An Online Educational Hackathon to Foster Professional Skills and Intense Collaboration on Software Engineering Students

Caio Steglich, Sabrina Marczak, Luiz Guerra, Cássio Trindade, Alessandra Dutra, Ana Paula Bacelo
Pontifícia Universidade Católica do Rio Grande do Sul (PUCRS)
Porto Alegre, Brazil
(caio.borges,luiz.guerra,cassio.trindade)@edu.pucrs.br
(sabrina.marczak,alessandra.dutra,ana.bacelo)@pucrs.br

ABSTRACT

The global pandemic of COVID19 demanded that professors rethink teaching strategies considering the use of online environments due to the social isolation stipulated to reduce the rate of contagion of the disease. A challenge for software engineering professors is to develop fundamental professional skills in students who are in the process of learning using these virtual environments. The purpose of this study is to identify how an online educational hackathon can support students of a Software Engineering program to develop professional skills. We also seek to understand how intense collaboration takes place between student teams, considering the digital context for the production of a technological solution. We conducted a Case Study on an educational hackathon that took place in the online context, collecting data through questionnaires, interviews, and observations. As some results, the skills that students most considered that this hackathon helped them to develop were communication, initiative, and creativity/innovation, among others. Also, the strategies of collaboration adopted by the students during this competition, considering the remote context. Therefore, the main contribution is the identification of how the realization of this event supported students to develop professional skills and to practice collaboration skills with each other.

CCS CONCEPTS

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

KEYWORDS

Software Engineering Education, Educational Hackathon, Intense Collaboration, Case Study

ACM Reference Format:
Caio Steglich, Sabrina Marczak, Luiz Guerra, Cássio Trindade, Alessandra Dutra, Ana Paula Bacelo. 2021. An Online Educational Hackathon to Foster Professional Skills and Intense Collaboration on Software Engineering Students. In *Brazilian Symposium on Software Engineering (SBES '21), September 27-October 1, 2021, Joinville, Brazil.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3474624.3476973

SBES '21, September 27-October 1, 2021, Joinville, Brazil

ACM ISBN 978-1-4503-9061-3/21/09.
https://doi.org/10.1145/3474624.3476973

1 INTRODUCTION

The popularly called hackathons consist of intense competition over a limited period of time in which an individual or group must design the solution to a problem. A hackathon applied to a Software Engineering problem is usually a competition in which the participants must design a prototype of software that a solution to the proposed problem, respecting this time limit (Komssi et al. [11]).

A hackathon can be conducted in several areas, such as scientific, biological, business, technological, social, among others [17]. Usually, these hackathons can be organized by:

- **Government:** Some hackathons whose purpose is usually to think of solutions to social problems, also called Civic Hackathons [4].
- **Companies:** Some hackathons whose purpose is usually to prospect future employees for the company, who can stand out through promising proposals given a problem situation [21].
- **Academy:** Some hackathons whose purpose is usually to support their students to practice the knowledge gained throughout their degree programs [26].

Hackathons have been used as a pedagogical tool to deepen important concepts for the area of Software Engineering, encouraging the student to practice the concepts learned in the classroom. As an example, Sadovykh et al. [23] report the experience of adding educational hackathons as a curricular practice to foster students' contact with the technology industry. Another example would be the study by Steglich et al. [25] who studied how an educational hackathon can support students to adopt software engineering practices in problem-solving. However, the new context of the global pandemic of COVID19 formed an unforeseen scenario that has been less explored in terms of its impact on the teaching strategies previously used.

This study aims to present a case study on an online educational hackathon experience for students of Software Engineering. We investigated in this study what professional skills this kind of event develops in the students who participate and how this process of intense collaboration between students takes place considering an atypical scenario while solving a problematic situation.

Some of our findings include:

- The students consider that the most developed skills during this hackathon were: Communication Skills, Initiative/Motivation to Work, Creativity and Innovation, Interpersonal Relationships, Teamwork, Autonomy.

- The students considered that the skills they had least developed during this hackathon were: Self-Esteem, Stress Tolerance, Flexibility, Attention to Details / Organized.
- The synergy of the teams, the organization strategies during the event, and the establishment of simple protocols among teammates supported the collaboration, improving the teams' performance.

The remainder of paper is organized as follows: Section 2 describes the theoretical foundations of this work. Section 3 argues the studies that have similar contributions in this topic. Section 4 presents this hackathon setup. Section 5 describes the research methodology. Section 6 presents the findings obtained in this study and Section 7 discusses them. Section 8 discuss the lessons learned from this study. Section 9 lists the study limitations. Section 10 concludes the paper with considerations and future work.

2 BACKGROUND

This section presents the context of Software Engineering Education (SEedu), the COVID19 context, and their implications in education and hackathons for education.

2.1 Software Engineering Education

Academia and industry are often evolving. However, some challenges and gaps arise between both that is pointed out in some literature studies. Ouhbi and Pombo [19] presents a set of challenges in the teaching of software engineering faced by teachers, these being: i) Engaging students; ii) Designing practice activities; iii) Too much material to cover; iv) Finding adequate textbooks; v) Create assessments/exams that go beyond memorization; vi) Finding adequate technology, and tools; vii) There is a significant discrepancy between textbooks and practical life, and viii) Students tend to do well on the test, but in practice, they are lost.

Oguz and Oguz [18] studied the possible causes of the gaps between academia and industry, these being: i) The software industry expands to new areas; ii) The feedback from industry to the academy is missing; iii) Software Engineering Education is not agile enough to incorporate the new practices that emerge in the industry; iv) Courses are isolated, but related, which prevents students from seeing the connection between them; v) Academics are not out of touch with the industry; vi) Seniors software engineers may widen the gap; vii) Soft Skills are required for Software Engineering practices, and viii) Course projects do not have realism.

Considering these gaps between the teaching offered in the academy and the knowledge demanded by the industry, several different approaches could be used in teaching, such as, for example:

- **Gamification:** Souza et al. [24] described an experience in the usage of two-game elements, namely badges and leaderboards, in an introductory Software Engineering course.
- **Games:** Monsalve et al. [15] studied the usage of a game-based learning tool called SimulES-W to teach Software Engineering in an undergraduate engineering course.
- **Flipped Classroom:** Kiat and Kwong [10] studied the usage of flipped classroom techniques in the teaching of some software engineering topics.
- **Hackathons:** Porras et al. [20] presented a taxonomy for software engineering education through hackathons in supporting the participants to decide which kind of intensive event approach is most suitable for them.

Although some studies in the literature explore the use of educational hackathons as a pedagogical resource, the new context of the COVID19 pandemic raised several issues about how previously applied practices in a face-to-face context now occur in a remote context.

Ferreira et al. [6] explored the Brazilian context in teaching software engineering, in which he categorized the problems in software engineering Brazilian education as follows: Resources-related (Lack of bibliographic resources, Resources blocked on the internet, and Lack of tools), Students-related (Disinterest, Difficulties in programming, Lack of maturity, and Difficulties in understanding the requirements), and Content taught related (Very extensive content, and Very theoretical content).

2.2 SEedu During COVID19

The new global context of COVID19 meant that universities needed to readjust their teaching practices for virtual environments. However, it can cause some side effects, such as, for example, Motogna et al. [16] found that in most software engineering disciplines, students' understanding dropped considerably from face-to-face to remote teaching, for example, "Ability to understand and apply software design principles" was understood by approximately 79% of students in the face-to-face format, decreasing to approximately 26% in the remote format.

Barr et al. [3] present some experiences from different disciplines being taught in this new online modality, considering that the teaching practices that worked well were: i) Weekly online multiple-choice quizzes using the Moodle platform; ii) Tutorial sheets; iii) Lab exercises; and iv) Links to additional material. In addition, they also explain that the practices that did not work well were: i) The rapid pace of the module was generally found to be overwhelming; ii) Group activities did not seem to go very well, especially when the live classes moved online and virtual breakout rooms were used; iii) The dedicated time for labs with a tutor was not used very effectively, and iv) The instructor did not have good visibility of student engagement with the recorded lectures.

Kanij and Grundy [9] explain the biggest challenges of this new teaching paradigm, which are: i) Challenging Engagement; ii) Stressful Transition. iii) Asynchronous Learning; iv) Offline program demonstration; v) Open Access Tests; vi) Nature of assessments, and vii) Distribution of grades. As lessons learned, they point out that: i) Communication from the teacher is critical; ii) A regular pattern of communication can do the trick; iii) Long lecture videos are too bandwidth expensive; iv) Recorded videos were less motivating; v) Assessment of open access tests is challenging; vi) Designing problem-solving questionnaires are difficult; vii) Program demonstration should be done online, and viii) Students should be taught to set up their environment.

2.3 Educational Hackathons

Frey and Luks [7] explains that a hackathon is often structured into the following phases: preparation, development, and post-hackathon. Also, the preparation phase has some stages: i) the problem identification; ii) the analysis of possible solutions; iii) prototyping; and iv) presenting the pitch of the solution and receiving feedback. Hackathons have been used in several contexts, one of which is educational, and can be used to practice skills developed during undergraduate courses. For example, Gama et al. [8] propose a methodology based on challenge-based learning and design thinking to support teams that will perform hackathons, with some of the recommended techniques: i) Brainwriting, ii) Voting Heuristics, iii) Persona, and iv) Physical Computing Cards. They also explain the methodology as a model, dividing it into three phases: research, engagement, and action. The investigation must take place at all times. The engagement phase has three stages: big idea, essential question, and challenge. The action phase also has three stages: solution, implementation, and evaluation.

Čović and Manojlović [5] present that the main skills that an educational hackathon usually develops in its students are: i) collaboration; ii) teamwork; iii) negotiation; iv) project management; v) time management; vi) communication; and vii) troubleshooting. In addition, there are advantages for students by companies supporting the universities in this type of events, which are: i) Admitting students to practical lessons; ii) Cooperation with professors in providing conditions for the realization of practical teaching in the company following the curriculum; iii) Engagement of experts from the company for the realization of teaching; iv) Willingness of employees to transfer professional and practical knowledge to school professors; and v) Presentation of the company's activities to students for their employment, scholarships for successful students.

The virtual context that was imposed by the COVID19 pandemic presents new challenges that need to be explored. In this research, we seek to understand how these hackathons occur considering this new context.

3 RELATED WORK

Some studies present investigations during educational hackathons, such as, for example, Warner and Guo [27] who investigated the motivation of students to participate or not to participate in an educational hackathon. Among the results identified by them, social factors are fundamental for those who participate, learning among peers is extremely necessary for solving problems, and many of those who do not participate feel discomfort because they are novices in some technologies.

Steglich et al. [25] investigated students' adoption of Software Engineering practices during an educational hackathon. This investigation took place in stages of developing the solution for each team, in which the authors cataloged: i) how the students dealt with the requirements; ii) how they defined the solution architecture; iii) how they measure the quality of their proposals, iv) how they managed tasks among their team members; v) the feeling they had during these stages and their perceptions about the competition.

Sadovykh et al. [23] present a proposal to introduce hackathons as an element in a graduate program in software engineering. The results from the students' perspective would be: i) New Knowledge; ii) Practical experience; iii) New contacts with the industry; iv) Insights about the business domain; v) Insights about a future career; vi) Inputs for the research; vii) Potential project with a company; viii) Inputs for writing a paper; and ix) internship opportunity.

Unlike others studies, this work took place in an online context due to the COVID19 pandemic, not having a focus on investigating student motivation or adopted software engineering practices, but rather the professional skills that an educational hackathon can develop and how collaboration takes place between students.

4 HACKATHON

The event lasted 30 hours, with 3 presenting the problem situation, 24 hours of developing the solution proposal, and 3 of pitching the proposals made by the students. More specifically, 1h for team formation, 1/2h for the problem presentation by the stakeholder, 1h for the teams to define a software solution proposal, 1/2h for all teams pitching their proposal, 24h for coding, 2h for pitching the solution, and 1h for awarding the winners. The event received 46 registrations, but each team could invite an extra member from outside the competition to participate, and 6 from 8 teams invited an extra member, being 52 competitors.

The students, being in their homes, should use their own computers and the Discord tool was enabled, containing a general room for announcements from the organization of the event and each team had a specific group to be able to work. The teams should be composed of six students from some of the undergraduate computer programs, the majority belonging to the Software Engineering program, and being able to invite an external member who was not officially registered at the event to support the team.

The problematic situation that students should propose solutions to support during this hackathon would be: to support the challenges that society started to face because of the pandemic of COVID19. Students should, in groups, debate, plan, propose, and develop a solution that would help to solve some of the problems that emerged during this global crisis, and could be focused on helping people, companies, or the government.

The ideas chosen by the teams were (and their composition):

- **Team 1:** They proposed a web platform that offers support for people who are suffering from depression, anxiety, emotional stress, or just wanting to chat (6 students, 1 extra member).
- **Team 2:** The idea is an application "Help your neighbor" that aims to help a person in your region who is experiencing financial problems to sell a product or service, intermediating this person's contact with potential buyers (6 students, no extra member).
- **Team 3:** The proposal was a gamified virtual queue for establishments, thus avoiding crowding, helping small businesses, and allowing people to leave their homes without crowding (6 students, no extra member).
- **Team 4:** The team proposed an application that explains to the relatives of a person hospitalized by COVID19 their health status and the medical procedures performed (5 students, 1 extra member).
- **Team 5:** The idea is to create an application to finance companies that were affected during the pandemic, promoting

benefits for the user who contributed to the company by contracting services/purchasing from the same company (5 students, 1 extra member).
- **Team 6:** The idea was a chatbot to support the home office in telegram and discord chats (6 students, 1 extra member).
- **Team 7:** The team proposed a psychologist referral platform for people in social isolation who may be experiencing a problem (6 students, 1 extra member).
- **Team 8:** The proposal was an application to promote donations for needy people who are experiencing difficulties because of the COVID19 pandemic (6 students, 1 extra member).

The process evaluated the teams to compose the solution and the results of the solution itself. Some of the evaluation criteria were feasibility, creativity, reach a good prototype during the event, and meet the judges' expectations. These judges were: 2 professors of Software Engineering Undergraduate program, 1 professor of Business Undergraduate program, 2 directors of different companies, 1 student from Computer science graduate program.

The award was given to students from the three best-evaluated teams (it means the students in the teams that placed first, second and third place in the competition), and every student from these teams earned a brand-new headset. The best-placed teams in this competition were, respectively: Team 6 (1st place), Team 3 (2nd place), and Team 2 (3rd place).

In the week following the event, students who participated in the hackathon were invited to fill out the questionnaire for this study, and the teams were invited for interviews, which took place up to 3 weeks after the event.

5 RESEARCH METHOD

The approach chosen to carry out this study was the Case Study, according to Runeson and Höst [22]. In this study, the research questions formulated are:

(RQ1) What skills do students consider best to develop in an online educational hackathon?

(RQ2) How does intense collaboration between students take place during an online educational hackathon?

5.1 (RQ1) Students Skills - Questionnaire

We initially identified a skill set to assist in response to RQ1. This initial set was elaborated from three studies found in the literature, namely: Lacher et al. [12], Matturro et al. [14], and Albena et al. [1]]. The list of skills identified in the literature is shown in Table 1.

We developed an instrument considering the 21 skills identified in the literature presented in Table 1. In this instrument, for each one of these skills, we asked how much the students consider that this hackathon supported them to develop this skill. Each question has a 7- point Likert scale where students have seven options to score, with the option on the far left being "totally disagree", in the center the option "neutral," and on the far right the option "totally agree".

This questionnaire was validated by two Software Engineering researchers who have more than ten years of researching the topic of teaching Software Engineering. In addition, three students who did not participate in the event were asked to answer the questionnaire as a pilot, the first from the Computer Science undergraduate program, the second from the Information Systems undergraduate program, and the third from the graduate program in Computer Science.

Table 1: Skills from Literature

ID	Skill	References
F1	Communication Skills	[1, 12, 14]
F2	Learning	[1, 12, 14]
F3	Leadership	[12, 14]
F4	Teamwork	[12, 14]
F5	Critical Thinking	[1, 12]
F6	Problem Solving	[12, 14]
F7	Initiative / Motivation to Work	[12, 14]
F8	Autonomy	[1, 14]
F9	Ability to receive criticism	[12]
F10	Interpessoal Relationships	[12]
F11	Social Sensivity	[12]
F12	Attention to Details / Organized	[12]
F13	Stress Tolerance	[12]
F14	Time Management	[12]
F15	Customer Orientation	[14]
F16	Planning skills	[14]
F17	Collaboration	[1]
F18	Creativity and Innovation	[1]
F19	Self-Esteem	[12]
F20	Flexibility	[12]
F21	Results Orientation	[14]

The responses were collected using the Qualtrics[1] tool, receiving 34 responses. However, only 28 were valid, as the others were incomplete or the respondent left without concluding the answering the questionnaire. Table 2 presents the students profile that answered our questionnaire thoroughly, being each column: ID (an identification for each student who answered the questionnaire), Team (An ID for the team this student participated in during this hackathon), Gender (the respondent's gender), Semester (in which semester the student is currently in the program) and Undergraduate Program (contain acronyms for Software Engineering (SE), Information Systems (IS) and Computer Engineering (CE)).

The respondents are primarily students of the Software Engineering undergraduate program, being between the 1st to the 7th semester. In addition, this sample got a response from students who made up the eight teams that competed throughout this hackathon.

5.2 (RQ2) Students Collaboration - Interviews

Initially, we found a collaboration framework in development teams, presented by Maksimov and Flicker [13] that bring the following elements that must be present for a collaboration:

- **Agreement:** An agreement can be seen as a compromise, and it is the more general term that expresses the recorded

[1]https://www.qualtrics.com | Accessed on May, 19/2021 at 09:02 PM

Table 2: Profile of students who answered the questionnaire

ID	Team	Semester	Undergraduate Program
SQ1	T1	4th	SE
SQ2	T2	2nd	SE
SQ3	T2	2nd	SE
SQ4	T3	6th	SE
SQ5	T4	2nd	SE
SQ6	T4	6th	SE
SQ7	T4	4th	SE
SQ8	T5	2nd	SE
SQ9	T3	6th	SE
SQ10	T3	7th	SE
SQ11	T6	6th	IS
SQ12	T3	2nd	SE
SQ13	T6	6th	IS
SQ14	T7	7th	SE
SQ15	T3	7th	SE
SQ16	T8	5th	SE
SQ17	T1	5th	SE
SQ18	T5	4th	CE
SQ19	T2	2nd	SE
SQ20	T8	4th	SE
SQ21	T7	7th	SE
SQ22	T6	6th	IS
SQ23	T1	5th	SE
SQ24	T6	6th	IS
SQ25	T1	7th	SE
SQ26	T2	2nd	SE
SQ27	T4	1st	SE
SQ28	T7	7th	SE

rules for working together as well as for creating and exchanging resources and knowledge [13].

- **Actors and Parties:** Specialists and clients can be independent companies, organisations or individuals and are the actors or the parties in the collaboration [13].
- **Competences:** The abilities or the competences are the skills utilised by actors to achieve the goals of the collaboration [13].
- **Goals:** The objectives are the goals of the collaboration [13].
- **Resources:** The resources are material or immaterial goods such as hardware or software utilised to achieve these goals [13].
- **Collaboration Process:** The collaboration process is a sequence of activities where competences and resources are integrated to achieve the goals of the collaboration [13].

Through these elements presented by the Maksimov and Flicker framework [13], the following questions were elaborated, considering one or more of the elements presented:

(1) How did the idea for your project come about?
(2) How did you organize the activities between you? What criteria were used?
(3) What agreements or rules have you defined to support the team during the project's development?
(4) What purpose did your team have in working together on this Hackathon?
(5) Did you choose an external guest to help you? Who did your team select and why?
(6) How did this external guest support the team?
(7) Did you consult any stakeholder? How was that interaction? How much did this interaction help in understanding the problem of the project?
(8) What tools did you use to support the development of the project?
(9) Which functionalities of these tools were essential to support the team's collaborative work in the development of the project?
(10) How did you share knowledge among the team members during the Hackathon?
(11) Did the team need to ask for help from another team or someone outside the team besides the special guest? Who and for what reasons?
(12) In the specific case of facing a difficulty during the development of the project, how did the team proceed to resolve the issue?
(13) What were the biggest difficulties you felt in carrying out this project?
(14) Which skills were most useful throughout the project you participated?
(15) What skills did you perceive that were missing from the team during the development of the project?
(16) What were the activities carried out by the team and how did you work on it?

This interview protocol was validated by two Software Engineering researchers who have more than ten years of researching the topic of teaching Software Engineering. In addition, two students who did not participate in the event were asked to participate in the interviews as a pilot, being them both from the graduate program in Computer Science.

After this, interviewed 15 students under four interview sessions, grouped by the teams they participated. These students profile are presented in Table 3, being each column: Session (An identifier of which interview the student participated in), Team (An ID for the team this student participated in during this hackathon), Student (an identification for each student who answered the interviews), Gender (the respondent's gender), Semester (in which semester the student is currently in the program) and Undergraduate Program (contain acronyms for Software Engineering (SE), Information Systems (IS) and Computer Engineering (CE)). In addition, some of the interviewees did not participate by answering the questionnaire in RQ1.

The interviews lasted an average of 40 minutes, where we conducted in digital format through the Zoom tool[2], having been recorded with the participant's consent and later transcribed for data extraction. Data analysis was conducted based on the content analysis proposed by Bardin [2], to safeguard the quality of the data obtained and its correct understanding in each context.

[2]https://zoom.us | Accessed on May, 20/2021 at 01:38 PM

Table 3: Profile of students who participate the interviews

Session	Team	Student	Semester	Undergraduate Program
I1	T3	SI1	7th	SE
I1	T3	SI2	7th	SE
I1	T3	SI3	6th	SE
I1	T3	SI4	7th	SE
I2	T7	SI5	7th	SE
I2	T7	SI6	6th	SE
I2	T7	SI7	7th	SE
I3	T6	SI8	6th	IS
I3	T6	SI9	6th	IS
I3	T6	SI10	6th	IS
I3	T6	SI11	5th	SE
I3	T6	SI12	5th	SE
I4	T4	SI13	4th	SE
I4	T4	SI14	1st	SE
I4	T4	SI15	6th	SE

6 RESULTS

This section aims to represent the data collected to answer the research questions, having been obtained following the methodologies as explained previously.

6.1 (RQ1) Students Skills - Questionnaire

We collected data from 28 students from the 46 who participated in this hackathon. We asked these students how much they think they have developed each of the skills shown in Table 1. The results are shown in Figure 1, wherein the central area are the people who chose the neutral option in each factor, that is, they did not opine about it. In the area on the left, represented in shades of yellow and red, are the number of students who disagreed, that is, who believe they did not develop or practice this skill during the hackathon. On the right, in blue tones, the number of students who agreed, that is, who believe that the hackathon has supported them to some degree in developing these skills.

These skills were selected because they represent the desired skills of software engineering professionals in the 21st century, some of which the students consider to have managed to develop during this competition. The skills that students consider to have developed the most during this hackathon were: i) Communication Skills; ii) Initiative / Motivation to Work; iii) Creativity and Innovation; iv) Interpersonal Relationships; v) Teamwork; and vi) Autonomy. The skills that the students consider that they were less able to develop during this event were: i) Self-Esteem; ii) Attention to Details / Organized; iii) Ability to receive criticism; iv) Leadership; and v) Stress Tolerance.

Some skills, when assessed by students, were categorized as neutral. This can occur because answering the questionnaire can lead the student to a profound reflection on the skills they have practiced throughout the hackathon, and eventually, they may not have encountered situations in which the development of such skills was evident. The skills that had the highest neutral score were: i) Stress Tolerance; ii) Flexibility; iii) Planning skills; iv) Ability to receive criticism; and v) Attention to Details / Organized.

6.2 (RQ2) Students Collaboration - Interviews

A key element to such events as hackathons is the strategy students use to interact with each other to solve a problem. For the first time, this event took place online to protect people's health, considering the current context of COVID19 present in society. However, it is essential to understand how this interaction differs from the face-to-face and how students have been structured to solve a problem considering these challenges. Thus, it is essential to understand how the collaboration between students occurred, and they are presented as follows.

How did the idea for your project come about?

Usually, the teams started with a brainstorm strategy, in which all members propose solutions and together chose the best alternative: - "We brainstormed at the beginning of the hackathon on a whiteboard, and we identified how to help in the context of the pandemic, and I do not know who was the idea or inspiration for the solution we made." (SI2 - T1).

Some teams usually consult with stakeholders to compose the idea of their solutions: - "We talked to a psychologist, so the ideas came from the members, and we voted for the best one" (SI6 - T7).

How did you organize the activities between you? What criteria were used?

The teams present different strategies regarding the division of work, where one team explains that they have voluntarily organized tasks between their members: - "we organized the tasks, and each one volunteered to take what task want to develop" (SI5 - T7). Another team organized itself according to the technologies that each student would like to learn or practice throughout the hackathon - "We used Trello for the tasks and started together to make a low-fidelity prototype on Figma to get an idea of what would be developed. The logic for dividing the functions was organic. Each could do based on what they wanted to learn according to the screens that needed to be implemented" (SI1 - T3).

A team appointed two more experienced members as leaders to pull the development, considering frontend and backend: - "It was natural, we designated SI9 and SI10 to pull the frontend and backend parts. They were dividing the tasks little by little, and the members were taking turns doing some tasks, making small sprints, and introducing the team" (SI8 - T6). Finally, other teams were more objective, focusing on knowledge that people already knew in advance: - "We organized the tasks according to what the person knew how to do. At Hackathon, you do not have much time to learn. The ideal is to already know. It is more about putting into practice what you already know than learning something new" (SI15 - T4).

What agreements or rules have you defined to support the team during the project's development?

A recurring agreement between the team refers to seeking technologies in common among the students of the each team so that everyone can help each other: - "We agreed to look for common technologies among our colleagues. Another arrangement was to validate the ideas we had with a psychologist" (SI7 - T7). In addition, some groups agreed to do micro-validations with team members,

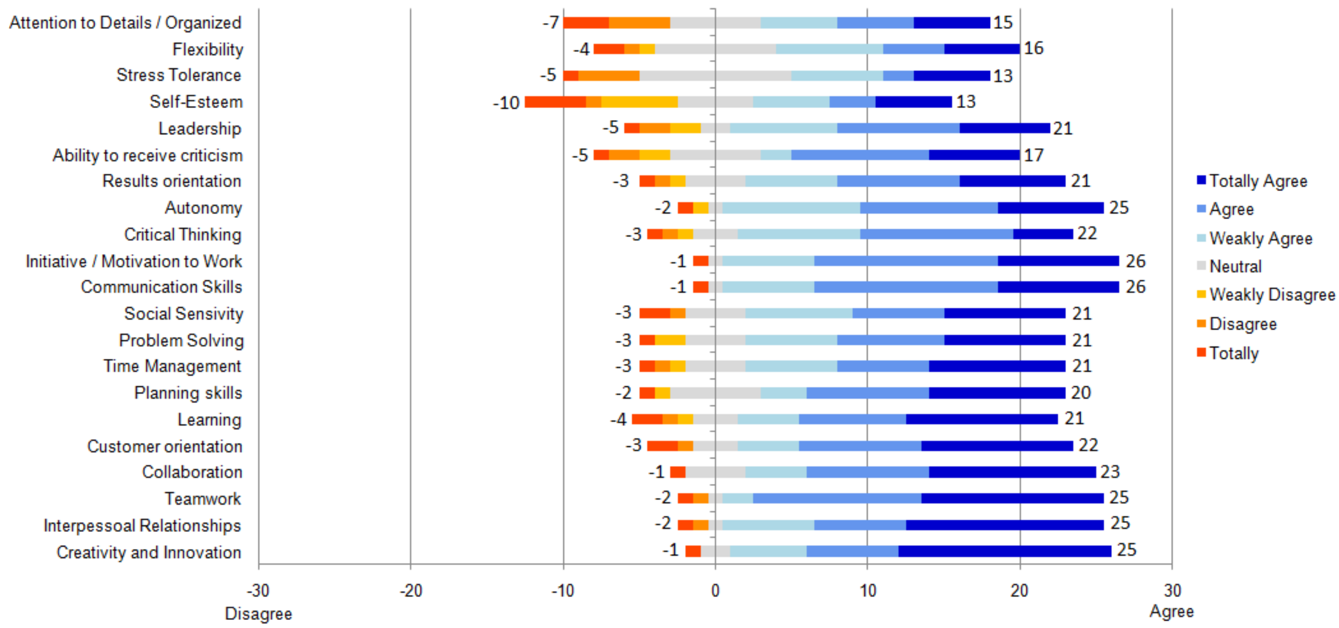

Figure 1: Students' Opinions About How Much they Developed from Each Skill

each time a task was done: - "We made the sprints, and at the end of each one we agreed to talk as a team about what we had done. We did it to exchange knowledge and gave good results. In addition, we did code review through pull requests for another colleague using clean code techniques" (SI8 - T6).

What purpose did your team have in working together on this Hackathon?

The objectives that lead students to compete can be diverse, but when it comes to forming their teams, each team had its own motivations. One of the teams worked together because they were colleagues and had the same vision on technologies: - "We all take the same subjects in the course and we all have a visual perspective of the projects, being guided by mockups" (SI4 - T3).

The bonds of friendship are equally important in the composition of a team: - " We being friends made it easier to carry out the project. From the beginning, we wanted to be first place this time. For us, being remote was much better than being in person" (SI8 - T6). In addition, the teams that compete have the intention of winning the competition by composing themselves with these colleagues: - "We wanted to develop something interesting and win the hackathon" (SI14 - T4).

Did you choose an external guest to help you? Who did your team select and why?

This event allowed the teams to invite an external guest member to assist throughout the competition. Some teams did not invite anyone external to support them: - "We did not pick anyone outside the team" (SI1 - T3). A team invited a fellow newcomer to learn about how these competitions work: -" We invited a newbie friend to learn about programming competitions, but he did not stay for long" (SI13 - T4).

The teams that they invited sought to be more strategic in what they missed most in their teams. One team invited a fullstack developer to help: - "We invited a fullstack developer, he is my friend and we have never worked together. He deals well with front and back, knows other technologies like php, and this helped us in the part of sharing knowledge, and helped a lot" (SI12 - T6). Finally, another team invited a tester to support the development of the project: - "We invited a tester who is the girlfriend of one of the team members. She helped us with a little bit of everything, especially in the design" (SI6 - T7).

How did this external guest support the team?

The teams that had external members received assistance on several points, such as design, programming: - " Our guest helped us with the design and the project schedule " (SI7 - T7). In addition, any technology that the team did not have much knowledge of: - "In technologies of little team domain, being very efficient in arriving and interacting with the project" (SI11 - T6).

Did you consult any stakeholder? How was that interaction? How much did this interaction help in understanding the problem of the project?

The teams that did not consult a stakeholder did so based on their own knowledge: - " We were more guided by our own experiences, much of it was by speculation. We did not actually talk to anyone if they would use our solution" (SI2 - T3). Some teams consulted health professionals, for example, a medical doctor to obtain some information about the virus: - "SI14 asked a doctor friend the data we needed to know about a patient in a hospital" (SI13 - T4).

Also, a team consults some psychologists to validate the functionalities to be developed: - "We consulted psychologists to validate the functionalities and they explained about the context that people live during the pandemic. One of the psychologists commented that she did volunteer work helping people who were experiencing problems during the pandemic" (SI5 - T7).

What tools did you use to support the development of the project?

The groups' communication took place mainly through Discord, being the official tool of the event, but some teams also used Whatsapp. The files shared by students regarding their solutions were arranged on Github or Google Drive. Tasks in most groups were organized on Trello's virtual boards or Jamboard. Mockups mainly were made with the Figma tool and, in some cases, graphically edited with Photoshop. The development of each group relied on different languages and technologies. However, the solutions used to be either web or mobile applications, making some choices more frequently, such as React or HTML with CSS. The most used programming interface was the VSCode.

Which functionalities of these tools were essential to support the team's collaborative work in the development of the project?

The most used technologies are reported by the teams are: **Discord:** ("Discord was fundamental for verbal communication and screen sharing" SI2 - T3), **Trello:** ("Trello helped us manage the deliverables, so we know which tasks have the highest priority" SI6 - T7), **Figma:** ("Figma allows us to share mockups in real-time, and everyone on the team can follow the screen and their mouse at all times" SI1 - T3), **Google Drive:** ("Google Drive allowed us to share resources, in particular the information or data needed to help produce functionality" SI15 - T4), **VSCode:** ("VSCode is an interface that allows us to develop in various languages and import different resources for use" SI6 - T7), **Github:** ("Github supported us manage proposal versions and share code among teammates" SI7 - T7), and **Jamboard:** ("Jamboard helped us manage the tasks, and everyone could watch and change at the same time" SI9 - T6).

How did you share knowledge among the team members during the Hackathon?

The groups organized different strategies to talk to each other, some of them using verbal approaches: - "When it was necessary, we went into a voice room, to talk about the points to do some functionality" (SI1 - T3). Other groups communicated by text messages in the tools used: - "We shared knowledge through discord, often sharing screens, but it was difficult to teach something in a hackathon, it was easier for the person who knew how to do it" (SI15 - T4)

Did the team need to ask for help from another team or someone outside the team besides the special guest? Who and for what reasons?

Most teams did not consult anyone other than the special guest. However, sometimes it is necessary to talk to the event organizing staff to clarify some rules or limits: - "We consulted the event staff on specific issues" (SI3 - T3).

In the specific case of facing a difficulty during the development of the project, how did the team proceed to resolve the issue?

The biggest difficulties faced by the teams tend to appear in the development stages: - "We had programming difficulties. The team needed to stop activities and focus on solving the specific one" (SI13 - T4). The most used strategies were: **Try to solve it alone first:** We remained calm and tried to get around first, thinking of alternatives and studying in detail things that we had difficulties (SI9 - T6), **Search online:** We search the internet, for example, on StackOverflow (SI7 - T7), and **Consult teammates:** After a while of suffering, we called some of our colleagues online. People with more free time helped those who were in more trouble (SI4 - T3).

What were the biggest difficulties you felt in carrying out this project?

The teams had two significant challenges that are common in this type of competition, the first being the knowledge limitations: - "Knowing how to program better, so as not to unravel the team, as this ended up centralizing the codes in just one person" (SI15 - T4). The other difficulty was the limitation of time to come up with a solution: - "We wanted to have used technologies that we couldn't because we did not have time, and so we had to change course to other technologies" (SI9 - T6).

Which skills were most useful throughout the project you participated?

The skills identified as most useful throughout the hackathon by the interviewees were: **Communication:** Understand what other colleagues are talking about and support their needs (SI14 - T4), **Technical knowledge:** Prior knowledge of the technologies that need to be used (SI6 - T7), **Proactive:** Do not waste too much time on just one task (SI1 - T3), **Organization:** We need to know how each member works and make it manageable (SI1 - T3), and **Resilience:** Do not give up in the face of difficulty (SI10 - T6).

What skills did you perceive that were missing from the team during the development of the project?

Among the skills that they felt were most lacking in the teams, two were scored as the greatest, with the first being little technical knowledge: - "We could have had more knowledge before the hackathon, but everything worked out" (SI10 - T6). The second skill that was missing was for the students to have a closer relationship with the business area and know how to explain their ideas better: - "It lacked someone with greater communication/business skills to present the project at the end (pitch)" (SI6 - T7).

What were the activities carried out by the team and how did you work on it?

The teams, despite having carried out these phases with different strategies, generally organized themselves into the following sequence of tasks: i) consider ideas; ii) decide design and scope; iii) mockups; iv) division of tasks; v) development; vi) pitch.

7 DISCUSSION

The technology professionals market is usually demanding regarding the skills demanded from its professionals. These demands create gaps between academia and industry, as it is quite complicated for academia to keep pace with industry demands. Therefore, one of the first steps to understand this gap is to explore what is demanded by this industry, identifying the necessary skills for technology professionals in today's society. For this, the set of 21 skills presented in Table 1, coming from the studies of Lacher et al. [12], Matturro et al. [14], and Albena et al. [1], was raised as a starting point.

Hackathons can be learning opportunities in which students can, in a limited amount of time, practice skills, exercise knowledge to solve a problem situation, but it is necessary to understand what these skills are to practice. For this, RQ1 seeks to make this self-assessment by students, understanding that a hackathon could not necessarily help to develop all these skills but makes it possible to identify that some are extremely common to develop this event proposal.

The development of a broad set of skills needed by the industry is a process that takes a few years in academia and can be supported by practices such as the use of hackathons or other educational events to strengthen some of these, such as those mentioned in the results. Therefore, this research provides evidence that some of these skills can be developed in hackathons, such as Communication Skills, Initiative / Motivation to Work, among others. Despite this, events like hackathons also have limitations. Some skills are little practiced during these competitions, such as, for example, i) Self-Esteem; ii) Attention to Details / Organized.

Collaboration among students to solve problems could be compromised when the forms of interaction are not in person, expanding the use of technologies for division of tasks that in face-to-face formats can occur in classrooms. Communication strategies were also different in the virtual environment, whether by voice or text, and the solution implementation tools also needed to have real-time sharing resources among participants, such as Discord, Figma, Github, or Google Drive.

Nevertheless, a hackathon is an event with a limited time, which means that students have to organize themselves in a simple way without spending too much time on any of the non-essential stages of the competition. Prior knowledge of both the technologies and the skills of other teammates generates an advantage that allows for an organic strategy for competition.

Finally, it is common to think that Software Engineering students would not have significant problems adapting to the digital context by studying technologies, but it is a careless conclusion if one considers that each person can have a different form of interaction. Some people are more embarrassed about expressing themselves through technologies, especially when members did not know their peers previously.

8 LESSONS LEARNED

In this section, we will share some reflections obtained through the execution of this study.

There is no Silver Bullets in SE education

There is no perfect or better teaching method for teaching Software Engineering since several factors can influence teaching, such as the profile of students, forms of interaction, content to be worked on, among others.

Hackathons have the potential to support students in developing some skills that are considered essential to 21st-century Software Engineering professionals, but certainly, this event is not entirely effective in developing all of these skills in all students.

Challenges in SE Education during COVID19

The practices used for teaching Software Engineering need to be rethought to be framed in an Online format. Hackathons are events that can take place both in-person and online. However, the COVID19 pandemic makes all the practices that were conducted face-to-face become digital.

This study points out that the main difficulties are related to the interaction of students, but once they establish collaboration strategies, the problem is usually considerably reduced. Therefore, teams that have bonds of friendship tend to be successful, as established protocols tend to be respected by team members.

Hackathons to Students Skills development

Hackathons are usually interesting strategies for teaching Software Engineering as they allow students to practice skills in a short period of time, having to use various knowledge developed throughout the undergraduate program.

However, hackathons often practice skills that students have previously developed, making it much more difficult for them to develop a new skill during competition. In addition, often in hackathons, the student can also carry out a self-assessment and understand which skills they were able to develop and which ones they still had room to improve.

Winners' Features and Profiles

Some characteristics were observed in the teams that performed better in this hackathon: students were generally friends before the event, communication protocols were clear, leaders were accessible to support problem-solving, technological choices were made based on prior knowledge of the team.

Usually, these teams consulted with stakeholders and sought to gather a significant amount of information about the problem before thinking of alternatives to solve it. During the development phase, these teams used to do small tasks to avoid any member getting stuck on a long task or having little knowledge of how to solve it.

When should I use an educational hackathon?

Educational hackathons are events that allow for healthy competition (i.e., without creating rivalries among students), in which students usually spend a few days. Teachers who want to use this type of event as an educational practice need to organize an online structure so that students can interact with each other, a support team that can be composed of teachers, and some kind of award that motivates students to participate in the activity.

The proposed problem can be framed in demand from the university or society, as in this hackathon, which was about making software proposals to support people considering the context of the COVID19 pandemic.

9 LIMITATIONS

This research is a Case Study to identify the skills that a Software Engineering Student could develop in an Online Hackathon and how these students practice communication in their team to solve a problem, considering the online context due to the COVID19 pandemic.

The students who answered the questionnaire referring to RQ1 in full were 28 of the 46 students enrolled, representing approximately 60% of the participants in the event. However, the closer to the total number of students, the more appropriate. Also, the interviews carried out in RQ2 only had the participation of 4 of the eight different teams. Therefore, we conducted four interviews with 15 students. Also, We chose to be as less intrusive as possible throughout data collection, considering how tired students would be from the competition and the Covid19 scenario as a new challenge. As such, We do not collected data on student performance after the hackathon.

In addition, the experience with this online hackathon was quite positive, and it is perhaps essential to analyze other events of the same type for a generalized view of the impact of hackathons for teaching Software Engineering.

10 CONCLUSION

Hackathons are interesting strategies to support the teaching of Software Engineering, which can even be adapted to adverse contexts such as social isolation due to the COVID19 pandemic, which alters the dynamics of how people in the world interact.

Meanwhile, hackathons are not perfect educational practices, having their limitations as well, since not all the skills needed in a 21st-century Software Engineering professional can be fully developed in this type of event. However, as this is an event held in a short period of time, it is interesting for students to understand the extent of their knowledge and their limitations, since the competition will require a set of professional skills.

Communication is the primary key in collaboration for teams gets to succeed in this type of competition in transcribing their ideas into viable solutions. Although the COVID19 pandemic has dramatically altered the ways people interact, students mainly created effective strategies to deal with the gaps it created.

Some future work can be conducted to seek the generalization of the data identified through this research, the skills developed by students, and the forms of communication and collaboration adopted by students through the observation of more hackathons on the teaching of Software Engineering.

ACKNOWLEDGMENTS

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

REFERENCES

[1] Antonova Albena, Stefanova Eliza, Nikolova Nikolina, Mihnev Pencho, and Bontchev Boyan. 2020. 21st Century Skills of ICT Professionals: the Requirements of Business and Readiness of Higher Education in Bulgaria. In *Proceedings of the International Conference on Computer Systems and Technologies*. ACM, Ruse, Bulgaria, 270–277.

[2] Laurence Bardin. 2011. *Content analysis*. Vol. 70. São Paulo: Edition, São Paulo, Brazil.

[3] Matthew Barr, Syed Waqar Nabir, and Derek Somerville. 2020. Online Delivery of Intensive Software Engineering Education During the COVID-19 Pandemic. In *Proceedings of the IEEE Conference on Software Engineering Education and Training*. IEEE, Munich, Germany, 1–6.

[4] Sara Jensen Carr and Allison Lassiter. 2017. Big data, small apps: premises and products of the civic hackathon. In *Seeing Cities Through Big Data*. Springer, Ljubljana, Slovenia, 543–559.

[5] Zlatko Čović and Helena Manojlović. 2019. Developing Key Competencies through Hackathon Based Learning. In *Proceedings of the IEEE International Symposium on Intelligent Systems and Informatics*. IEEE, Subotica, Serbia, 167–172.

[6] Thaís Ferreira, Davi Viana, Juliana Fernandes, and Rodrigo Santos. 2018. Identifying emerging topics and difficulties in software engineering education in Brazil. In *Proceedings of the Brazilian Symposium on Software Engineering*. ACM, São Carlos, Brazil, 230–239.

[7] 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.

[8] Kiev Gama, Breno Alencar, Filipe Calegario, André Neves, and Pedro Alessio. 2018. A hackathon methodology for undergraduate course projects. In *Proceedings of the IEEE Frontiers in Education Conference*. IEEE, San Jose, USA, 1–9.

[9] Tanjila Kanij and John Grundy. 2020. Adapting Teaching of a Software Engineering Service Course Due to COVID-19. In *Proceedings of the IEEE Conference on Software Engineering Education and Training*. IEEE, Munich, Germany, 1–6.

[10] Pang Nai Kiat and Yap Tat Kwong. 2014. The flipped classroom experience. In *Proceedings of the IEEE Conference on Software Engineering Education and Training*. IEEE, Klagenfurt, Austria, 39–43.

[11] Marko Komssi, Danielle Pichlis, Mikko Raatikainen, Klas Kindström, and Janne Järvinen. 2014. What are hackathons for? *IEEE Software* 32, 5 (2014), 60–67.

[12] Lisa Lacher, Gursimran Walia, Fabian Fagerholm, Max Pagels, Kendall Nygard, and Jürgen Münch. 2015. A behavior marker tool for measurement of the non-technical skills of software professionals: An empirical investigation. In *Proceedings of the International Conference on Software Engineering & Knowledge Engineering*. OPUS, Pittsburgh, USA, 409–414.

[13] Yuliyan V Maksimov and Samuel A Fricker. 2019. Framework for Analysis of Multi-Party Collaboration. In *Proceedings of the IEEE International Requirements Engineering Conference Workshops*. IEEE, Jeju Island, South Korea, 44–53.

[14] Gerardo Matturro, Florencia Raschetti, and Carina Fontán. 2015. Soft skills in software development teams: A survey of the points of view of team leaders and team members. In *Proceedings of the IEEE/ACM International Workshop on Cooperative and Human Aspects of Software Engineering*. IEEE, Florence, Italy, 101–104.

[15] Elizabeth Suescún Monsalve, Mauricio Toro, Raúl Mazo, David Velasquez, Paola Vallejo, Juan F Cardona, Rafael Rincón, Vera Maria Werneck, and Julio Cesar Sampaio do Prado Leite. 2018. SimulES-W: A collaborative game to improve software engineering teaching. *Computación y Sistemas* 22, 3 (2018), 953–983.

[16] Simona Motogna, Andrian Marcus, and Arthur-Jozsef Molnar. 2020. Adapting to online teaching in software engineering courses. In *Proceedings of the ACM SIGSOFT International Workshop on Education through Advanced Software Engineering and Artificial Intelligence*. ACM, Sacramento, USA, 1–6.

[17] 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.

[18] Damla Oguz and Kaya Oguz. 2019. Perspectives on the Gap Between the Software Industry and the Software Engineering Education. *IEEE Access* 7 (2019), 117527–117543.

[19] Sofia Ouhbi and Nuno Pombo. 2020. Software Engineering Education: Challenges and Perspectives. In *Proceedings of the IEEE Global Engineering Education Conference*. IEEE, ACM, Porto, Portugal, 202–209.

[20] 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.

[21] Bard Rosell, Shiven Kumar, and John Shepherd. 2014. Unleashing innovation through internal hackathons. In *Proceedings of the IEEE Innovations in Technology Conference*. IEEE, Providence, USA, 1–8.

[22] Per Runeson and Martin Höst. 2009. Guidelines for Conducting and Reporting Case Study Research in Software Engineering. *Empirical Software Engineering* 14, 2 (2009), 131.

[23] Andrey Sadovykh, Maria Beketova, and Mansur Khazeev. 2019. Hackathons as a Part of Software Engineering Education: CASE in Tools Example. In *Proceedings of the International Workshop on Frontiers in Software Engineering Education*. Springer, Villebrumier, France, 232–245.

[24] Mauricio Ronny de Almeida Souza, Kattiana Fernandes Constantino, Lucas Furtini Veado, and Eduardo Magno Lages Figueiredo. 2017. Gamification in software engineering education: An empirical study. In *Proceedings of the IEEE Conference on Software Engineering Education and Training*. IEEE, Savannah, Georgia, 276–284.

[25] Caio Steglich, Larissa Salerno, Thaís Fernandes, Sabrina Marczak, Alessandra Dutra, Ana Paula Bacelo, and Cássio Trindade. 2020. Hackathons as a Pedagogical Strategy to Engage Students to Learn and to Adopt Software Engineering Practices. In *Proceedings of the Brazilian Symposium on Software Engineering*. ACM, Natal, Brazil, 670–679.

[26] Walter F Uys. 2019. Hackathons as a formal teaching approach in information systems capstone courses. In *Proceedings of the Annual Conference of the Southern African Computer Lecturers' Association*. Springer, Northern Drakensberg, South Africa, 79–95.

[27] 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.