# Hackathons as a Pedagogical Strategy to Engage Students to Learn and to Adopt Software Engineering Practices

Caio Steglich, Larissa Salerno, Thaís Fernandes, Sabrina Marczak,
Alessandra Dutra, Ana Paula Bacelo, Cássio Trindade
Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Porto Alegre - Brazil
(caio.borges,larissa.castro,thais.fernandes,cassio.trindade)@edu.pucrs.br
(sabrina.marczak,alessandra.dutra,ana.bacelo)@pucrs.br

## ABSTRACT

Teaching Software Engineering is not a trivial duty since several pedagogical strategies can be used and sometimes the impact of these on students is uncertain. Hackathons are similar to marathons, however used to produce solutions to solve a specific problem in a short period of time and based on intense collaboration. Educational hackathons aim to promote learning in such an environment. The Undergraduate computing programs of PUCRS decided to use a hackathon as a pedagogical strategy aiming to motivate the students to practice the adoption of software development practices and to work in groups as a means to practice the development of social skills. Therefore, we conducted a case study to investigate: 1) The motivations to students to attend or not attend an educational hackathon, 2) The students perceptions about this hackathon, 3) The Software Engineering practices adopted by students. In this study, we identified factors that may affect students motivation to participate (e.g., improve the teamwork skills), some students expectations about the hackathon (e.g., work in teams), and the practices adopted by the students (e.g., pair programming). Some of our findings include that students enjoy participating in an informal educational environment (e.g., hackathons) to improve their technical skills and to build network with some colleagues. This study can provide insights to teachers that wants to organize some activity than traditional teaching and the students perspective about this kind of strategy.

## CCS CONCEPTS

• **Social and professional topics** → *Computing education*;

## KEYWORDS

Software Engineering Education, Educational Hackathon, Student Motivation, Case Study


*SBES '20, October 21–23, 2020, Natal, Brazil*

ACM ISBN 978-1-4503-8753-8/20/09…$15.00
https://doi.org/10.1145/3422392.3422479

**ACM Reference Format:**
Caio Steglich, Larissa Salerno, Thaís Fernandes, Sabrina Marczak, Alessandra Dutra, Ana Paula Bacelo, Cássio Trindade. 2020. Hackathons as a Pedagogical Strategy to Engage Students to Learn and to Adopt Software Engineering Practices. In *34th Brazilian Symposium on Software Engineering (SBES '20), October 21–23, 2020, Natal, Brazil.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3422392.3422479

## 1 INTRODUCTION

The term "Hackathon" combines the words "hacking" and "marathon" and implies an intense, uninterrupted, period of project development. In Software Engineering, a hackathon is a highly engaging, continuous event in which people produce a working software prototype in small groups within a limited amount of time [7].

Hackathons have been used for over a decade as a pedagogical tool, for instance, to foster students' learning of social skills such as teamwork and collaboration (e.g., [13]) or to stimulate the students' interest to learn something new (e.g., [8]).

Educational hackathons offer an informal environment to support the learning process. These hackathons can offer different formats and duration according to their learning objectives [10].

PUCRS organized an educational hackathon as a joint effort among the technology-based Undergraduate programs (e.g., Software Engineering, Computer Science, Information Systems, etc) to offer the students an informal learning environment to practice the adoption of software development practices, such as problem comprehension and requirements elicitation, and to work in groups and practice the development of social skills, such as communication, negotiation, and conflict resolution.

Inspired on the studies of Warner and Guo [14] and Gama [6], we conducted a case study of this educational hackathon aiming to understand what factors may affect students to participate, what are the perceptions of the students about the event and what Software Engineering practices are adopted by the students.

Our pre- and post-hackathon interviews and observations during the event revealed:

- Why the students attend these events and what motivates them to participate
- What the students expect to learn from these event
- What Software Engineering practices are adopted by students over the event

The lessons we learned in conducting this case study can be of use to researchers as a means to further the knowledge on the topic but mainly to teachers and Undergraduate Program Chairs who aim to organize similar events as a means to promote learning. Also,

to those who want to reflect upon learning strategies to engage students of the XXI Century.

The remainder of paper is organized as follows: Section 2 describes the theoretical foundations of our work. Section 3 introduces the studies that this paper is inspired on. Section 4 presents the educational hackathon and its settings. Section 5 describes our research methodology. Section 6 presents the study findings organized by research question and Section 7 discusses the results. Section 8 introduces the lessons learned from this educational hackathon. Section 9 lists the study limitations. Section 10 concludes the paper with considerations and future work.

## 2 BACKGROUND

In this section, we present an introduction to hackathons as an event and its applications in Software Engineering education.

### 2.1 Hackathons

Komssi et al. [7] explain that a hackathon is a kind of marathon of hacking, meaning that such kind of events promote competition among the participants. Also, the authors explain that a hackathon offers a problematic situation to be solved by their participants, and that this problematic situation can be from different fields, such as Health or Business, or bring multidisciplinary problems such as Transportation or Smart Cities [7].

Hackathons may have several purposes, such as Scientific, where specialized scientific communities such as e.g. Bioinformatics or Astronomy organize hackathons aiming to grow interest and membership, and to create new or to expand existing technical artifacts [9]. Also, the research community explores this events with different purpose, such as to understand the interaction of the participants and their social ties [13].

There is no single formula for conducting a hackathon, Howewer, Frey and Luks [5] argue that a hackathon is often organized into three main phases, namely: preparation, development, and post-hackathon. The preparation focuses on in the efforts to organize the event. The development is composed of 4 stages–identifying the problem, analyzing alternative solutions, prototyping, and pitching the results and receiving feedback. Last but not least, the post-hackathon usually refers to the intention of the participants to continue acting in the proposals they presented in the pitch.

More specifically, the 4 development stages aim to:

(1) Identifying the Problem:
   - (re-)define the field and context of potential customers,
   - analyze and identify relevant needs, pain points and real problems of the customers,

(2) Analyzing Alternative Solutions:
   - identify alternative solutions for each relevant problem.
   - identify and define the customer value proposition (what is the benefit for the customer? what does she gain with this solution? which does relieve the customer's pain points?) for each alternative solution

(3) Prototyping:
   - create a prototype which shows the key (minimum set of) features and helps presenting the concept of the solution (the prototype can be an app, but also a paper-based model, box or anything else) for each alternative solution

(4) Pitching & Feedback:
   - test each prototype by presenting it in a short pitch and collecting feedback and ideas for improvement from potential customers (ideally) and/or their representatives.

### 2.2 Hackathons in Education

Some authors report having used hackathons for pedagogical purposes. Calco and Veeck [2], for example, report on the definition of an area-based model of hackathon named "Markathon". It aims to improve the teaching of Marketing students through a competition. The authors argue that such kind of competition fosters the improvement of skills such as thinking critically, working in teams to complete a task, communicating effectively, identifying and solving problems by apply marketing concepts, and innovating.

Fowler [4] presented a case study about a gaming hackathon which goal was to teach technology to students. Participate to gain knowledge, network with colleagues and professionals, and promote social impact were among the top reported reasons to join a game-based educational hackathon according to the students [4].

Decker et al. [3] presented the concept of community-based hackathon model. This kind of educational hackathon aims to solve a problem pointed out by the community that these students are engaged in. The authors identified several motivational factors for students to participate in such kind of hackathon, such as:

- To collaborate with and to learn from each other
- To connect students with real organizations and develop real products
- To students perceive how much they know about development
- To meet new people with similar interests
- To feel fulfilled when finally done
- To code with awesome people and learning new stuff

The study by Decker et al. [3] also reports on strategies to attract students to this activity (e.g., attract students to develop social solutions to their community), given that it is not enough to just organize this event but to make students feel comfortable to participate, reaching the motivations of these students and thus bringing them into the competition.

## 3 RELATED WORK

This study is inspired by two studies by other researchers, to understand in-depth the benefits of a hackathon in Software Engineering education (Table 1).

The study by Warner and Guo [14] examined student perceptions of college hackathons, dialoguing about their motivation, how they learn, and the factors that may discourage with these students. They had observed and interviewed six students (3 male and 3 female) in three moments, being before the event, one week after, and one month after. This study from Warner and Guo [14] is different from ours because we investigated an educational hackathon with students from several computing-related undergraduate programs (e.g., Software Engineering, Information Systems), not only for computer science, and we observed the participation 56 students and interviewed 15 students.

The study by Gama [6] investigated civic hackathons to understand how teams in these competitions are tackling the different

**Table 1: Setup of the Previous Studies**

| | Warner and Guo [14] | Gama [6] |
|---|---|---|
| Setup | Academic Hackathon | 3 Civic Hackathons |
| Research Questions | RQ1. Why are students motivated to attend college hackathons?<br>RQ2. What kind of learning environment do these events provide?<br>RQ3. What factors discourage students from attending? | RQ1. How are functional requirements being identified for applications in civic hackathons?<br>RQ2. Is there evidence of any concerns with software quality during the construction of their civic app?<br>RQ3. How do teams create and manage their tasks in civic hackathons?<br>RQ4. How are the civic apps being released and maintained? |
| Method | Case Study | Survey |
| Findings | Perception of the students in Hackathons<br>(Des)motivation of students in Hackthons | Perception of the participants about their software engineering practices. |

activities in their software development process. They observed 3 hackathon events from different countries, surveying 123 participants. The qualitative data from the study [6] demonstrate tendencies from different activities in their software development process. This study from Gama [6] is different from ours because we investigated a educational hackathon and not a civic hackathon, and we do not surveyed the participants or used quantitative data, but we observed these students to explain in detail what practices or technologies they have used on this event.

## 4 HACKATHON

AGES department organized this hackathon as an educational strategy to engage students to adopt and learn software development practices, being these hard skills (e.g., good development practices) and soft skills (e.g., communication with their teammates).

### 4.1 The AGES Department

The AGES department provides an environment dedicated to teaching and learning technology from practice, is linked to the Software Engineering Undergraduate program from the University A [15].

This department, which is dedicated to providing a practical learning environment for students, aims to:

- Provide students with an experience of real project situations in an environment focused on learning, with real projects and clients.
- Enable students' contact with organizations and companies
- Assist students in building their project portfolio.

This department is responsible for teaching 4 courses over the Software Engineering undergraduate program in the University A, with each course consisting of 120 hours dedicated to a real project. Every Course presents a role to the students attempt, being these:

- Course 1: The student will experience the role of a developer and a tester.
- Course 2: The student will experience the role of a Requirement Analyst and a Database developer.
- Course 3: The student will experience the role of a Software Architect and a Quality Analyst.
- Course 4: The student will experience the role of a Project Manager and a facilitator of agile practices.

At the end of this experience, the student will be able to defend their Undergraduate thesis, in which he will be able to present a report on the projects he participated in during the realization of the 4 courses carried out in this department.

The professors responsible for this department realized that in addition to these courses provided to encourage students' practical experience, they could add another opportunity, testing the use of an event in which students could in a short time try different roles and activities to interact on a real project, proposing a solution to this hackathon problem.

### 4.2 Setting

This hackathon was designed to last 30 hours, being 3 hours for introduce the students to the problematic situation, 24 hours of software development and 3 hours to pitch their proposals. The event received 74 registrations and among these, 66 students attended to the hackathons. The physical space was the same structure that students usually use during their courses (at the university) and even the laptops provided by the organization of the event unless the student wants to use their own laptops. The hackathon organizers offer to students 3 rooms to work, and every student could ask for a laptop if they do not bring their personal one.

The event had three basic rules:

- The teams should be composed of 4 students from the Software Engineering course, and 2 from another course (Computer science, Information systems, among others).
- Students should not leave the hackathon venue until the end of the event.
- The event would only award the first, second, and third best-placed team.

The problem is about disaster control and the students should propose solutions to solve it, considering before, during, or after the risk situation occurs. Event organizers invited members of a startup that works creating software to help with disaster problems, where students could consult with them to further understand the problem. These guests made an initial presentation explaining the problem. The planned schedule is presented in Table 2.

When organizing this event, the organizers aim to create a differentiated learning environment, where students can practice, compete in a healthy and fun way. However, what was expected of students participating in this hackathon was:

- Use of knowledge acquired in technology-based programs to conduct and elaborate the solution of problems;

**Table 2: Event Schedule**

| Time | Activity |
|---|---|
| 8 AM - 10 AM | Students' reception with coffee |
| 10 AM - 11 AM | Track / problem presentation |
| 11 AM - 12 PM | Assembling the teams and defining the software solution proposal |
| 12 PM - 1 PM | Lunch |
| 1 PM - 8 PM | Beginning of the solution elaboration stage |
| 8 PM - 9 PM | Dinner |
| 9 PM - 8 AM | Finishing the Solution |
| 8 AM - 9 AM | Breakfast |
| 9 AM - 10 AM | Pitch presentation elaboration |
| 10 AM - 12 AM | Award the best teams |

**Table 3: Prizes Provided in this Hackathon**

| Placement | Prize[1] |
|---|---|
| 1st place | - A short course at a renowned computer science company<br>- A voucher worth approximately US$ 65 in Amazon digital library |
| 2nd place | - A short course at a renowned computer science company<br>- A voucher worth approximately US$ 40 in Amazon digital library |
| 3rd place | - A short course at a renowned computer science company<br>- A voucher worth approximately US$ 25 in Amazon digital library |

- Development of a complete and functional application, i.e., a prototype, adhering to the proposal developed in the "solution proposal definition phase";
- Development of soft skills for teamwork, communication, and leadership.

The teams were evaluated by the solutions they produced and the process that led them to this solution. The evaluation criteria were: technical feasibility, creativity, the evolution of the prototype during the event, the satisfaction of the proponent (stakeholder) with the solution.

The judges of this hackathon were: 2 members of the startup that proposed the theme of the problem, being them both graduated (former students from the undergraduate program on Information Systems from University A), 2 professors of Software Engineering Undergraduate program (both PhDs, and working in the program coordination), and 1 professor from Computer Science Graduate program (professor and researcher). The award was given to students from the three best-evaluated teams, according Table 3.

## 5 RESEARCH METHOD

We investigate this academic hackathon based on the research questions inspired by Warner and Guo [14] and Gama [6]. These questions are presented in Table 4.

[1]The Brazilian minimum wage is about to $ 196,80.

Aiming to answer these research questions, we decided to follow a qualitative approach, using the Case study method proposed by Runeson and Höst [12], collecting information by interviews and observation during the event.

In the RQ1 and RQ2, we become inspired by the study of Warner and Guo [14], who had interview some students from a hackathon in three moments, 1) before the event, 2) a week after, and 3) a month after, to understand the students' opinion in participating from the hackathon that they observed. In the 3 phases of interviews, we used the same questions of the study of Warner and Guo [14]. The students profile are presented in Table 5.

In the first interview round, before the hackathon, interviewed 15 students that registered for the hackathon, being 9 from the Software Engineering (SEng) program, 3 from Information Systems (IS) program, 2 from Computer Science (CS) program, and 1 from Computer Engineering (CEng) program. This is approximately 20% of the students that participating in the event, chosen for convenience. We asked these students the questions presented in Table 4, according to Warner and Guo's [14] questions protocol.

In the second interview round, just a few days after the event, we interviewed 12 from the previous 15 students (3 did not participate from the hackathon), being 8 from the Software Engineering course, 2 from the Information system course, and 2 from Computer Science course. We asked to these students the questions presented in Table 4, according to [14].

In the last interviews round, a follow-up one month after, we interviewed 10 from the 12 previous students (2 do not want to answer this follow-up interview), being 7 from the Software Engineering course, 2 from Computer Science course, and 1 from Information system course. We asked to these students the questions presented in Table 4, according to [14].

These interviews were conducted by two researchers, where every of the students were interviewed by one of those two. In all interviews, we recorded the talk with the participants' consent and later transcribed it. We also took notes during the interviews to support the recording of interview highlights. Data analysis was conducted based on the content analysis proposed by Bardin [1], to understand the response of interview respondents.

In the RQ3 inspired in the study of Gama [6] who investigate civic hackathons (it means, producing software to government or population), where they surveyed the participants of 3 hackathons. Different from Gama [6], we do not survey the participants but observed how they work, and we took notes from every activity, besides, our study aims to analyze an educational hackathon. This observation was conducted by 5 researchers, being 2 undergraduate students, 2 master students, and 1 Ph.D. student. This observation was along with the whole competition, the Ph.D. student watched the 30 hours of the competition and the other 4 researchers watched up to 12 hours. We took care to do not interrupting students in their activities and not asking questions that could divert their attention from the problem. The topics we observed are presented in Table 4, according to Gama [6].

All data collected by 5 researchers, were carefully analyzed by 2 researchers and were reviewed by another 2 that do participate of the collected data, to ensure that the data answers the research questions adopted.

**Table 4: Research Questions**

| ID | Research Question | Based in | Data Collection Method | Specific Questions |
|---|---|---|---|---|
| (RQ1) | What factors may affect positively or negatively the students motivation to participate? | Warner and Guo[14] | Interview | Before the Hackathon:<br>- What do you think the point of hackathons are?<br>- Why do you want to go to this upcoming hackathon?<br>- What do you hope to gain from attending the hackathon?<br>- Do you already have project ideas or team members?<br>- Are you nervous about any aspects of this hackathon?<br><br>One Week After:<br>- Did anything unexpected happen?<br>- What (if anything) has changed about your initial perceptions of hackathons?<br>- What was most memorable about this hackathon?<br>- How did attending this hackathon affect your confidence in your abilities?<br><br>One Month After:<br>- What criticisms (if any) do you have of hackathons after attending this one?<br>Do you remember any discouraging moments? |
| (RQ2) | What is the students perceptions about the Hackathon? | Warner and Guo[14] | Interview | Before the Hackathon:<br>- Who are you hoping to learn from at this event?<br><br>One Week After:<br>- What project did you worked on?<br>- Who did you learned from this hackathon?<br><br>One Month After:<br>- What new skills did you learn at the hackathon?<br>- How do you think you will use these skills in the future? |
| (RQ3) | What Software Engineering practices are adopted in an intensive-collaborative learning environment? | Gama [6] | Observation | During the Event:<br><br>- How did students organize themselves into teams?<br>- How did the team organize themselves to work?<br>- How did the team discussed the problem and the solution?<br>- How did the team worked on the software requirements?<br>- How did the team handle the architecture of their solution?<br>- How did the team carry out the coding / programming?<br>- How did the team use version control?<br>- How did the team test the developed application?<br>- How were project tasks created and managed?<br>- How do team students collaborate to carry out the work?<br>- What tasks did the teams have difficulty to conduct?<br>- Did the teams finished their project?<br>- How did the teams deal with conflicts?<br>- Did the team consult the technical mentors or support team during the event? |

## 6 RESULTS

This section presents the results organized by research questions.

### 6.1 Factors that may Affect the Students (RQ1)

**Factors that may Motivate the Students**

In the students perspective, a hackathon can have several purposes, such as **an opportunity to learn and improve skills** (*"This event brings several concepts that we learned and makes me practice to*

**Table 5: Interviewed Students**

| Student | Gender | Program | Inbound semester | Practical Course | Interviewed Before the Event | Interviewed One Week After | Interviewed One Month After | Current Job Position |
|---|---|---|---|---|---|---|---|---|
| S1 | M | SEng | 2017/1 | Course 3 | X | | | Data Scientist |
| S2 | M | SEng | 2017/1 | Course 3 | X | X | X | Software Developer |
| S3 | M | CEng | 2013/2 | - | X | | | Software Developer |
| S4 | M | IS | 2018/2 | - | X | | | Software Developer |
| S5 | M | SEng | 2017/1 | Course 3 | X | X | | Software Developer* |
| S6 | M | SEng | 2017/2 | Course 2 | X | X | X | Unemployed |
| S7 | M | SEng | 2017/1 | Course 3 | X | X | X | Software Developer |
| S8 | M | SEng | 2017/1 | Course 3 | X | X | X | Software Developer* |
| S9 | M | SEng | 2015/1 | Course 4 | X | X | X | Software Developer* |
| S10 | M | SEng | 2017/1 | Course 3 | X | X | X | Software Developer |
| S11 | M | CS | 2017/1 | - | X | X | X | Quality Analyst |
| S12 | F | IS | 2018/1 | - | X | X | X | Junior Researcher |
| S13 | M | IS | 2017/1 | - | X | X | | Data Scientist* |
| S14 | F | SEng | 2018/1 | Course 2 | X | X | X | Junior Researcher |
| S15 | F | CS | 2019/1 | - | X | X | X | Software Developer* |

*These students are interns in Software Industry.

*deal with my technical limitations"* - S1), **networking with other people** (*"The team activities support us to meet new people, and I know some people that come to hackathon only to Network"* - S5), **improve their teamwork skills** (*"I believe the objective is to work in teams, with people from other courses and work on an idea from 24 hours"* - S9) and **the social impact of the solution** (*"I think hackathons are better when the problematic situation is about social content, understanding the user journey"* - S8).

The students also pointed out particular reasons to participate in a hackathon, such as **working with friends** (*"I think it is cool to work with people that are my classmates, and it promotes knowledge exchange"* - S1), **curiosity** (*"Last year I could not participate, but this year I decided to try, developing some project during the 24 hours"* - S9), and **fun** (*"I am seeking for fun, I do not wanna the prize but to programming and eat pizza"* - S10).

The students have different expectations of what they could receive in a hackathon like this, such as **gain experience** (*"I am always seek to gain experience, knowing new ideas and learning how to solve problems"* - S1), **knowledge** (*"I wanna go deep in my knowledge and learn even more with other people"* - S6), **friendship** (*"I wanna get new friends and be recognized by them"* - S13), and **the prize** (*"Last year we earned sunglasses (I do not use mine, by the way), but I believe the shopping voucher most interesting this year (amazon voucher)"* - S8).

**Students Feelings and Fears**

The students' have some fears and suspicions about the event, for example, **when they are in a hackathon for the first time** ("*I was surprised because. I do not know why I never participated before"* - S9), or they had **other experiences** (*"I had already participated in others, but I see that it counted a lot on the product being delivered and working, having a well-developed front-end"* - S1).

Also, some students also show nervousness before the event, mainly because they do not feel confident in their **technical skills** (*"I am afraid to do not have enough knowledge and be dependent on colleagues"* - S3), **limited event time** (*"Do not know the selected technology and the short time to develop"* - S6), and **not having good ideas** (*"I am afraid to do not get a good idea to the solution"* - S7).

The impact of the hackathon on student confidence is usually **positive** (*"It affected my confidence a lot, I changed jobs a week after the hackathon, because I realized that I was ready for new challenges"* - S7), but the students recognize that they need to **improve some skills** (*"I realize that I have improved my soft skills a lot but my hard skills are not as good as I would like them to be"* - S6).

We asked students what skills they feel they have improved by participating in this hackathon. Most reported that have **improved their technical skills** (*"I improved in HTML and CSS, I also developed in python, learning about python web"* - S11), however, some also reported that **management and organization** were also learned throughout the competition (*"I learned how to manage the team and how to develop a project in a short period of time"* - S9).

Students report that they believe they can use the acquired knowledge and skills acquired in future situations, such as in the **job market** (*"I think this will apply to the job market, due to the experience of knowing how to make code under pressure due to time, and knowing how to delegate tasks, functions"* - S12, and *"In my internship, why it was basically doing a project and how we are going to work with projects will be useful"* - S15).

Students also reported some criticisms, such as the **evaluation process being clearer** (*"I think they should give the general classification for us to know what was evaluated and how each thing was evaluated in a more transparent way"* - S2) and **not allowing beginner students** (*"I think they should put a minimum semester for people to enter because taking on very inexperienced students can bring negative points to the competition"* - S6).

## 6.2 Students Perceptions about the Hackathon (RQ2)

**Students Strategies to Build Teams**

Students organized into **closed team** (*"It is the same time from the last year. We will develop according to the problematic situation"* - S8), **almost closed team** (*"We have an almost complete team, we do not make anything, but we are talking about this hackathon, and defining the roles"* - S4), or **opened team** (*"I have some ideas of classmates that can work with me if they accept my request."* - S9).

**Students Expectations to Learn**

Before the event, students explain that they sought to learn skills in this hackathon, such as **software architecture** (*"I will try to do a good software architecture, but I also wanna do good business decisions focused on the proposed activities"* - S1), **software development** (*"I wanna know more about front-end, highest technologies, frameworks and work in teams"* - S11), **work in teams** (*"I hope to better understand the market and how I work with different persons and ideas"* - S15), and **communication** (*"It is a way to improve my communication in a short time. I do not expect to learn more technologies but to learn how to communicate"* - S6).

The projects that the students worked on were the most diverse, but in general or the teams focused on **websites** (*"It was a website where we put some tutorials to teach people how to act in emergency situations and there was a session where they had questions regarding the tutorials to ensure that the person learned the content"* - S10) or **applications** for mobile devices (*"We developed an app that would help preventively because it would show the risk zones"* - S12).

The students' learning through some elements, such as **technology** (*"I learned Python and researched with colleagues about frameworks that could help with the solution"* - S11), **team work** (*"Maybe I learned to work better in a team"* - S13), and **time organization** (*"That 24 hour is much time, but it is short too. It is hard to organize the tasks in the schedule, doing parallel tasks"* - S6).

**Students Perceptions about the Event**

Also, some points surprised students, such as **technology** (*"We tried to use new technology, but after this, we learned that it did not work as well for iOS as it did for Android"* - S8), **team performance** (*"I did not expect my team to develop a project so well"* - S13), and **diversity of the backgrounds** (*"Something that surprised me is that in my team we had a colleague who studies Production Engineering and that guy did very well"* - S9).

Several moments were considered positive by the students, where practically everyone considered that the most memorable moment was the **pitch** (*"I think it was during the pitch because we asked everyone in the audience to use our system and it worked"* - S8), where the teams were able to present the developed solutions. Also, students have enjoyed **working in teams** (*"At dawn, when we were very sleepy, we started makes jokes with our group colleagues to laugh and stay awake"* - S12).

## 6.3 Software Development Practices Adopted by Students (RQ3)

**Students Dealing with the Software Requirements**

Initially, the organizers presented the problem situation of this hackathon in an auditorium that is located in the same building where the event would take place. After this presentation, the students went to the entrance hall of the event building, where they began to communicate with colleagues. **Some teams were formed**, usually when they repeated the **same ones from the last edition** of the event, and **some were quickly formed at the beginning of the event**, thus meeting new colleagues.

Once gathered in teams, **students begin to talk about the skills of their members**, some with laptops writing down the roles and technologies that each one would perform. Some students decided to go and **talk to the organizers and proponents of the hackathon problem** issue, to understand the solution well, while **some went directly to start designing a solution** from what they understood from the presentation. While the students discussed the problem situation, it was as if they used other commercial tools as a benchmarking, discovering which features would be interesting that theirs also presented. One team researched real cases in which the system could assist by reading art.

To understand the problem and create the software requirements of the solution, **students adopted several techniques**, such as User stories mapping, Empathy maps, Personas, Business model canvas, User journeys, Product vision box, Desk research. The students used to sit at tables where they could talk and **use a flipchart with postits** to better visualize the elaborated requirements.

**Students Concern about Software Quality**

There were also several resources used by the students, with **some teams defining the solution architecture only after the requirements** and **others had already decided before the competition even started**. Some of the resources chosen to be used in the development of the solutions were: Android Studio, Github, Figma, DBeaver, MySQL, Firebase, Heroku, Postgree, PyCharm, Visual Studio, Trello, Photoshop, IntelliJ IDEA, Slack.

In the code production step, we perceived three common strategies that sometimes changed throughout the hackathon, these being: **Pair programming** (used most of the time), **Dojo** (where a developer develops and after a certain time rotates with a colleague), and **individual** (usually used to solve simpler tasks). Some teams had more students with an aptitude for development while other students more beginners, thus centralizing the development of more experienced colleagues.

Before the competition, a directory was created on Github and each team should place their projects in this directory, and the organizers added the list of allowed emails according to the registration for the event. **Most teams did not perform any automated testing of their solutions**, some teams performed unit tests and not a far-reaching tool (such as Bugzilla) or technique such acceptance or performance test.

**Students Strategies to Manage their Tasks**

The teams managed the tasks in 3 ways, **most teams used the board** to demonstrate their user stories mapping or kanban board, while some were organized by **Trello**. **Few teams did not have organized task management**, e,g. two teams had a centered leadership of a student, while most teams had distributed control.

Most teams, in the end, managed to present their ideas, however, **the teams that managed to do better delivered an MVP** (Minimum viable product), while some teams presented mockups or software that would simulate the desired behavior of their solution

proposal. Few teams showed disagreement among their members during the event, just a certain discomfort as time went by and the delivery time became shorter. The two groups that had conflicts were among the worst positions during the award, as the groups ended up distancing themselves from any member who could not work well with the rest of the team.

The teams presented several **difficulties with some tasks**, such as **understanding the problem** proposed by the event, **organizing the requirements for an MVP**, **some teams were stuck in development**, one team did not perform version control correctly, losing their entire project, or centralized the tasks in just one member. The teams that consulted with the event's mentoring team and the stakeholders used to present a greater mastery over the proposed problem, and were better evaluated in the end, as their proposals were more feasible to help in the problem situation.

**Students' Intention to Release and Maintain the app**

Students are advised to submit their projects in directories on Github, provided by the event organizers. **All projects developed at this event deliver an MVP** at least, otherwise, they are disqualified, and these versions are completely free to the event's stakeholders after the end of the hackathon.

It was quite **unusual the cases that the students showed interest in continuing** the project after the hackathon, or to keep developing this proposal. From what was perceived, no student chose to continue developing the project, however, some students were willing to continue the project if the stakeholder wanted and hired them.

# 7 DISCUSSION

The discussion will be based on comparisons between the original studies and our study carried out in this paper. In the first paragraph of each subsection will be presented the findings of the original study and in the second paragraph we will present the new findings found in this study.

## 7.1 Students Motivation to Participate (RQ1)

**Factors that may Motivate the Students**

Warner and Guo [14] explain that there are a set of social factors that influence a student's participation in a hackathon, such as the ease of finding teammates, being colleagues and the technical opportunities to demonstrate their knowledge in practice.

In our study, we perceived that these events are also an opportunity to **learn from colleagues** during the development of the project, elaboration and establishment of **networkings with colleagues and stakeholders** that the student did not know previously, **improve their teamwork skills**, dealing better with the functions they performed and the **social impact of the applications** produced during the event.

**Students Feelings and Fears**

Warner and Guo [14] claim as the biggest factors for not participating in a hackathon being: Novice fears, No time, No team/idea, Discomfort, Hacker culture, competitive, no substance. Novice fear and No time was the most appointed factors in their study.

In our study, we found that participants who compete for the first time often say they were **afraid before competing**, or that they competed in hackathons of **other formats that were uncomfortable**. However, the students who participated evaluates the experience as positive, being an **opportunity to improve their skills**, and **to approach the job market**. In addition, some students stated that they wanted the **evaluation process to be clearer** and more experienced students feel a little **uncomfortable with students who are new to technology** in their teams.

## 7.2 Students Perceptions about the Hackathon (RQ2)

**Students Strategies to Build Teams**

Warner and Guo explained that the most common team building strategy adopted by students was to team up with classmates or try to team up at the venue. Usually, students who did not know the team previously felt a little more nervous about forming a team on the day of the event.

In our study, we identified that teams tend to come semi-structured, with members who know each other but with the possibility of joining new colleagues from other programs.

**Students Expectations to Learn**

Warner and Guo [14] describe several differences between the traditional classroom and the event, such as proximity to industry experiences (greater authenticity in simulating the job market), the opportunity to be hired or to meet important people to their professionals' choices. These authors describe that the format of a hackathon, despite being unstructured, must be accompanied by pedagogical methods and that the students' relationship regarding time is a great differential when compared to a discipline.

In our study, we identified, in addition to the described by Warner and Guo [14], that students **learn from their peers**, sharing knowledge among colleagues, and that an **experience different from that provided in the classroom** makes them develop their technical skills even more. Also, we observed social skills, such as **teamwork**, **time organization** and the advantages of working with **people from different backgrounds**.

**Students Perceptions about the Event**

Warner and Guo [14] present that one of the things that surprised students the most was learning in an informal teaching environment, without realizing that they were learning. In addition, the moment considered most memorable for students at the observed event was learning in pairs, with their colleagues, throughout the presentation[14].

In this study, we found that several things surprised students (e.g., the diversity of background of teammates). These students point out that the most memorable moment during the hackathon was the pitch, i.e., the moment when they presented their projects to the stakeholder.

## 7.3 Software Development Practices Adopted by Students

**Students Dealing with the Software Requirements**

In the 3 hackathons investigated by Gama [6], the author explained that most of the participants created the requirements inspired by their own experiences, where a considerable part of them researched on the Web more about the problem situations to be

solved. It means, most of the participants played the role of stakeholder, without consulting.

In this study, we perceived that the teams behaved differently according to their requirements, however, **the teams that were most successful at the event are the ones that talked intensively with the stakeholders** and organizers of the event, seeking to understand the details of the proposed problem. In addition, s**everal techniques and tools were used to organize the requirements**, from more traditional approaches, such as taking notes in notebooks or whiteboards, as well as digital approaches, such as the use of tools such as Trello.

**Students Concern about Software Quality**

Gama [6] explains that deliveries in a hackathon are usually a functional prototype, but in the 3 hackathons he observed, the teams discussed little the software architecture. Also, although Gama [6] has identified in the literature that Pair programming reduces the risk of defects in software development, he realized that few teams have adopted pair programming technique and the software tests have been done only in an exploratory way.

In this hackathon that we observed, **architecture was little discussed by students**, as they had similar knowledge about some technologies, as they learned from the same teachers, and **pair programming was much more used**, especially by teams that did better. In addition, **students did not perform any automated tests**, and students who did better did simple functional tests. However, we observed that some of the best placed teams had done some simple tests, such as unit test or exploratory tests.

**Students Strategies to Manage their Tasks**

Gama [6] explains that most of the tasks in the hackathons were managed as the project evolved. The position of project manager used to be decentralized, and this was indicated as the favorite form of management by students.

In this study, we perceived that each team **managed their tasks in different ways**, physical (such as using the whiteboard) or digital (such as Trello). Few teams had **internal conflicts**, usually when a student assumed the role of project manager and acted in a way that disliked the others when sharing tasks, and these teams were among the worst at the event.

**Students' Intention to Release and Maintain the app**

Gama [6] explains that most of the participants informed that they used version control, however, most of the projects at the end of the hackathons are not finished. Finally, many participants abandon the projects after the hackathon.

In this hackathon, **all teams used version control**, and almost all **projects featured an MVP**. Few students have shown **interest in continuing their projects**, but only if they are hired to be able to dedicate themselves entirely to the project.

## 8 LESSONS LEARNED

In this section, we will share some of the lessons learned during the conducting of this study.

**Impacts of Using Informal Learning Approaches**

Software Engineering professors often look for opportunities to improve the teaching of their courses and finding effective ways to improve the students' learning process can be a big challenge.

A hackathon can be used, in a complementary way, as a promising teaching tool in an environment that simulates configurations of an industrial real environment, with students solving real problems. Another impact is that the students learns without realizing that them are in a learning process, unlike a traditional method that, although useful, can generate a series of blocks in the students that can be undone in an informal teaching environment.

**Motivating Students in a Real Learning Environment**

Students are often afraid to participate before having their first experience, as they leave the comfort zone of a classroom to face a different activity. Hackathons usually have the format of the competition and offer very limited time for the proposal of an adequate solution to the theme problem, which can put pressure on students in challenges never taken before.

However, when students allow themselves to participate once, they recognize the positive impact that this competition can bring, such as the possibility of working with friends and improving technical skills in a more relaxed environment, but which still has a focus on teaching. Often, it is in an adverse situation that students realize how much they have managed to evolve in a certain area of knowledge or skill.

**Importance of Teamwork to Students**

Working in teams helps students to learn from their peers, and if the group has different backgrounds, students can complement each other. Hackathon, despite being an informal event for learning in this context, presents different mechanisms for the dissemination of knowledge, in addition to the possibility for students to test what they have learned in the formal teaching environment.

Teamwork is also a motivating factor for students to participate, as they have the possibility to work with friends and meet new people to create a network of professionals. In addition, the combination of students' skills and knowledge can build more robust or interesting applications to solve the problem proposed by Hackathon, such as, for example, a team that presented at the end of the event two solutions, one in mobile application and one on a website in the pitch.

**Advantages to Students that Participate**

The advantages that the student can have when participating in an event like this is recognized by the students themselves. Among these advantages, the student can have the experience of working on a real project, being able to work with friends, testing their skills and knowledge, testing new technologies not yet explored by team members, developing creative ideas that meet the interests of stakeholders, gain experience from an environment closer to the industry and, if your solution is successful, win a prize.

This event offers a safe learning environment, technological resources, and teaching materials so that teams can organize themselves, a support team that can answer questions about the competition, accompanying professors and industry professionals to support learning and the possibility of exchange information even with students from other teams.

**Hackathon and the Short Time to Activities**

A problem that bothers students in some situations is the short time that the hackathon has so that they can develop an adequate solution to the proposed problem. However, the proposal of a

hackathon is that students, in a short period of time, try different roles and activities to propose something at the end of the event to the invited stakeholders.

In this study, we realized that as time goes by, students abandon the known practices and trying to do whatever they can to get the solution done. For example, no team in this hackathon has used any type of automated testing or had a concern for the quality of the proposed software. This is because the important thing in this type of event is to prove the concept that the idea proposed by the team is valid, even if it is not perfect at first.

**Hackathons as a complementary Pedagogical Approach**

Individually, a Hackathon does not solve the learning or teaching difficulties present in the academy, but it helps both teachers and students to observe the problems from a different perspective, in which the student approaches reality to test their knowledge.

However, a hackathon allows students to evaluate themselves during the learning process, being the true owners of their learning process, realizing where their strengths and weaknesses are while developing a real project. In addition, teachers can understand their students' difficulties and propose activities aiming to support them in these difficulties.

## 9 LIMITATIONS

The study contains some limitations, such as the background of the students from the same university, although the teams are composed of students from different technology programs. This is because all students are from the same university, with little diversity between students' knowledge or techniques.

This study was inspired in two studies by other authors, however, it was necessary to adapt it, since not all were focused on teaching, but rather often to hackathons as an event to expand the networking of professionals. In these adaptations, some changes were made to other forms of collection or analysis to ensure that the reflections also apply to the participants' learning.

However, in almost all research questions, we used two research methods, these interviews, and observations, to safeguard a better understanding of the observed event and its participants. Another limitation is that the students may have not understood in-depth the domain context, but they were able to ask questions to organization staff or customers who brought the problem throughout the event having the chance to deepen the required knowledge to propose a solution.

## 10 CONCLUSION

In this study, we report how an educational hackathon was organized and its results, both from the student and project perspective. From the students' perspective, we seek to understand their motivations, their fears, and how they organize themselves for a competition like this. As a result, it was discovered that events like hackathons can be used for effective pedagogical purposes since students consider that they improve their skills in these competitions. From the project perspective, we identified the students' choices regarding the development cycle of their proposals and the problem situation of this hackathon.

The two previous studies (Warner and Guo [14], Gama [6]), supported us to observe the event from these two perspectives, as well as to understand the impact that this event can have on the students of this university.

This study can support teachers who wish to use different pedagogical resources for their teaching of Software Engineering, understanding how to promote healthy competition in order to further improve the skills of their students.

The future work will observing this year edition of the event, that will be completely virtual, due to COVID-19 [11]. In this future study, we aims to investigate how the pedagogical environments, teamwork skill development and students behavior may change due to the virtual interaction in a hackathon.

## ACKNOWLEDGEMENTS

We thank AGES for this partnership that allowed us to investigate this event.

## REFERENCES

[1] Laurence Bardin. 2011. *Content analysis*. Vol. 70. São Paulo: Edition, São Paulo, Brazil.
[2] Michelle Calco and Ann Veeck. 2015. The markathon: Adapting the hackathon model for an introductory marketing class project. *Marketing Education Review* 25, 1 (2015), 33–38.
[3] Adrienne Decker, Kurt Eiselt, and Kimberly Voll. 2015. Understanding and improving the culture of hackathons: Think global hack local. In *Proceedings of the IEEE Frontiers in Education Conference*. IEEE, El Paso, USA, 1–8.
[4] Allan Fowler. 2016. Informal stem learning in game jams, hackathons and game creation events. In *Proceedings of the International Conference on Game Jams, Hackathons, and Game Creation Events*. ACM, San Francisco, USA, 38–41.
[5] Frank J Frey and Michael Luks. 2016. The innovation-driven hackathon: one means for accelerating innovation. In *Proceedings of the European Conference on Pattern Languages of Programs*. ACM, Kaufbeuren, Germany, 1–11.
[6] Kiev Gama. 2017. Preliminary findings on software engineering practices in civic hackathons. In *Proceedings of the IEEE/ACM International Workshop on CrowdSourcing in Software Engineering*. IEEE, Buenos Aires, Argentina, 14–20.
[7] Marko Komssi, Danielle Pichlis, Mikko Raatikainen, Klas Kindström, and Janne Järvinen. 2014. What are hackathons for? *IEEE Software* 32, 5 (2014), 60–67.
[8] Jabu Mtsweni and Hanifa Abdullah. 2015. Stimulating and maintaining students' interest in Computer Science using the hackathon model. *The Independent Journal of Teaching and Learning* 10, 1 (2015), 85–97.
[9] Alexander Nolte, Linda Bailey Hayden, and James D Herbsleb. 2020. How to Support Newcomers in Scientific Hackathons-An Action Research Study on Expert Mentoring. *Human-Computer Interaction* 4, 1 (2020), 1–23.
[10] Jari Porras, Jayden Khakurel, Jouni Ikonen, Ari Happonen, Antti Knutas, Antti Herala, and Olaf Drögehorn. 2018. Hackathons in software engineering education: lessons learned from a decade of events. In *Proceedings of the International Workshop on Software Engineering Education for Millennials*. ACM, Gothenburg, Sweden, 40–47.
[11] Hussin A Rothan and Siddappa N Byrareddy. 2020. The epidemiology and pathogenesis of coronavirus disease (COVID-19) outbreak. *Journal of autoimmunity* 19, 1 (2020), 1–4.
[12] Per Runeson and Martin Höst. 2009. Guidelines for Conducting and Reporting Case Study Research in Software Engineering. *Empirical Software Engineering* 14, 2 (2009), 131.
[13] Erik H Trainer, Arun Kalyanasundaram, Chalalai Chaihirunkarn, and James D Herbsleb. 2016. How to hackathon: Socio-technical tradeoffs in brief, intensive collocation. In *Proceedings of the ACM conference on computer-supported cooperative work & social computing*. ACM, San Francisco, USA, 1118–1130.
[14] Jeremy Warner and Philip J Guo. 2017. Hack. edu: Examining how college hackathons are perceived by student attendees and non-attendees. In *Proceedings of the Conference on International Computing Education Research*. ACM, Tacoma, USA, 254–262.
[15] Marcelo Yamaguti, Flávio de Oliveira, Cássio Trindade, and Alessandra Dutra. 2017. AGES: An Interdisciplinary Space Based on Projects for Software Engineering Learning. In *Proceedings of the Brazilian Symposium on Software Engineering*. ACM, Fortaleza, Brazil, 368–373.