Hackathon as a Project Based Teaching Tool

Employing Programming Challenge in the Class

Bolatzhan Kumalakov
Dept. of Artificial Intelligence and Big Data
al-Farabi Kazakh National University
Almaty, Kazakhstan
b.kumalakov@gmail.com

Alla Kim
Dept. of General and Applied Psychology
al-Farabi Kazakh National University
Almaty, Kazakhstan
allakim2013@gmail.com

Salima Mukhtarova, Assel Shukurova
School of engineering management
Almaty Management University
Almaty, Kazakhstan
mukhtarova.s@gmail.com

Natalya Khon
Department of Psychology
Turan University
Almaty, Kazakhstan
natalya.khon@gmail.com

Abstract*—Project Based Learning (PBL) develops student knowledge and skills through solving authentic, engaging and complex challenge in a fixed time span. Academic literature reports its application over a wide variety of disciplines, having both immediate and long-lasting positive effect on team working, project management, and creativity when applying technical skills. We aim to enhance student learning in software engineering. 80% of student respondents up to the date feel that programming assignments that last longer than a week overlap with other subjects enough to draw dedication away. While it unquestionably develops transferable skills, such as time management, it also might not necessarily be good through introductory courses, where students face completely new material. As a result, our aim is to employ "hackathon" - a gathering, where programmers collaboratively code in an extreme manner over 12 to 24 hours – as a teaching tool. While there is a precedent of hackathon being used at University level, it was employed to substitute classic approach via excluding lecturing and lab work completely. Current paper presents research design and relevant discussion on measuring and validating learning outcomes when blending hackathon into traditional programming class as one of the components, not a substitute to other types of classes.

***Keywords*—Project Based Learning (PBL), Hackathon.**

I. Introduction

Project-based learning (PBL) is a very prevalent learning method where the classes are constructed around one or more projects [8, 20]. In the field of computer science PBL develops abilities to apply technical knowledge, to acquire programming skills, to get involved into team processes, and to master project management [2, 3, 19].

However, above-mentioned research indicates that typical projects take from a few days to a few weeks to complete. Our evaluation of student experiences at Almaty Management University revealed that 80% of them feel that programming assignments overlap with other subjects enough to draw the dedication away. Considering that, our argument is "introductory programming class students, who are not familiar with concepts and cannot relate to previous experience would benefit from shorter period projects." Of course, having to deal with multiple non-trivial issues at the same time is the essential condition for developing time management, self-discipline, negotiation and other transferable skills. However, the keyword here is introductory course, which implies that student is not able to produce reasonable estimation of efforts, required to complete the project.

As an alternative, "hackathon" is the gathering where programmers collaboratively code in an extreme manner over 24 to 48 hours [14]. It creates friendly environment for students with different abilities, interests and comprehension of the material, as group achieves the result, not individuals. Hackathon starts with participants grouping into teams either before or during the contest. Every team starts by generating the idea, which results in the concept design. Next step is the development of a working prototype. The bottom line is everyone gets to utilize different skill set and managing such an event promises considerable learning merits.

In [4, 5, 10] authors come to the conclusion that Hackathon students study just enough to "hack together" (design) a possible solution. The intense nature of hackathon increases the significance of using teaching strategies and definite kinds of knowledge to be able to promote to the event. Participants often have to learn a specialized programming language and/or new hardware in a short time or integrate previously learned concepts into a system (macrostructure).

Furthermore, as project develops skills necessary to complete the challenge change. Participants have to be flexible and responsive to what they are learning. With limited time, hackathon constraints push participants to find effective, personalized learning methods. There is no doubt; such conditions are favourable in many ways.

However, the question is "how one can employ hackathon such that it increases student comprehension and motivation, but also ensures that the full range of standard curricula topics is covered in expected depth?"

In [12] authors report the case study where 9 universities joined together to create hackathon projects in accordance with their curricular. In 2015 at the end of this research, a survey among students showed that the most of them liked this type of teaching and Hackathon improved understanding of the material.

On the other hand, [12] experiment involved students of different universities gather in one place and stay there for 2 days, and register their teams. Each team get a mentor – typically University professor - and works without any lectures or seminars.

Remainder of the paper is structured as follows. Section 2 defines the problem and formulates research hypothesis. Section 3 presents research design and elaborates on its environment, while Section 4 presents relevant discussion and results. Finally, Section 5 concludes the paper and outlines future research direction.

II. PROBLEM DEFINITION

In [18] have been mentioned that nowadays the number of students who choose to study Science, Engineering, Technology, and Mathematics is very low across the globe. Also have been studied the main reasons for the low interest in Computer Science (CS), some of them including teaching and learning strategies, how students understand computer classes, social relevance of CS, and student.

In [15] authors describe that programming consists of combination of activities such as designing, debugging, planning and testing. First of all students should learn how to develop programs via theoretical materials about programming syntax. The difficulty in understanding the theoretical program often lead disappointment. The formula of successful learning process, according to Garner, is learning division between theory and practical classes. Finally, these factors facilitate to the highest rate of dropouts.

All of these factors have promoted us to find new ways to support students in programming classes in future. The most important advantage of our model is that students are exciting because of the limits of time like deadlines and team works. From psychological point of view Hackathon could be considered in terms of promoting comprehension as macrostructuring of information during team-work [23].

At the preliminary stage team members insert their individual contributions, or personal meanings describing their understanding of the problem situation and how to deal with it, in the second stage collaborative macrostructuring resulting into mutual understanding, or joint comprehension needs to be established. Construction of joint meaning cannot be done through simple accumulation of the contributions of individuals. When one member of a team inserts meaning by describing the problem situation and how to deal with it, the other team members are actively listening and trying to grasp the given explanation [17].

These processes of macrostructuring of meaning can develop into cooperative construction (co-construction), a joint process of building meaning, by improving, building on, or modifying the original offer in some way [14]. The result of this process is that new understanding in the collaborative work that was not previously available to the group emerge. The differences in interpretation are normal in a team, they can promote mutual cognition, but it is important to keep all team members psychological safety, which should be of special concern and object of observation by the organizers of hackathon.

Research hypothesis is therefore formulated, as “Applying Hackathon to the curricular will improve comprehension of new topic and academic records (GPAs) of students in programming classes.”

III. RESEARCH DESIGN

A. Research Methodology

Research method is based on combination of quantitative and qualitative measures [9]. First quantitative measure is ‘Academic Motivation Scale’ Questionnaire, which gives us opportunity to evaluate intrinsic and extrinsic motivation and monitor the level of student’ interest for the discipline. The validity and reliability of this scale is discussed in [22]. Second, GPAs is used as a quantitative measure for academic productivity (student' performance). We measure them in groups of participants before and after the Hackathon is held.

Based on [15], knowledge transfer heavy relies on the capability of pedagogical process, including teachers and students and their quality like flexibility and learning capability influences the transfer process, so talking with students directly is necessary to understand the process of knowledge transfer, implementation and learning. There are 2 types of qualitative methods that will be used in this study: Preliminary and Post- hackathon Interview, Structured observation. Preliminary interview will define how participants embrace themselves and their studying habits prior to the hackathon.

Post-Hackathon Interview observes participants’ actions and define how they study and share data to ensure maintenance of their claims.

Student' comprehension impacts learning, and measuring level and peculiarities of comprehension provides an important contribution to our research. For this reason we have conducted structured observation. This method includes monitoring and recording student’ behaviour on a regular basis, peculiarities of student’ comprehension and in-group processes during hackathon as an informal, game-format and practical tool of teaching. Prior to the observation, an observation schedule has been produced which details what exactly we have looked for and how those observations have been recorded and interpreted.

Observation scheme

1. Students' attention - at the beginning of the Hackathon; - in the middle of the Hackathon; - at the end of the Hackathon;
2. Students' activity during -process of creating commands -process of product development -summing up the results
3. Interest in the topic
4. Attitude to tutor
5. Discipline *-at the beginning of the Hackathon;* *-in the middle of the Hackathon;* *-at the end of the Hackathon;*
6. Psychological safety -constructive conflict resolution -active listening
7. Comprehension -individual meanings -joint meaning
8. Assessment of student contentment

Observation scheme was measured by these five marks: 5-stable, 4- constant, 3 - persistent, 2 - diffused, 1 - absent.

Conducting this research, qualitative pre- and post interviews have been conducted and analyzed using thematic analysis for 26 participants with individual and team experiences. Analysis gives better comprehension of self-regulated learning through context from an internal and external perspective. All materials collected after our study - like interviews and observations, will provide visual evidence.

Pre-hackathon interview questions:

Question 1: Have you ever heard about Hackathon/HackDay? Yes/No

Question 2: If Yes - Where and how did you know about the Hackathon?

Question 3: Have you ever participated in the Hackathon? Yes/No

Question 4: If Yes – Did you like it and why?

Post-Hackathon Interview questions:

Question 1: How much did you like hackathon? Mark on the scale from 1 to 10

Question 2: In your opinion what was the main result of hackathon?

Question 3: Share your experience of what was the most interesting for you?

Question 4: What was the most difficult for you?

Is there something that you would like to share with us but we didn't ask you about?

B. Research Environment

Initially as we know there are lots of versions and variants of implementing Hackathon as a teaching tool. But as we have found in [7], one thing is universal and very important such as collaboration. Therefore it is not so easy to cover all rules of previous Hackathons, but we have observed from [7]:

– Events in Hackathon should be inclusive.
– The most important things are learning and sharing.
– Failure is acceptable and valuable too.
– Comfortable seating, Wi-Fi and electricity are required.
– Whiteboards and windows with an inspiring view are preferable.
– Participant should create their own teams by yourself.
– Leaders shouldn't be mandated.

IV. DISCUSSION

Overall, we believe that we can achieve our goal of making the course practice-centric while still ensuring that students get realistic practice in programming.

We consider the time pressure, competitive environment, and intrinsic motivation existing in hackathon as factors which can improve the level of study and comprehension of computer courses [1, 11]. At the present time traditional education implies the presence of a teacher. But in Hackathon students are equal and can cope without a leader and tutor, in other hand teacher always should be responsible for the successful conclusion of a learning courses. However, we do not seek to replace the teacher role. We are going to make learning process as a "critical making" process - deriving from the conjunction of critical thinking and reflective making. In this case, the teacher is the process designer for the critical making. And also the teacher does not dictate the technology but is responsible for creating the conditions for learning through making. We use the aforementioned [5, 12] experience as a starting point in order to define how maker scenarios can be integrated in formal learning activities in more detail. Or rather before the end of each midterms (with a traditional method) among students, within a group, Hackathon have been held. The final product of Hackathon covers all topics learned during this course session. Duration: less 12 hours, venue: Almaty Management University, tutor: course teacher.

V. RESULTS

Analysis of this study focus on results of both quantitative and qualitative methodology (Preliminary and Post-hackathon Interview, Structured observation, Academic records, Motivation Scale).

Some significant mean differences between the before and after measurements in experimental and control group have been probative of the accuracy of studies' hypothesis.

To ensure that the experiment is pure and for understanding the impact of the hackathon in teaching programming, an experimental and control group have been involved. In addition to traditional training, the experimental group have participated in the Hackathon, which have been conducted during the second midterm. Hackathon have been conducted according to the above rules. The control group have been trained in the traditional mode. The number of participants in both groups was equal - 13 people.

During the experiment, results were obtained on interviewing, observation, and differences in academic performance.

Structured observation's results that we have analysed by the SPSS23 tool was revealed a correlation relationship: 1. students' attention with comprehension. The result of maximum student involvement in the educational process is always a positive impact on comprehension; 2 - attention to the attitude to tutor. The attitude of the student to the teacher becomes more trustful because of increased students' attention; 3. comprehension with students' activity (non-parametric coefficient) - negative correlation. The last non-parametric coefficient indicates that students with a good theoretical base do not show excessive activity (Table 1.1).

The following indicators were used to evaluate the (from 1- 5 scale):

Students' Attention: frequency of asking recurring questions;

Students Activity: frequency of asking questions and participating in team discussions; frequency of work in the group;

Interest in the topic: discussion of additional information on the topic;

Attitude to tutor: correct and courteous attitude to the teacher, not allowing personal insults and raising the tone;

TABLE. 1.1 CORRELATIONS

Pearson Correlation (Two-sided)				
	Student attention	***Student activity***	***Interest in the topic***	***Attitude to tutor***
	1	-0.372	0.114	.600*
Student attention		0.21	0.71	0.03
	13	13	13	1303
	.638*	-0.548	0.12	0.149
Comprehension	0.019	0.052	0.695	0.628
	13	13	13	13
Nonparametric Correlation				
	0.337	-.630*	0.225	0.031
Comprehension	0.26	0.021	0.46	0.921
	13	13	13	13

Discipline: frequency of failure to comply with Hackathon conditions;

Psychological safety: Adequate attitude to criticism;

Comprehension: frequency of discussions with the mentor of issues that arose during the course of the Hackathon, which shows the students' knowledge level;

Assessment of student contentment: according to the post Hackathon interviews' results [16].

The most important quantitative measurement in our article is the comparison of academic performance in the experimental and control groups, taking into account the hackathon of the first group, thereby determining how the use of hackathon as a teaching method affects academic achievement.

As a result, we received a slight difference, namely, an increase in the performance of the experimental group by 2% relative to the control group, which indicates a positive effect of the proposed method. The reason of the small growth may be a lack of necessary students' background.

Pre- and post-hackathon interviews gave us results reflecting the satisfaction of the students who participated in the hackathon - 77% (average of students' assessment grades from Fig. 1.2), which indicates an increase in motivation and possibly understanding. Also, following the results of the post of the Hackathon interview, there was a high interest in further participation in such projects.

Also in the experimental group we identified 2 clusters: first - a cluster of 5 participants whose knowledge and academic performance were the highest (100%) before the Hackathon already and stayed the same after it took place, so no change in terms of academic records could be registered even if it probably may be very high and the second cluster is the others. First cluster's data made it difficult to compare the effect of the new method. According to [22], we can state that their academic performance indicators have not changed, but at the same time, using the intercorrelation scales of academic motivation, we can judge a significant connection between these students in Motives of achievement and Cognitive motives. These students took first place in the Hackathon.

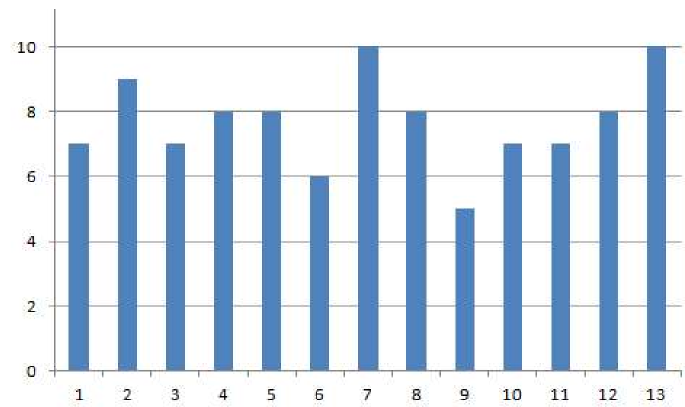

Fig. 1.2 Assessment of student contentment

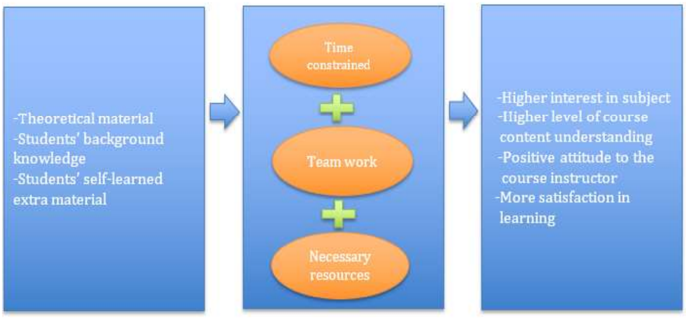

Fig. 1.3 Hackathon description

VI. Conclusion and Future work

[6, 8, 14] indicate that the majority of Hackathon participant students found its environment useful for learning and - above all - changing their comprehension of computing.

Data obtained throughout presented research allows us to conclude that:

- there is a group of top-students, who can be productive under any learning conditions, both traditional and Hackaton-based teaching,
- for the remaining 60% students Hackaton-based teaching demonstrates maintaining student' interest and understanding in computer science, enabling computing students to cooperate with subject - matter experts and most importantly industry professionals to make better understanding of the relevance of CS in a real-world setting, which is very promising in a longer perspective.

Main contribution of our work is, therefore, a proposition of a Hackathon blended model as an appropriate method for increasing students' interest and understanding in computer science and encouraging their engagement in using computing skills and knowledge in future work.

Our preliminary findings have produced not significant, but positive results acceptable to apply hackathon to the curricular of Almaty Management University. Further research work points to the replication of our preliminary results within broader quantity of students and universities.

VII. References

[1] A. Fowler, F. Khosmood, A. Arya, G. Lai, 'The Global Game Jam for Teaching and Learning', Conference: Proceedings of the 4th Annual Conference of Computing and Information Technology Research and Education New Zealand (CITRENZ2013)

[2] A. Intykbekov, 'Teacher perceptions of project-based learning in a Kazakh-Turkish Lyceum in the northern part of Kazakhstan', 2017, pp.11-16

[3] A. Jaime, J. M. Blanco, 'Spiral and Project-Based Learning with Peer Assessment in a Computer Science Project Management Course', 2016, pp.2

[4] C. Kelleher, R. Pausch, 'Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers'. ACM Comput. Surv. 37, 2 , 2005, pp. 83-137

[5] C. La Place, S. S. Jordan, M. Lande, S. Weiner, 'Engineering Students Rapidly Learning at Hackathon Events', pp.1-13

[6] E. H. Trainer, A. Kalyanasundaram, C. Chaihirunkarn, J. D. Herbsleb, 'How to Hackathon: Socio-technical Tradeoffs in Brief, Intensive Collocation', CSCW'16, San Francisco, CA, USA, 2016, pp.1118-1120

[7] G. Briscoe, C. Mulligan, 'Digital Innovation: The Hackathon Phenomenon', Creativeworks London Working Paper No.6 (2014), pp.1-15

[8] H. Kienzler, C. Fontanesi, 'Learning through inquiry: a Global Health Hackathon', Informa UK Limited, trading as Taylor & Francis Group, 2016, pp. 130-138

[9] Jamilya B.Akhmetova, Alla M.Kim, Delwyn L.Harnisch. Using mixed methods to study Emotional Intelligence and Teaching Competencies in higher education.// Procedia - Social and Behavioral Sciences 128 (2014) 516 – 521

[10] J. Duhring 'Project-Based Learning Kickstart Tips: Hackathon Pedagogies As Educational Technology', NCIIA, 2014

[11] J. Musil, A. Musil, D. Winkler, S. Biffl, 'Synthesized Essence: What Game Jams Teach About Prototyping of New Software Products', Conference: 32nd ACM/IEEE International Conference on Software Engineering (ICSE 2010), pp. 184-186

[12] J. P. Suárez, C. Domínguez, R. Cabello, S. Sánchez, 'Hackatones y docencia: educación disruptiva inclusiva basada en el aprendizaje basado en proyectos', 2015, pp.1-6

[13] M. Baker, (1994). A model for negotiation in teaching-learning dialogues. // Journal of Artificial Intelligence in Education, 5(2), 199-254

[14] M. Komssi, D. Pichlis, M. Raatikainen, K. Kindström, J. Järvinen (2014). What are hackathons for? Aalto University, Finland, pp.60-67

[15] M. Lara, K. Lockwood, 'Hackathons as Community-Based Learning: a Case Study', TechTrends, 2016, pp.486-495

[16] M.Pravdina "Styles of teaching and involvement of HSE students in the learning process: evaluation of positive effects // Monitoring of the University. 2010. № 3" (In Russian)

[17] P Van den Bossche, Wim H. Gijselaers, Mien Segers, Paul A. Kirschner (2006). Social and Cognitive Factors Driving Teamwork in Collaborative Learning Environments. Team Learning Beliefs and Behaviors//Small Group Research, vol.37, Number 5, 490-521.

[18] R., Firth, (2014) 'South Africa needs to get kids interested in computer science, or risk falling badly behind memeburn'https://memeburn.com/2014/09/south-africa-needs-get-kids-interested-in-computer-science-or-risk-falling-badly-behind/ (Accessed 11 February 2015).

[19] R. Pucher, M. Lehner, 'Project Based Learning in Computer Science ± A Review of More than 500 Projects', Procedia - Social and Behavioral Sciences 29, 2011, pp.1561-1566

[20] S. Cocco, 'Student leadership development: The contribution of project-based learning '(Unpublished Master's thesis). Royal Roads University, Victoria, BC, Canada, 2006

[21] S. Mohd Salleh, Z. Shukur, H. M. Judi, 'Analysis of Research in Programming Teaching Tools: An Initial Review', Malaysia: Universiti Kebangsaan Malaysia., 2013, pp. 127-135

[22] T. O. Gordeeva, O. A. Sychev, E. N. Osin. 'Academic Motivation Scale' Questionnaire. //Journal of Psychology, 2014, vol.35,#4,98-109

[23] W.Kintsch (1998) Comprehension: A Paradigm for Cognition. Cambridge University Press,