was mostly based on achieving good grades. According to 4PM, at the initial phases of interest development, the source of engagement with the object relies on external aspects. Also, we realized that she did not develop an autonomous behavior with regard to her learning process or set exploratory questions to pursue.

These aspects are expected in later stages of interest. However, unlike other participants, Maria did not disconnect from learning programming completely, even when some difficulties took place: "Even not knowing how to code very well, I see programming as an amazing thing. I persist because it's impossible to give up. It's fascinating, even if it's something so tough to learn (excerpt from her first interview)." After a while, frustration became a constant emotion pointed out in her diaries: "I'm interested in programming at the same way, but I feel discouraged when I don't know how to apply content that I thought I had learned by myself at classes (excerpt from the second interview)." As the time went by, she kept recognizing the value of learning programming but she lacked technical knowledge and basic skills.

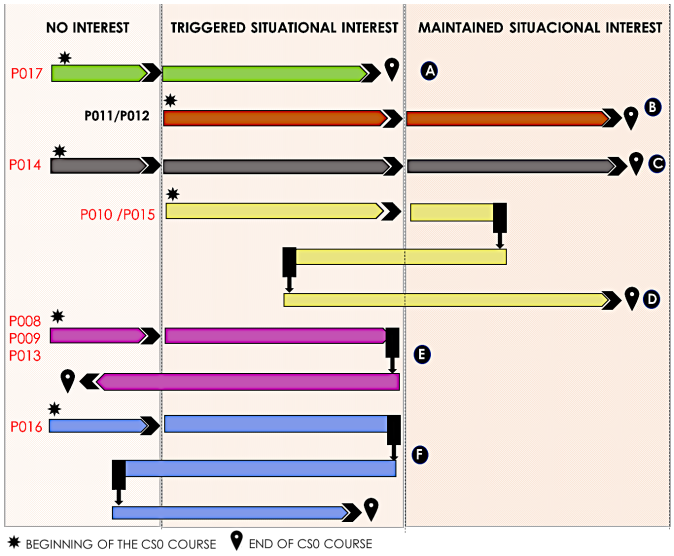

Fig. 2. Interest development of the participants majoring in management information systems

As Maria expressed, she missed support and guidance to learn. These circumstances adversely affected her willingness to engage. At the end of the course, her interest did not reach the individual dimension, but she was still interested in learning programming. In part, because of a personal effort of being connected to it, most of time Maria was reframing her expectations about what in programming could please her in the future. The nature of her interest remained situational during the entire semester. This said much about how she took action during the course.

Why their interests changed

In these trajectories, different types of transitions occurred. Situational interest was activated mostly because of circumstances that acted as triggers. At the first class, the instructor Skyped in a former student to speak to the class. This student was working at The New York Times as a developer and shared that "he had been in their shoes once". Because participants come from public schools and a small town with few career opportunities, they did not realize how far they could advance being a coder. Being aware that "one of them" made it provided a large impact to move them to a "state of interest" in learning programming. This circumstance was an opportunity to demonstrate this content as something possible that could fire their curiosity about it. Because they did not have experiences with computing in high school, participants were excited to attend the programming classes. Beginners felt "inspired" to learn coding when they coded a "Hello world" program. Because of this experience, some of the participants borrowed books from the library and others tried to improve their domain knowledge about programming and Python by searching specialized information on the internet.

When investigating what might have influenced the regression observed in some trajectories, some circumstances worked as a "waterfall" effect. Looking at the environment, we noticed that these circumstances were not appealing and engaging. Using the top-down approach to teach programming was not effective with some beginners. The sense of being exposed to the same content was an inhibitor to engagement with programming. Because participants were taught many leading structures in the beginning, they missed the fact that new structures and knowledge were yet to be introduced. As the instructor moved deeper into each structure and provided new information related to Python, participants felt that nothing new was forthcoming. Participants were not motivated to engage with classes and this was reflected in their engagement outside of classes. Beyond the fact that participants did not expect novelty in terms of new content, the instructor used the same pattern of tasks in classes. For example, the instructor worked on trivial domains, solving small problems in the context of restaurants and banks. The codes progressed, class by class, to larger and more complex systems. Participants often noted they were working on the "same tasks" even though they were not. Because the teaching environment became something predictable, it was no longer interesting and a disposition to engage with programming was being lost.

The absence of a structured system of tasks and assignments, combined with the instructor's expectation that they could learn by themselves, was a strong negative force for participants' interest. We noticed that engagement relied heavily on personal effort that did not happen frequently. Not only because they did not have substantial knowledge in programming to set an agenda for learning by themselves, but also, because without external stimulus and guidance, participants were not encouraged to engage with the content. As long as participants did not develop a routine for studying programming, it was realized that difficulties were taking place when it was too late. There were very few reports associating non-engagement with difficulty to perform the task of programming. Difficulty did not emerge by itself as a strong inhibitor compared to the weight that other contextual factors had. However, some beginners were so disengaged that they did not take on their tasks, nor did they make sense of its complexities and their own problems. We realized that some participants were more skilled in programming than the classroom average. For those students who had to follow the classroom' pace of learning, this condition was an inhibitor for their interest. Even though their interest did not regress, it was not boosted to the next level.

When observing the reasons of why participant's interest was pushed forward, we noticed other forces. Two months after the course began, students were offered the opportunity to participate in sessions with a mentor. As soon as some participants started attending mentoring sessions, novelty came up in the form of a new pedagogical design, based on group discussions, guidance and a structured system of tasks. However, for some, this factor was neutral at that point since their interest had already dropped. Working on the final project was an opportunity offered to beginners almost at the end of the course. Thus, the project carried different meanings to participants, whether it was authorship, a challenge or a sense of working on something that they care about. Many participants referred to the project as noteworthy and also as a mechanism of self-assessment which influenced their disposition of engaging with programming. Furthermore, the project created a new practical experience that provided learning by doing. However, like the mentoring system, this was a neutral factor for some students. The educator was another driver of positive force. Students recognized his passion for teaching coding and were affected by this demonstration.

Interest development was influenced by external and individual factors. We observed both of those factors and how they played a relevant role to explain why the engagement of some beginners went away little by little. Their habits acquired from high school could not be replaced immediately and most of participants reported problems with time management. While observing the learning experiences we realized that some beginners developed a belief that they were not able to successfully perform tasks. This construct is known as self-efficacy and it is not about what we know, but how we judge our abilities of doing something [2]. Low performance at other courses, especially those related to mathematics, also helped them to reinforce their beliefs in programming. The worst case scenario occurred through trajectory D. A participant in this trajectory lost confidence in being able to learn programming and as a result, his interest for learning it. We observed trajectories where participants' interest reached phase 2 and stayed steady. This explains why interest did not regress as in other pathways. We found when facing difficulties some participants were capable of looking for new forms of studying or adjusting their goals and expectations in programming to retain their engagement. They had more capacity to adjust their behavior in the face of external circumstances.

B. The beginners majoring in CS

How their interests changed

Participant's interest unfolded in five trajectories of interest (In Fig. 3, labeled as letters G through K). Except for the one participant, who owned trajectory G, all of them had their situational interest for coding triggered before the course has begun. Gameplay experiences and relative's influence were identified as the main triggers. After the end of the semester, three participants maintained their situational interest (trajectories G, H and I). Indeed, these participants' interest was evolving to the individual dimension.

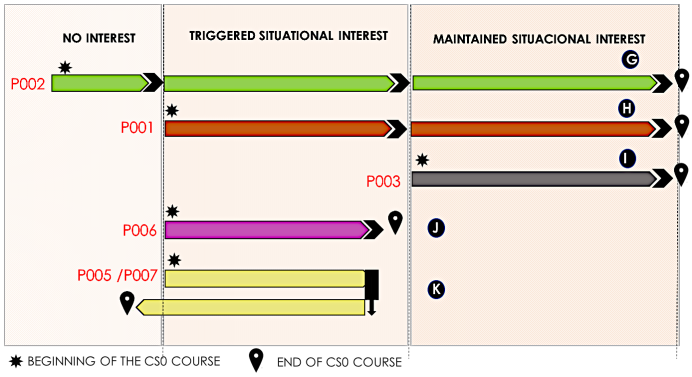

Fig. 3. Interest development of the participants majoring in CS

They were more curious about programming than before. They tried to explore it as attempts to meet and exceed tasks: "I searched how to develop a game in Python. I've already seen some videos but there's too much information I don't understand. It's difficult. […] During the mentoring sessions, I asked my mentor to explain to me things that I don't see in the classroom. I wanted to learn more, extra information" (P002, trajectory G - excerpts from his second interview). Those three participants were more likely to independently engage. However, because they did not have enough knowledge, their engagement was based upon support and incentives that did not always happen. Their interest was towards phase 3, but it did not cross that border. In trajectory J, interest of P006 stayed the same during the entire semester. He occasionally engaged outside classes. The interest of the two other participants was lost at the end of the course for the reasons we explain further (trajectory K).

Why their interests changed

In general, all participant's interest was particularly affected in a positive way by factors surrounding the instructional context. The teaching approach (slow paced, detailed and structured) was familiar to beginners. It retained the way they were used to being taught in school. By adopting this instructional design, the instructor did not push students to immediately adapt this learning approach of which they not capable, something that could have added complexity to the learning process. At first, beginners were feeling optimistic about learning how to code. This optimism was an important incentive to moderate their vulnerability in facing the learning environment. As they relaxed, the students were positively inclined to engage. The instructor delivered content through a clear and comprehensible way. He adopted a visual approach to explain abstract topics by using metaphors and analogies. Because they were able to successfully follow what occurred in classes, students developed a sense that acquiring coding skills could be possible for them. This approach allowed them to feel successful. Awareness of learning was a facilitator to sustain interest. The educator was another facilitator. He took action on beginner's motivational states by encouraging and supporting them as much he could over time. He intentionally demonstrated enjoyment and his own passion for programming. Assignments were planned to be achievable. As the tasks were short, beginners could experience frustration on a smaller scale. Working on a game as the final project was extremely engaging to some. For those who played video games, that activity was extremely meaningful. But, in general, beginners saw a number of other opportunities the project brought: to explore one's own ideas, to express authorship, face a challenge, spend energy on something concrete in the real world and self-assessment.

Despite all these positive forces, we observed why interest in some trajectories was inhibited (trajectories G, H and I) or regressed (trajectory K). To those beginners who were more skilled (P001, P002, P003 and P005), the way that programming was practiced became unpleasant. Exercises were released every time a new content was introduced. For a long time, this was the only form of training. Students solved daily problems and enjoyed this initially. However, as their technical skills were evolving, new expectations of using their knowledge came up. Usefulness became a relevant aspect they expected to have in a task. In addition, some beginners did not find a proper level of complexity on tasks. They became bored because they always had to solve the same kind of problems. It was simply a matter of rebuilding known algorithms that incorporated new structures. They were willing to make more effort but since the tasks were not appealing, they became caught in their comfort zone. This prevented them from exploring curiosity questions. Interest was also inhibited since a scaffolding system was absent, which would have allowed those participants to engage in different ways. From the beginning, some participants struggled to develop the computational thinking skills. At that point, the mentoring program was not available yet. So, they did not have enough support to overcome difficulties. Their engagement with tasks relied mostly on assistance provided by classmates. This was a severe negative force impacting their interest, especially in lowering their levels of self-efficacy. This factor made P007 (trajectory K) disengaged. As soon as his sense of capability went down, he avoided engagement with tasks because he started to anticipate his failure. Exposing his struggles to others and reliance on friends were also reasons to be ashamed. So, he pulled himself away from that frustrating environment. At the end, his interest was completely lost. Family issues and loss of a relative were external forces disfavoring beginners who own trajectory K. Those influences must be taken into account to explain the regressions observed in those trajectories. Another external force that influenced how participants felt interest was the other courses they were attending. Sometimes they had to disconnect from programming to study other subjects that were even more challenging to them and required an educational background that some of them did not have and had to acquire on their own. They also struggled to manage their time for all tasks they needed to accomplish. Since they felt their own need to succeed, when they failed at the other courses, their belief in self-efficacy also propagated to programming.

P002, P003 and P006 were able to self-regulate. When P002 and P003 felt bored in classes because they had to follow their classmate's pace, they reframed learning goals and study strategies. P002 tried "not to be so efficient", slowing down the way he completed assignments. They looked for extra content to learn by themselves. P006 and P007 struggled too. Differently from P007, P006 did not disengage. In the face of problems, he could regulate his motivational state. He did it by trying to remember constantly how programming was a good thing for him. By doing this, he was reinforcing positive emotions and seeing the environment as more interesting than it really was. In this group, the other courses students were attending also appeared as a negative influence.

VI. DISCUSSION

In this paper, we presented how the interest of Brazilian students in learning programming changed through a CS0 course. As we noticed, all participants had their situational interest triggered. The intrinsic novelty of coding was itself a positive aspect to spark interest. Likewise, listening to others' experiences with programming (vicarious experience) and gameplay experiences were also positive aspects. The multiple changes observed in some trajectories reflect how interest was volatile at the initial phases of its development and responsive to the environment. Understanding interest from this

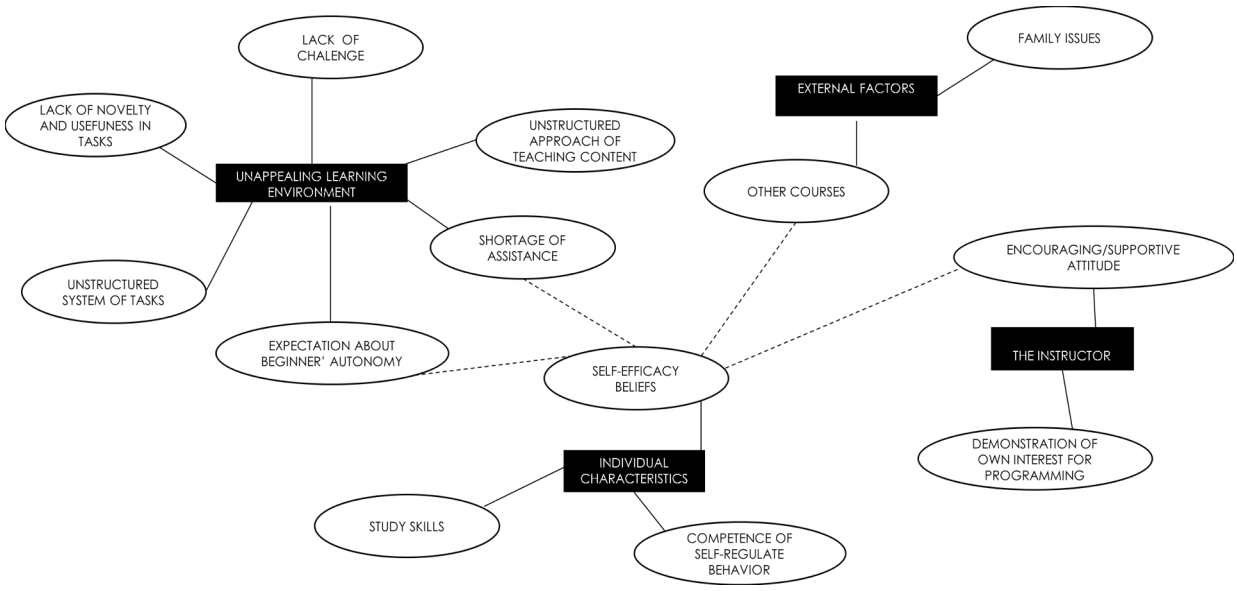

Fig. 4. A thematic map that illustrates the narratives of why participants' interest in learning programming changed over time. Note: Rectangles characterize a theme and ovals, its subthemes. When dashed lines are used, a subtheme is influenced by other subtheme(s)

perspective brought more context to shape our understanding of how beginners experienced learning programming and how diverse the influences are to their interests. However, some themes emerged in both groups. In Figure 4, we illustrate those themes as a map to reinforce that interest development is a complex phenomenon influenced by multiple forces.

Student's awareness of learning emerged as the main trigger to move their states of interest to a next stage. As [18] points out, in earlier phases of interest, because students are still developing knowledge, they want to be told what and how to do. To some participants, complexity was a matter of how they addressed their capacity to learn to code. Observing specific trajectories in which interests regressed, some students built a greater sense of complexity around programming that jeopardized their willingness to engage with programming. As we saw, some beginners were not supported to overcome their initial struggles and feel empowered to learn. Some students did not have the competencies to set goals, properly manage time to perform tasks or look for effective learning resources. In addition, they could not develop new attitudes by themselves (self-regulating). As happened in the group of students majoring in management information system, the instructional design created a challenging setting. As a result, the demand for competencies they had not developed yet caused students to feel powerless and discouraged. Under these circumstances, learning was compromised. Issues related to the program curriculum design deserve to be noted as well. Other courses were influential to both CS0 groups. When failing at math courses, some beginners carried their negative perceptions about their capacity to programming. It took time for both groups to gain access to the mentoring system. For quite a long time, they were required to develop computational thinking skills without proper support and rely on classmates who were also new learners. For both groups, in one or another way, contextual factors strongly impacted students' beliefs of self-efficacy. As we noticed, those trajectories of interest were heavily influenced by human factors (individual characteristics). Since programming is a brand-new content to many, it is important to pay attention to that content to understand what should be balanced in the learning experience to order to sustain interest in its primary forms of development.

VII. FINAL REMARKS

Using the 4PM, we could investigate how and why sixteen students felt interested in programming over time. If we consider how simplistically some prior studies handled this intricate construct, the results discussed in this paper are relevant because of its potential to influence local practices and spark discussion. At this point, it is worth noting that we do not assume that our results can be generalized to a larger population. This is the thinking in how quantitative studies are designed. In fact, a qualitative researcher's intent is to obtain a rich description of a particular context and understanding of participants' meanings and experiences. Unveiling the real world complexity is where the value of this kind of study resides. As generalization is not the goal, transferability might be the concept to be used here, which it refers to how a reader apply understandings and knowledge built from a research to similar contexts or settings [4].

This work also raises questions to be discussed in the future. As an example, how would CS educators strengthen individual factors, such as beliefs of self-efficacy and autonomy over a CS0 course? What kind of training should educators be provided to teach beginners in programming or to work better with their interests? How can we support students to acquire non-technical skills, like developing higher levels of self-regulation? The instructional design of a CS0 course is not a one-size-fits-all solution. Knowing the student better and recognizing his/her limits, along with environmental restrictions, is something we need to take into account for targeting to create a more interesting experience in a CS0 course.

REFERENCES

[1] Anderson, M., MacKenzie, A., Wellman, B., Brown, M., and Vrbsky, S. 2011. Affecting attitudes in first-year computer science using syntax free robotics programming. *ACM Inroads*. 2, 3 (August. 2011), 51–57.

[2] Bandura, A. 1982. Self-efficacy mechanism in human agency. American psychologist, 37,2 (Feb. 1982), 122.

[3] Barron, B. 2006. Interest and self-sustained learning as catalysts of development: A learning ecology perspective. *Human Development*. 49, 4 (October. 2006), 193–224.

[4] Bloomberg, L. D., Volpe, M. F. 2015. Completing Your Qualitative Dissertation: A road map from beginning to end. 3 edition ed. [s.l.] SAGE Publications.

[5] Braun, V., Clark V. 2013. Teaching thematic analysis: Overcoming challenges and developing strategies for effective learning. The Psychologist 26, 2, 1-14.

[6] Braun, V., Clark V. 2006. Using Thematic analysis in psychology. Qualitative Research in Psychology 3, 77-101.

[7] Burguillo, J. C. 2010. Using game theory and Competition-based Learning to stimulate student motivation and performance. *Computers & Education*. 55, 2 (Sep. 2010), 566–575.

[8] Cavaye, A. L. M. 1996. Case study research: a multifaceted research approach for IS. Information System Journal, 6, 227.

[9] Hidi, S. 1990. Interest and Its Contribution as a Mental Resource for Learning. *Review of Educational research*. 60, 4 (1990), 549–571.

[10] Hidi, S. and Renninger, K. A. 2006. The Four-Phase Model of Interest Development. *Educational psychologist*. 41, 2 (2006) 111–127.

[11] Krapp, A. 2002. *An Educational-Psychological Theory of Interest and Its Relations to SDT*. Handbook of Self-Determination Research. (E. L. Deci and Richard M. Ryan, Eds. University of Rochester Press, 2002), 405.

[12] Konecki, M., Kadoic, N., and Piltaver, R. 2015. Intelligent assistant for helping students to learn programming. In *8th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. (May. 2015), 924–928.

[13] Lakanen, A. J., Isomöttönen, V., and Lappalainen, V. 2014. Understanding differences among coding club students. In *Proceedings of the 2014 conference on Innovation & technology in computer science education*. (2014), 159–164.

[14] McGill, M. M. 2012. Learning to Program with Personal Robots: Influences on Student Motivation. ACM Transactions on Computing Education, v. 12, n. 1, p. 1–32.

[15] Pears, A., Seidman, S., Malmi, L., Mannila, L, Adams, E., Bennedsen, J., Devlin, M., and Paterson, J. 2007. A survey of literature on the teaching of introductory programming. In Working group reports on ITiCSE on Innovation and technology in computer science education. 39,4 (December, 2007), 204.

[16] Renninger, K. A., Ewen, L., and Lasher, K. 2002. Individual interest as context in expository text and mathematical word problems. *Learn. and Instr*. 12, 4 (Aug. 2002), 467–491.

[17] Renninger, K. A., Bachrach, J. E., and Posey, S. K. E. 2008. *Learner Interest and Achievement Motivation*. Social Psychological Perspectives, 2008, p. 559.

[18] Renninger, K. A. Working with and cultivating the development of interest, self-efficacy, and self-regulation. 2010. In *Innovations in educational psychology. Perspectives on learning, teaching, and human development*. (Springer Publishing Co, 2010), 107–138.

[19] Renninger, K. A. and Hidi, S. 2011. Revisiting the Conceptualization, Measurement, and Generation of Interest. *Educational Psychologist*. 46, 3 (July. 2011), 168–184.

[20] Sacharin, V., Schlegel, K., and Scherer, K. R. 2012. Geneva Emotion Wheel rating study. Geneva, Switzerland.

[21] Scaico, P. D. 2015. Raising Novices' interest in Learning to Code: A Systematic Review of this Literature from 2000 to 2015. Recife, Brazil. Unpublished.

[22] Schiefele, U. 1991. Interest, learning and Motivation. *Educational Psychologist*. 26, 3 (Nov. 1991), 299–323.