

Student Feedback for Globally Distributed Team Application Development

David Drohan
Dept. of Computing & Mathematics
Waterford Institute of Technology
Waterford, Ireland
ddrohan@wit.ie

Patrick Seeling
Dept. of Computer Science
Central Michigan University
Mount Pleasant, MI, USA
pseeling@ieee.org

Abstract—This Research Work-in-Progress Paper presents an evaluation of feedback gathered from a post-semester survey jointly taken by students in Ireland and the United States. The ability to function in international team-based environments becomes increasingly important and represents a desirable skill for employers of graduating students. Despite incentives, an international course offering in mobile application development resulted in only two cross-institutional teams by student choice, which we investigate here.

We encountered that about half of the respondents considered dropping the course for at least one reason at one point, with one third identifying the course difficulty level as a reason, but also indicating agreement with the course being worthwhile. Some of the feedback presented indicates that students have a preference for project-based courses and like working in teams, but are also wary of the potential coordination overheads and dependability on their peers, despite modern communication tools as part of the class offering.

Index Terms—Global teams, Virtual groups, Mobile application development

I. INTRODUCTION

In recent years, the development of applications has become increasingly distributed in nature - be it for large corporations with global footprints or smaller local companies that partially outsource their development efforts to other teams. Providing students with this essential skill, however, is a desirable outcome of future educational offerings, especially for domains that lean themselves towards these agile and flexible teams, such as the software engineering domain we consider in this contribution. Following the characterizations in [1], we provided an international multi-university course offering that empowered students to self-select based on their individual preferences. While team-based, active learning projects can readily be integrated into the coursework for an individual institution, crossing institutional boundaries commonly encounters additional problems, such as synchronization of instructional content and timing.

Some of the challenges we assumed have been documented extensively in the past, e.g., in [2], where the authors perform an extensive review of the literature and identify best practices to address potential hurdles. Our course offering comes after an initial experimental connect where we asked students to critique each other's developed applications several semesters before this offering. Similar to the reported difficulties and

need for iterative refinements in [3], we performed a revision of the course to be more integrated and flipped as well, to enable significant dedicated time for help from instructors throughout the semester. While little buy-in was attained in the first offering, we developed a coherent full semester offering with additional support, of communications tools and the TutorStack framework [4], [5].

Our course is a common approach to integrate mobile devices and designing applications for them into the curriculum, dating back to the beginning of broad and cost-sensitive mobile device availability. In [6], [7], the authors describe industry-driven forays of employing J2ME-based programming while in [8], [9], Android-based capstone projects are reported. As the course iteration we consider here was based on our own prior experiences, the overall development was upgraded to switch from Java to Kotlin, overall similar to [10]. In the remainder of this contribution, we describe the course offering in Section II and evaluate the feedback gathered in Section III before we conclude with a discussion in Section IV.

II. COURSE DESCRIPTION

The course under consideration covers mobile application development through Kotlin, with 16 students in their fourth year of study at the Waterford Institute of Technology, Ireland, and 24 students at Central Michigan University, United States, during Fall 2019. The majority of students were working towards their Bachelor of Science degree in Creative Computing (Ireland) and in Computer Science (United States). Both classes were aligned with respect to instructional content and delivered at their respective institution by local instructors. The major difference for delivery was an offset start date. As part of their instructional sequence, students were initially introduced to Kotlin and the Android Framework as well as relevant development environments.

The overall course delivery format was based on a prototypical version of the TutorStack environment, which incorporates a four layer system embracing open and free technologies and is illustrated in Fig. 1. As the partnering institutions each employ a different Learning Management System (LMS), the students were managed independently. The layer for instructional materials, Tutors, was employed for the joint delivery of content, and identical in experience for students from both

institutions. Communications and community building was performed utilizing a shared Slack environment. Additionally, a media layer incorporating YouTube and other recordings was employed to deliver narrated instructional content.

As part of the course activities, the students in the United States were asked to develop mobile application ideas and present those using videos, which were shared with both classes and students were asked to join in teams. Subsequently, students were guided through the steps of a mobile application development with git as a versioning system and to provide additional on-ramping for a team project component. Students were strongly encouraged to join in cross-institutional teams and given extra credit as an incentive to compensate for the increased scheduling and communication overheads. Despite these motivations, only two international teams were formed and developed a mobile application each, four team projects and eight individual projects were delivered by students in the United States, while 14 individual student projects were completed in Ireland.

We employed a survey modeled after the Persistence in Engineering (PIE) survey instrument [11] and administered voluntarily online after the semester to gather information about the hindrances for the team project. We have used iterations of the survey successfully in other works, such as [12], which guided our selection of questions.

III. RESULTS

The main joined course offering goal for international team building was not fulfilled, as only two teams with two students each were formed - despite the significant incentives. In the following, we evaluate the overall results in the contexts of student perseverance, reported reasons to drop the course, and open feedback concerning team work. Due to limited space, we present the corresponding survey items together with their results.

A. Survey Responses

As part of the overall activities, students filled out a survey component that asked them to self-evaluate their perceptions of math and science as well as their self-perceived performance of these two. We received 24 responses from 40 students enrolled in the course, which overall cautions in generalizing results. We initially present the aggregated student responses in Table I for the learners' answers to the survey questions.

Due to space constraints, we focus primarily on feedback items that apply directly to the main theme of this contribution. We initially observe that the overall feedback indicates that students lean towards fulfilling a degree in CS/IT, albeit with a fairly high variation. Students also provided a lower and heterogeneous rating for their teamwork affinity, but have a slightly better view on their team performance. Additionally, students had diverse views on the availability of instructors, with 5 students disagreeing with adequate faculty availability. Some of these concerns could be based on office hour scheduling or similar issues; however, faculty were available in Slack during the semester for feedback. Equally leaning

TABLE I
POST-CLASS SURVEY MODELED AFTER THE PIE (WORDING ABBREVIATED TO FIT HERE) AND VOLUNTARILY RETURNED BY STUDENTS IN BOTH COURSE OFFERINGS (LIKERT-TYPE SCALE FROM 1- STRONGLY AGREE TO 5-STRONGLY DISAGREE).

Item	N	Mean	Std.Dev.
I want to continue courses in CS/IT or am graduating	24	1.79	1.351
I have good math skills	24	1.75	0.608
I have good science knowledge	24	2.54	0.833
I am good at applying math or science to real-world problems	24	2.25	0.794
I like working in teams	24	2.54	1.141
I perform well on teams	24	2.04	0.624
Creative thinking is one of my strengths	24	1.79	0.721
I am skilled at solving multi-solution problems	24	1.75	0.737
Math skills are important in CS/IT	23	1.91	0.793
Science knowledge is important in CS/IT	23	2.43	0.788
Ability to apply math/science is important in CS/IT	23	1.87	0.626
Enough opportunities exist to interact w/faculty	24	2.54	0.977
The computer facilities are good	23	2.26	1.054
The classroom facilities are good	24	2.5	1.18
I feel stressed about the course load	24	1.88	1.191
This is a worthwhile course	24	1.96	0.751
This is an enjoyable course	24	2.46	1.062
I learned a lot in course	24	2.04	0.806
I liked the flipped course approach	23	1.74	0.81
Learning materials requires lots of effort	24	1.62	0.77
Learning additional online materials requires lots of effort	24	1.75	0.676
There was enough material to get me started	24	2.58	1.06
I would prefer to start mobile development only	22	3.05	1.214
Having lectures focusing on Kotlin introductions is worthwhile	24	2.83	1.341
I would prefer to have more graded items	23	2.52	1.31
I prefer project-based courses like this over exam-based ones	23	2.17	1.267

towards neutral (but with a wider spread) are the ratings for computing resources and facilities. It is noteworthy that neither of these supporting factors received average negative ratings and some of the spread can be explained by the offering in two locations. Ratings for the course induced stress level indicate that about half of the respondents felt stressed to any degree with the course load; this is reflected in our discussion of course continuation in Section III-B. However, the overall course perception feedback shows that while students had diverse views on enjoyability (with only 3 students rating disagreeing and below), they agreed overall that the course was worthwhile and that they had learned a lot.

An evaluation of our course delivery approach in Table I provides further hints at the source of these ratings. While students liked the overall flipped course delivery approach described in Section II, they indicated that it took significant effort to learn the material. This, however, was indicated to not be due to a lack of materials to progress into the application development. Most notably here, the approach to initially present Kotlin and only afterwards introduce the mobile development with Kotlin seems to have been a major

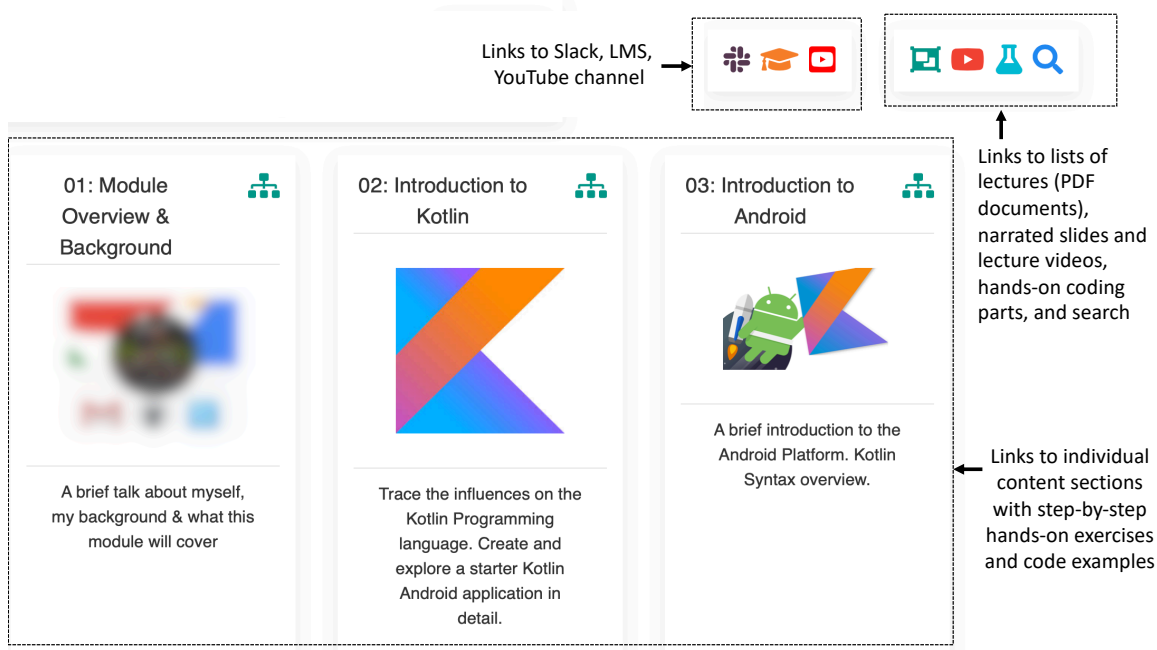


Fig. 1. Major component browser view (cropped for visibility) of the shared TutorStack learning environment: students had direct access to course materials, course communication, and management tools in a straight-forward fashion, with additional inclusions of code examples and narrated lecture slides.

TABLE II
11 (OUT OF 24) STUDENTS' REASONS FOR DROPPING THE COURSE.

Item	N	Mean	Std.Dev.
Course is too difficult	8	0.33	0.482
Course is too time consuming	6	0.25	0.442
Course is not interesting enough	0	0	0
Course conflicts with other classes	3	0.13	0.338
It is too difficult to catch up after missing class	0	0	0
Tuition concerns	1	0.04	0.204
Course conflicts with work/job	2	0.08	0.282
Course conflict with family obligations	2	0.08	0.282
Other, non-listed reasons	1	0.04	0.204

issue with about half of the students who provided feedback.

Finally, we note that while students overall indicate that they like the project-based course format, they are also desiring more graded items, which is somewhat of an opposite view on the course, but could be alleviated in future offerings through more graded steps towards the final mobile application presentation.

B. Reasons to Drop the Course

Next, we evaluate the students' self-reported reasons towards dropping the course. Specifically, we note that 11 out of the 24 students that responded to the survey indicated that they considered dropping the course at one point during the semester for at least one particular reason. We present the results obtained for different reasons we provided in the survey in Table II. We initially observe that no student indicated that the course was not interesting enough or that catching up after missing a course meeting caused them concerns to the point of considering a course drop. The majority of drop concerns

seem to stem from the course's difficulty level, corroborated by the required time for the course as second-highest reason. Conflicting classes were mentioned only by a small number of respondents. An even smaller number of respondents indicated that work or family caused them to consider abandoning the course. Lastly, only one student indicated tuition concerns. Interestingly, we noticed from our evaluation that about half of those students that indicated a consideration of dropping the course during the semester had more than one reason.

While we collected the information in a binary format, we also asked for specifics that students wanted to share in a free-form open-ended format. Specifically, students indicated that the actual programming of mobile applications with an object-oriented programming language such as Kotlin was stressful (though Java was part of the prerequisites) and that the prerequisites are not enough to ensure enough necessary prior student knowledge. Students furthermore indicated that the timing for assignments created additional concerns for them, with the on-ramping of the course resulting in a heavier workload in the course's middle and tail end. Lastly, one open response received was a student indicating health reasons.

C. Student Team Considerations

We now evaluate the student feedback provided with respect to their concerns for building teams. We provided two separate sections in the survey, which target the localized team formation as well as the international team formation separately. The open-format feedback results were coded and are presented in Table III, sorted by identified local team formation hindrances. Locally, students identified dependability on others as well as a preference to work alone and at their own pace as

TABLE III
REASONS PROVIDED AGAINST TEAM FORMATION.

Response (coded)	Local	Int.
Dependability on others	6	4
Work better alone	6	0
Prefer own pace	5	2
Different knowledge/skill levels	3	1
Liked own project	3	0
Time difference	2	6
Other/technical	2	2
Communication overhead	1	3
Grades	0	2
Language barrier	0	1

major hurdles to team formation. When regarding forming a team with their remote counterparts, they identified the time difference as most important hindrance. This was followed by not wanting to depend on others as well as a presumed communications overhead for international teams. Two respondents were also concerned about their grade, despite a generous extra credit that was communicated multiple times during the team formation time period. Interestingly, one student explicitly mentioned that joining an international team would be hindered by a language barrier, which could be explained by an international student being enrolled into the course offerings at the two participating English language institutions providing this feedback.

D. Impacts of Course Drop Likelihood and Team Affinity

In the following, we employ the course drop concerns as a grouping approach to evaluate the learner responses with respect to the other survey items to identify potential interplays. Facilitating this evaluation, we initially group students based on whether the learners indicated that they considered a course drop as part of their survey response. We employ these groups in an independent sample t-test analysis of the remaining survey items, but note that the following statistics have to be seen cautiously, especially with respect to the small sample size. The Levene statistic test for uneven variances indicates uneven variances for self-reported math skills, reported stress about course load, and whether the course is perceived as enjoyable. The first considerable difference is indicated for students indicating that there were enough opportunities to interact with faculty, $t(22) = -2.303$, $p = 0.031$. Specifically, the average for the group that indicated no drop was close to Agree ($M=2.15$, $SD=0.899$), while those that considered a course drop were closer to a Neutral rating ($M=3.00$, $SD=0.894$). Assuming non-equal variances for the course as by Levene's test outcome, $F(1,22) = 44.442$, $p < 0.001$, the reported high stress level for the course was significant as $t(15.108)=0.16$, $p=0.016$. For this item, students that considered dropping the course agreed more ($M=1.27$, $SD=0.467$) than their counterparts ($M=2.38$, $SD=1.387$). Similarly, we notice a significant difference for student responses regarding the course as worthwhile, $t(22)= -2.762$, $p=0.011$. For this survey item, students without consideration of dropping the course reported

a stronger agreement ($M=1.62$, $SD=0.65$) than the ones that considered dropping ($M=2.36$, $SD=0.674$).

As this course offering was geared towards cross-institutional teams jointly building mobile applications, we additionally employed the self-reported teamwork affinity of students in our evaluation (similar to the drop likelihood) as follows. We initially combined the two questions on the survey (*I like working in teams* and *I perform well on teams*) into a joined averaged variable, which we subsequently split at the *Agreed* level. Differently put, those students that have an average of more than 2.0 on the employed Likert-type scale are considered members of the less team affine group ($N=13$), with the remaining students being considered more drawn towards teamwork ($N=11$). An independent samples t-test based on these groups indicates that all survey items are fairly homogeneous between groups and no statistically significant difference was found. When evaluating the learner responses for dropping the course at one point, however, we observe a statistically significant difference $t(22)=2.683$, $p=0.014$. Students with a lower teamwork affinity indicated a higher level of potentially dropping the course during the semester ($M=1.73$, $SD=0.467$) than their more teamwork-oriented counterparts ($M=1.23$, $SD=0.439$).

IV. DISCUSSION AND CONCLUSION

In this course offering, students were empowered to self-select whether they would like to perform international mobile application development, with instructors on both sides heavily marketing this option as well as employing several common tools to support this team work. Nevertheless, students self-selected local teams or even individual application development when given the choice. Foregoing a significant grade bonus to not team up internationally and foregoing an opportunity to boost the resume has been very puzzling. However, a variety of reasons we identified are similar to prior works, such as in [13], [14]. Interestingly, we also find that less teamwork-affine students are more likely to want to drop the course, though our sample size limits the potential for generalization.

Additional survey modifications in future offerings should enable us to gather further insights into the student perceptions ex-ante as well as ex-post. Similarly, an iteration of this course could require international teams to gather further insights. Some of the non-asked questions that remain despite our exit survey could be whether a two-semester course offering would enable students to initially perform small collaborative work before moving onto bigger challenges, removing some of the initial unknowns and potential fears for teaming up internationally. This approach might also alleviate some of the identified course content anxiety, allowing a slower on-ramping. Due to collegiate scheduling and fit into coursework, such longer-term student commitment will require institutional buy-in, which might not be possible in times of budgetary challenges.

REFERENCES

- [1] S. Beecham, T. Clear, D. Damian, J. Barr, J. Noll, and W. Scacchi, "How best to teach global software engineering? educators are divided." *IEEE Software*, vol. 34, no. 01, pp. 16–19, jan 2017.
- [2] S. Beecham, T. Clear, J. Barr, M. Daniels, M. Oudshoorn, and J. Noll, "Preparing tomorrow's software engineers for work in a global environment." *IEEE Software*, vol. 34, no. 1, pp. 9–12, 2017.
- [3] A. Cajander, M. Daniels, C. Kultur, L. R. Dag, and C. Laxer, "Managing international student collaborations: An experience report," in *2012 Frontiers in Education Conference Proceedings*, Oct 2012, pp. 1–6.
- [4] E. de Leastar. (2019, Sep.) Case study: Tutors. Last accessed 2020-07-10. [Online]. Available: <https://aurelia.io/blog/2019/09/03/case-study-tutors/>
- [5] ——. Tutors. Last accessed 2020-07-10. [Online]. Available: <https://github.com/edeleastar/>
- [6] D. A. Umphress, J. H. Cross, J. Jain, N. Meda, and L. A. Barowski, "Bringing j2me industry practice into the classroom," *SIGCSE Bull.*, vol. 36, no. 1, p. 301–305, Mar. 2004. [Online]. Available: <https://doi.org/10.1145/1028174.971407>
- [7] Q. H. Mahmoud, "Integrating mobile devices into the computer science curriculum," in *2008 38th Annual Frontiers in Education Conference*, Oct 2008, pp. S3E–17–S3E–22.
- [8] S. Heckman, T. B. Horton, and M. Sherriff, "Teaching second-level java and software engineering with android," in *Proceedings of the 2011 24th IEEE-CS Conference on Software Engineering Education and Training*, 2011, pp. 540–542.
- [9] D. Rover, C. Ullerich, R. Scheel, J. Wegter, and C. Whipple, "Advantages of agile methodologies for software and product development in a capstone design project," in *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*, Oct 2014, pp. 1–9.
- [10] A. Esakia, "Transitioning to teaching android with kotlin and jetpack components," in *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, ser. SIGCSE '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 1302. [Online]. Available: <https://doi.org/10.1145/3328778.3372603>
- [11] Ö. Eris, H. Chen, T. Bailey, K. Engerman, H. G. Loshbaugh, A. Griffin, G. Lichtenstein, and A. Cole, "Development of the persistence in engineering (pie) survey instrument," in *Proc. American Society for Engineering Education Annual Conference & Exposition*, Portland, Oregon, USA, Jun. 2005, pp. 10.476.1–10.476.17.
- [12] "Introducing underrepresented high school students to software engineering: Using the micro:bit microcontroller to program connected autonomous cars," *Computer Applications in Engineering Education*, vol. 28, no. 3, pp. 737–747, Apr. 2020.
- [13] L. Wilson, S. Ho, and R. H. Brookes, "Student perceptions of teamwork within assessment tasks in undergraduate science degrees," *Assessment & Evaluation in Higher Education*, vol. 43, no. 5, pp. 786–799, 2018.
- [14] K. Falkner, N. J. Falkner, and R. Vivian, "Collaborative learning and anxiety: A phenomenographic study of collaborative learning activities," in *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*, ser. SIGCSE '13. New York, NY, USA: Association for Computing Machinery, 2013, p. 227–232. [Online]. Available: <https://doi.org/10.1145/2445196.2445268>