2023

A Tutoring Framework to Support Computer Science Programmes in Higher Education

Emer Thornbury
National College of Ireland, Ireland

Frances Sheridan
National College of Ireland, Ireland

Pramod Pathak
Technological University Dublin, Ireland, pramod.pathak@tudublin.ie

See next page for additional authors

Funder: This research received no external funding

A Tutoring Framework to Support Computer Science Programmes in Higher Education

Emer Thornbury
School of Computing
National College of Ireland
Dublin, Ireland
Emer.Thornbury@ncirl.ie

Frances Sheridan
School of Computing
National College of Ireland
Dublin, Ireland
Frances.Sheridan@ncirl.ie

Pramod Pathak
Faculty of Computing, Digital & Data
Technological University Dublin
Dublin, Ireland
Pramod.Pathak@tudublin.ie

Cristina Hava Muntean
School of Computing
National College of Ireland
Dublin, Ireland
Cristina.Muntean@ncirl.ie

Paul Stynes
School of Computing
National College of Ireland
Dublin, Ireland
Paul.Stynes@ncirl.ie

2023 IEEE World Engineering Education Conference (EDUNINE) | 979-8-3503-2050-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/EDUNINE57531.2023.10102887

Abstract— **Computing Support is the provision of academic supports such as individual tutoring and support classes to students studying computing at third level. Students can struggle with computing as it requires practice involving trial and error. This work proposes a research informed tutoring framework to support computer science students at third level. The tutoring framework combines three pillars; staff and training, pedagogies and activities. Support is put in place to help students develop technical and programming skills. Essential tutoring is provided for those who might otherwise drop out of college. The framework was applied to first and second-year undergraduate programmes and to the first semester in Higher Diploma conversion programmes. Results demonstrate that students' attendance has increased by 14%. In addition, student feedback shows that the framework supports students in computer science. This framework is a useful resource for academic managers that wish to implement computing support at third level.**

Keywords—computer science education, student success, retention, intervention

I. Introduction

Students who are new to computer programming, often find it challenging and an increased number of students are dropping out of computer science courses, leaving early from or not completing their course of study. There are recommendations at government level in Ireland and other EU countries to provide academic support to facilitate student success. Student success can be measured through the retention rate associated with a particular course and shows the persistence of a student in their studies. Therefore, student retention is becoming an increasingly important part of the college experience. Having high student retention rates is evidence of excellent educational delivery in a course. The aim of the research work presented in this paper is to investigate to what extent a tutoring framework supports computer science students with computer programming related modules. Institutions that are looking to strengthen their computing support, or create one, can plan the pedagogy and delivery in beginning. Pedagogy needs to be research based and oriented to the needs of students and a complete framework is needed for successful delivery of the service. The major contribution of this research is a novel tutoring framework for providing academic support to computer science students in third level education. The tutoring framework for computing support combines staff and training, pedagogy, and activities to support students in their technical and practical programming modules at third level.

The paper is structured as follows. Section II of this paper looks at current research that underpins the computing support tutoring framework. Section III outlines in detail the proposed tutoring framework for computing support, the 3 pillars that support the framework including the support activities all underpinned by the principles of Universal Design for Learning (UDL) and how the framework is implemented across the semester. Section IV discusses the research findings while last section concludes the research work.

II. Related Work

Effective pedagogy has an understanding of student learning at its core. However one needs to be aware of the following observation from Dewey who says not all educational experiences are "genuinely or equally educative", that some can be "miseducative" [1, p. 8]. There are many theories on teaching and learning and those that continue to hold weight are the theories of experiential learning or learning by doing, social learning and peer to peer learning. Teaching to encourage deeper rather than surface learning is also important as in computer programming subjects, students are scaffolded each week through topics that build on each other.

A. Experiential Learning

Research has shown that learning by doing or by practice, facilitates a cycle of intrinsic feedback from the task in hand and reflection on learning [2, p. 165]. Biggs says "what the student does is actually more important in determining what is learned than what the teacher does" [3, p. 97]. Students need to be engaged through dialog, given relevant material and be afforded opportunities to learn experientially [4]. Problem based learning is a method of providing practical, active learning through collaboration when working through problems [5]. This method encourages students to be self-directed and independent, which is empowering and engaging for learners. The focus of supporting students must be to encourage them to become more independent.

B. Social and Peer-to-Peer Learning

With regard to social learning, it is important to ensure that there is a place where students are confident in asking questions, such as a smaller support class or a tutorial session [6]. As part of these sessions, students can benefit by engaging in articulation of learnt material and discussion of their learning with peers. The comments from their peers can help to modify any misunderstanding of the educational material. This learning conversation plays an important part in student learning processes as shown by the research of Laurillard [2] and engages Vygotsky's theories on social learning. Research on learning done in collaboration with peers has shown that

students strive to perform better than they would on their own [2, pp. 187-209]. Vygotsky's theory of the zone of proximal development (known as ZPD) describes the place, or zone, that computing support assists the student with. The ZPD is the gap between the zone of development a student can reach by independent learning and the zone of potential development that requires collaboration or tutoring with a more capable other [7, p. 86]. Within the theory of ZPD, Chalkin identified what he termed the 'assistive assumption' in Vygotsky's theory, this outlines the state of the tutor who assists the student. The tutor needs to be flexible and observant, able to guide while not being fixed in their wish to see a specific outcome [8]. The tutor must facilitate a more individualised approach when working, this approach is valid both with individual tutoring sessions and with support classes.

C. *Deep versus Surface Learning*

Deep versus surface learning plays an important part in the activities delivered by the tutoring framework. Surface learning is where the student's intention is to memorise facts, to grasp enough to show achievement and reproduce facts, all with a view to managing their assessments. This type of learning is generally not retained over time and if learned for assessment, it may be forgotten quite fast. Conversely, deep learning is much more than the retention of facts. Learners are engaged, reflect on and are thoughtful about their learning. Deeper learning also allows for students to be able to move from mere retention of facts to creation. Blooms Revised Taxonomy [9] categorises this as starting with remembering information, moving to understanding. This is important in terms of computing supports as it looks to ensure that the foundational knowledge is in place before moving on. Progress can be seen when the student demonstrates analysis, evaluation of knowledge and then moving to creation. Thus, students show their use of the new knowledge to create something new, and the potential to apply the learning in new and abstract ways [9, p. 5].

Deep learning occurs when students are actively engaged with their learning. Active learning involves using higher-order thinking skills and is shown to be more effective when teaching STEM subjects [10] [11]. Higher-order thinking skills include analysis, evaluation and integration of learning. Students engaging in reading, writing, discussion and problem solving would be demonstrating active learning [12, pp. 33-34]. Here the key aspect is being engaged, where the student is commenting, asking questions and reflecting on what they are learning. In this process the internal organization and integration of knowledge is happening [2] [11].

D. *Academic Supports in Higher Education*

Research has shown that students often struggle when first introduced to computer programming topics [13] [14] and providing support early can make a difference to a student [15] [16] [17].

Sheridan et al. [18] have investigated the transition from second level education to higher education experienced by computing students and the challenges they face in terms of getting to know a new learning environment and the support they need to assist them with their learning. A novel higher education transition framework (S^3F) that combines three types of student feedback-based adaptive support activities in terms of Learning Environments, Academic Subject and Social interaction was introduced.

A Terminal Assignment Based Assessment Process Model was introduced [19] that provides faculty with structured guidance for the creation of student assessments that are valid, reliable and easily understood by students.

As dropout and failure rates are a major challenge with online learning, cluster analysis was used in [19] to identify low student engagement during the early stages of the course cycle. The aim of this research was to help instructors monitor student online engagement and provide timely additional supports to reduce attrition rates.

The Irish National Strategy for Higher Education to 2030 [20] states that students may need additional help as they often don't have all the necessary skills, especially in STEM subjects. Academic integration can be supported with extra classes and individual tutorials. A recent review of the academic supports for computer programming at third level education [21] shows that most Irish educational institutions do offer some form of programming support, with the focus being on first and second year undergraduate students. However, this support should be enhanced and extended to support students enrolled in all academic years.

III. Tutoring Framework For Computing Support

This paper introduces a novel tutoring framework for computing support that was deployed in School of Computing at National College of Ireland (NCI). The framework comprises of 3 pillars; staff who interact with and support students, pedagogies that inform the activities chosen, and the support activities themselves. This framework illustrated in fig. 1, is implemented over the course of the teaching term and beyond, to provide assistance to both students who are new to computer programming and students who may require some occasional help during their course.

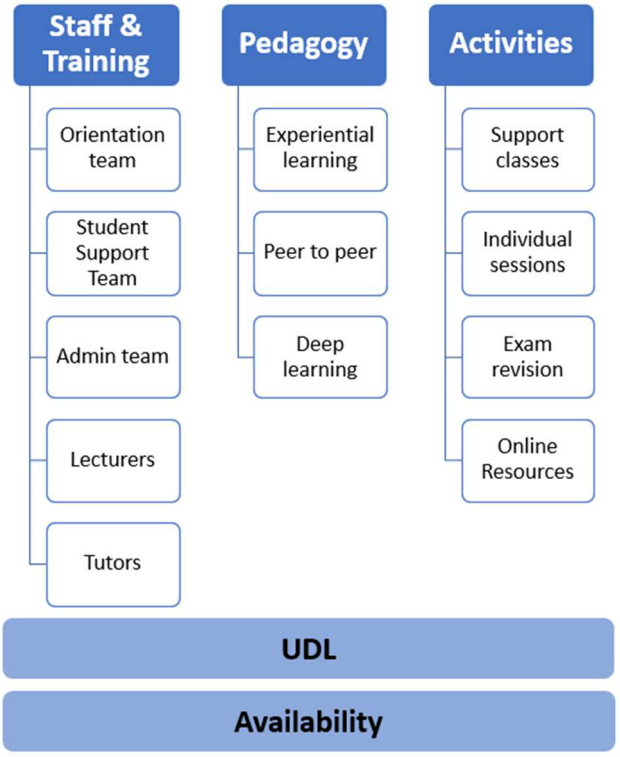

Fig 1. Tutoring Framework for Computing Support

A. *Staff and Training*

The staff and training pillar comprises of the staff who interact with students, this includes the orientation team, student support teams, the admin team, their lecturers and the Computing Support tutors. As part of this pillar training is also provided for the tutors.

In considering the stigma that can be attributed to accessing academic support services [22], the recommendation of Winograd and Rust is used to improve awareness on campus. From the week before the teaching semester starts, students are introduced to the computing support service. The orientation team walk the students past the Computing Support office when on campus and the open-door policy is highlighted to them. NCI's open-door policy in Computing Support says that if the door is ajar or fully open, staff are available for the student to drop-in. During orientation, the team encourages them to enrol in the Moodle page from the outset. When students in distress reach out to any of the college support teams or administration staff mentioning learning difficulties, they are directed to Computing Support as all aware of the service. All this helps to eliminate the idea, or stigma, that Computing Support is a remedial support. At the start of each term in NCI, the Computing Support team contact the Learning & Support team to know who is registered with this service and may require extra support. The team stay alert for any communication from those students. With all departments and teams working together, students will be directed to Computing Support when they reach out.

In terms of the Computing Support team there are 2 main full-time tutors, one who deals with students who attend college by day and one for those attending in the evening. The full-time tutors have qualifications in Computer Science and in Learning and Teaching. By having the full-time tutoring staff familiar with the theories and research of learning and teaching outlined in section I, they will understand the challenges the students face in their learning and know the types of interventions that are most effective. There are also a number of part-time tutors as part of the Computing Support team who are made up of 4th years, MSc students or teaching assistants. They are trained and supported by the full-time tutors. The use of students as tutor's is in line with the research on peer-to-peer learning [2].

Another recommendation to help further eliminate stigma is "facilitating service use" [21]. NCI uses a two-pronged approach for this. First is the addition of support classes for core programming modules to student timetables. Second is the presence of the full-time tutors as extra teaching assistants in lab sessions associated with computer programming related subjects. During the second week of the term, first and second year undergraduate students are visited by the full-time tutor. The full-time tutor also visits the Higher Diploma students who are taking conversion courses in the area of computer science. The lecturing staff are also informed of support classes and activities that are run by Computing Support so they can post this information to students and remind them of various support activities offered. When lecturers are aware of students who are struggling, they can refer them to the computing support service. The Computing Support team interacts with, and keeps the various staff up to date with the academic supports it provides. NCI looks for students to be informed and be comfortable with accessing the Computing Support service, by having informed staff who are continuously highlighting and directing students to Computing Support. Computing Support is a standing agenda item on School of Computing committee meetings thereby providing an opportunity to discuss, solve issues and plan.

As stated, part-time tutors are sourced from 4th year, Masters programmes or the pool of class assistants, as they are familiar with the module content. Part-time lecturers are also a valuable tutor resource for niche subjects. The part-time tutors are trained in ways to encourage questions and interaction from students and to work collaboratively with them. Highlighting the importance of practical, active and problem-based learning to encourage deeper learning is important so that the support work is effective. The tutor training consists of giving them guidelines, a format of a support class and an outline of what takes place during an individual session. Training takes the form of discussing scenarios that commonly arise and how the tutors might manage them, essentially the format of the training models what is done in a support session. The part-time tutors are continuously reminded that the full-time tutors are there to support them and answer any questions that might arise.

B. *Activities and Pedagogy*

The activities provided as part of the proposed tutoring framework are support classes, individual tutoring sessions, exam revision support and resources on the virtual learning environment, Moodle. These are all underpinned by theories of learning and teaching and are intrinsically linked.

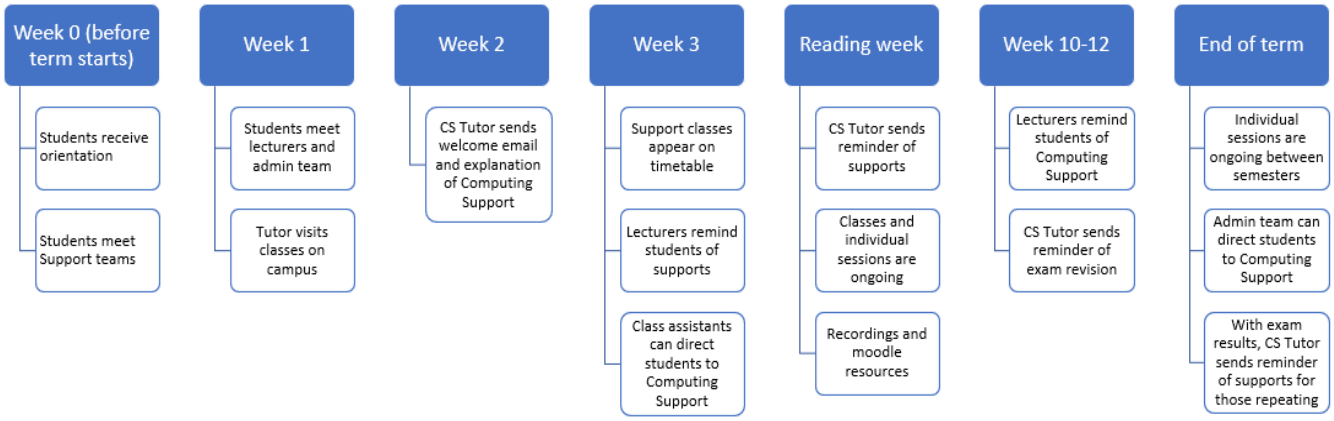

Fig 2. Timeline of Computing Supports during the semester

From week 3 of the teaching term, weekly support classes for core programming topics such as Java, MySQL, Python, HTML and CSS are automatically set up and added to student timetables. Lecturers involved in teaching modules related to the core topics are informed and asked to highlight to their students the weekly support classes as a place to get extra practice, ask questions and get help when studying All the above is done [23] with the aim of making computing support more mainstream.

Large classes can prove too daunting for some students to ask questions and may move at a pace that others find very fast. Support classes do not have specific learning outcomes and both pace and content is driven by the students. This means at the start of a support class, the students are asked what topics they would like to address and whatever the majority request, that is what is covered during the class. Thus support classes provide a space where it can be easier to ask questions and get to grips with module content at a slower pace. It means that the tutor needs to be familiar with the module content, this is where using 4th year students or teaching assistants is beneficial. The support classes proceed with describing a problem that incorporates their topics, then brainstorming with the students how to identify and plan the solution. Then time is given for the students to try work it out on their own or in pairs. Then they all come together to work on the solution which is live coded by the tutor. This approach uses problem-based learning, experiential and peer-to-peer learning all working to encourage deeper learning, with lots of interaction.

Students can also request an individual session, working one to one with a tutor for about 45 minutes to an hour. They can drop-in and see if one-to-one tuition can be provided on the spot or they can email and book a time slot.,These sessions are also practical in nature, with lots of interaction. With the theory of ZPD in mind, the tutor allows a student to reveal where they are having difficulties or need help. The tutor can also ask questions to determine the level the student is comfortable at and consider what the student needs to know to be able to progress. The tutor then makes use of various teaching methods outlined to assist the student to progress from there. By assessing and assisting students from the place where they are stuck, not where they should be in terms of the topics covered in the module to date, means that the support is delivered in an equitable and individualised manner.

At exam time, a series of revision classes are put in place by the Computing Support team to assist students in revising learning materials. The exam supports cover how to answer a past exam paper in terms of understanding marking rubrics, planning how to address exam questions, understanding what is being asked in exam questions and how to identify key aspects of problem domains.

There is also a dedicated page on Moodle for the Computing Support service, with resources in multiple formats based on UDL [24] and dyslexia guidelines. Resources include code videos, template code files, extra problems, recordings of support classes and more. Extra resources are created each semester and the Moodle page is reviewed annually.

The Computing Support service strives to be as a flexible as possible providing accessible resources for all. It supports full and part-time students in the School of Computing delivering group and individual supports through multiple means: online, face to face and through dual-delivered sessions. Dual-delivery involves having online students join the in-class students via a MS Teams meeting which is streamed to the room via projector. It also provides pre-recorded resources and live sessions. Computing support activities are ongoing and information is provided throughout the semester, with class events showing on student timetables. Fig. 2 shows that students first receive information on the service in the week before the start of the semester and then support events are timetabled from week 3. The support classes continue to the end of the semester with exam revision sessions taking place before and during the exam period. When considering new supports, the team looks to research to ensure the best use of the short time spent with students. Research has shown that supporting students for success empowers them and when they see themselves achieving, it can be a strong motivator in helping them to stay in college. By implementing the framework over the course of the teaching term, the availability of Computing Support is being continuously highlighted. Thus providing the information to ensure that students can access the relevant support needed for their success in programming.

IV. Discussion

Computing support centres are becoming more main-stream at third level [21]. NCI has had a Computing Support service available to students since 2013, with full-time tutors that are both expert in pedagogy and computer science. NCI's support classes have been delivered in multiple ways, looking to support students who may have disabilities, caring responsibilities or very little time for extra travelling. Since 2015 most activities were delivered either online or in person, however since 2017 an increasing number of support classes are being dual-delivered to facilitate access for students.

Individual supports have been available in person and since 2015, they have been available online. They can be accessed by dropping in or by appointment. Since the pandemic was declared in 2020, given the previous familiarity with online, it was an easy transition to deliver all computing support activities online. To replicate the drop-in service, MS Teams chat and phone functions were used to respond to students as soon as possible and calling if timetables permitted. As the service was well established and known to staff, with the tutors already embedded in module classes, there was no drop in participation with the transition. In fact there was an increase in students accessing supports.

The number of support classes delivered over the year has grown from 13 classes over the teaching term in 2020 to 16 in 2021. Participants in attendance at the classes increased by approximately 14% to participation of 1,489 students across the classes. Individual sessions also showed growth. Between 2019 and 2020 there was 5% growth in number of students attending individual sessions and between 2020 and 2021 there was a significant increase with growth of over 70% in the requests for computing support sessions. Exam revision sessions also showed an increase with more classes being delivered and with increased attendance.

Students have shown great appreciation for the service, providing feedback and posting on LinkedIn and Twitter. Comments include "I couldn't have got through it all without you and I appreciate everything you did for me", "Computing support is a somewhat unsung but incredibly valuable resource.", "I thank my lecturers, staff at NCI and in particular

Emer Thornbury in computing support for supporting me through a difficult academic year", "Computing Support and Emer Thornbury genuinely is the main reason I didn't drop out. #girlsinstem".

V. Conclusion

The aim of the research presented in this paper was to investigate to what extent a novel tutoring framework supports computer science students in their technical and practical programming modules at third level. The framework provides academic support through a combination of three pillars namely staff and training, pedagogy, and activities. Research shows that students entering third level computer science courses often need additional support with programming related subjects. The full-time tutors, with knowledge of both computer science and pedagogy, use the research in pedagogy to find innovative and effective ways to support students in computer science topics. Full-time tutors train part-time tutors to assist in the delivery of the various supports. The proposed framework includes support delivered in multiple ways through classes, individual tutoring sessions and resources, in line with the principles of UDL. All these activities are delivered in multiple formats; online, face to face, and dual-delivered and are available both during daytime and evening. Results demonstrate an increase of 14% in student attendance for the support activities compared to a previous year. In addition, student feedback demonstrates how valuable the framework is to them with comments such as "Computing Support is the main reason I didn't drop out".

References

[1] J. Dewey, Experience and Eduction, New York: Macmillan, 1938.

[2] D. Laurillard, Teaching as a Design Science: Building Pedagogical Patters for Learning and Technology, New York: Routledge, 2012.

[3] J. Biggs and C. Tang , Teaching for Quality Learning at University: What the Student Does., Maidenhead: Open University Press and McGraw-Hill, 2011.

[4] S. B. Merriam, "Andragogy and Self‐Directed Learning: Pillars of Adult Learning Theory," *New Directions for Adult and Continuing Education,* pp. 3-14, 26 February 2002.

[5] A. Chis, A. Moldovan, L. Murphy, P. Pathak and C. Muntean , "Investigating Flipped Classroom and Problem-based Learning in a Programming Module for Computing Conversion Course," *Educational Technology & Society,* vol. 21, no. 4, p. 232, 2018.

[6] I. Stamouli, E. Doyle and M. Huggard, "Establishing structured support for programming students," in *34th Annual Frontiers in Education*, 2004.

[7] L. S. Vygotsky, "Interaction Between Learning and Development," in *Readings on the Development of Children*, New York, Scientific American Books, 1978, pp. 34-40.

[8] S. Chaiklin, "The zone of proximal development in Vygotsky's analysis of learning and instruction.," in *Vygotsky's Educational Theory and Practice in Cultural Context*, Cambridge, Cambridge University Press, 2003, pp. 39-63.

[9] L. Anderson, D. Krathwohl and B. Bloom, A Taxonomy for Learning, Teaching and Assessing, London: Longman, 2001.

[10] S. Freeman, S. Eddy, M. McDonough, M. Smith, . N. Okoroafor, H. Jordt and M. Wenderoth, "Active learning increases student performance in SEM," *Proceedings of the National Academy of Sciences of the United States of America,* pp. 8410-8415, 2014.

[11] J. Allen, K. Clarke and M. Jopling, "Effective Teaching in Higher Education: Perceptions of First Year Undergraduate Students," *International Journal of Teaching and Learning in Higher Education* , pp. 362-372, 2009.

[12] C. Bonwell and J. Eison, "Active Learning: Creating Excitement in the Classroom," ERIC Clearinghouse , Washington DC, 1991.

[13] L. Thomas, M. Ratcliffe, J. Woodbury and E. Jarman, "Learning Styles and Performance in the Introductory Programming Sequence," in *SIGCSE02: The 33rd Technical Symposium on Computer Science Education*, New York, 2002.

[14] T. Beaubouef and J. Mason, "Why the high attrition rate for computer science students: some thoughts and observations," *ACM SIGCSE Bulletin,* vol. 37, no. 2, pp. 103-106, 2005.

[15] K. Nolan, S. Bergin and A. Mooney, "Facilitating student learning in Computer Science: large class sizes and interventions," in *International Conference on Engaging Pedagogy*, Dublin, 2015.

[16] C. Muntean, N. El Mawas, M. Bradford and P. Pathak, "Investigating the Impact of an Immersive Computer-based Math Game on the Learning Process of Undergraduate Students," in *48th Annual Frontiers in Education*, San Hose, 2018.

[17] N. E. Mawas, I. Ghergulescu, A. Muldovan and C. Muntean, "Pedaggical based Learner Model Characteristics," in *Ireland International Conference on Education*, Dublin, 2018.

[18] P. Stynes, S. Cogan , F. Sheridan and p. Pathak, "A Process Model for Transitioning to Alternative Assessment and Authentication in response to the COVID19 Emergency Restriction," in *IEEE World Engineering Education Conference (EDUNINE 2022)*, 2022.

[19] K. Palani, P. Stynes and P. Pathak, "Clustering Techniques to Identify Low-Engagement Student Levels," in *13th International Conference of Computer Supported Education*, 2021.

[20] "National Strategy for Higher Education to 2030," Department of Education and Skills, Dublin, 2011.

[21] M. Noone, A. Thompson, F. Galvin, M. Ward, K. Nolan, E. Thornbury, J. Andrew and D. William, "A Review of the Supports Available to Third-Level Programming Students in Ireland," *All Ireland Journal of Teaching and Learning in Higher Education,* vol. 14, no. 2, 2022.

[22] G. Winograd and J. Rust, "Stigma, Awareness of Support Services and Academic Help-Seeking among Historiacally Underrepresented First-Year College Students.," *Learning Assistance Review (TLAR),* vol. 19, no. 2, pp. 17-41, 2014.

[23] F. Sheridan, E. Thornbury, L. Murphy, C. Muntean, P. Pathak and P. Stynes, "A Framework for Managing the Transition from Second Level Education to Higher Education in Response to the COVID19 Emergency Restrictions," in *Frontiers in Education*, Uppsala, 2022.

[24] CAST, "CAST: About Universal Design for Learning," 2022. [Online]. Available: https://www.cast.org/impact/universal-design-for-learning-udl. [Accessed August 2022].

[25] E. Thornbury, F. Sheridan, C. Muntean, P. Pathak and P. Stynes, "A Tutoring Framework to Support Computer Science Programmes in Higher Education," in *EDUNIN23*, Bogota, 2022.