Performance Indexes for Assessing a Learning Process to Support Computational Thinking with Peer Review

Walter Nuninger[a]
Univ. Lille, CNRS, Centrale Lille, UMR 9189 CRIStAL, F-59000 Lille, France

Keywords: Reflexive Learning, Formative Assessment, Problem Based Learning, Test Unit, Self-Evaluation, Cross-Evaluation, Peer Review, Hybrid Learning, Digcomp.

Abstract: This article focuses on digital skills as defined in European Digital Competence Framework for Citizen (DigComp). In the framework of a hybrid and blended course, a formative pedagogical scenario is proposed. The training process consists of a formative situation of agile development of an application, supported by a gradual process of evaluation with and by peers called SCPR. The proposal is the result of several years of continuous improvement with engineering students enrolled in the IT module for non-developers. Learning outcomes relate to Computational and Algorithmic Thinking (CAT). It is then possible to compare the impact of our standard course design over several years with the group enrolled in full-time initial training between 2021 and 2023. A 3-index set, including the counter-performance index, enables us to analyse the effect of the pedagogical device on learning profiles, and the evolution of positive feelings and difficulties experienced. Qualitative data confirm the project's benefits and trainers' role in terms of student involvement and perspective-taking, and provides information on the impact of the previous training path and the obstacles. The proposed indicators confirm the pedagogical proposal and guide future prospects towards more relevant indicators for monitoring CT learning within the DLE framework.

1 INTRODUCTION

The European action plan for 2030, "*the way forward for the digital decade*" (EU no. 2022/2481), is aimed at job retention, adult education and social inclusion. This is pressing need due to the digitalization of society, reinforced by the health crisis of 2019, which has changed usage, but also the promising developments embodied by artificial intelligence and its integration into professions (including training).

Given the societal role played by future chartered engineers in France, training programs must meet these requirements. The learning outcomes cover the full range of professional skills of the coach, leader and manager, as well as the specific area of scientific expertise. Firstly, collective and social intelligence must be developed to enable beneficial interaction to achieve a common, shared goal in disrupted environments. Secondly, digital skills are essential for using digital technology in professional practice and learning to meet employment challenges. Since 1900s, issues of quality and learning performance in higher education have contributed to the promotion of active pedagogies and competency-based approaches (Gervais, 2016).

The pedagogical scenario presented in this article fits into this framework to support the computational thinking (CT) of non-development engineers. This hybrid blended-oriented course integrates a formative situation (Raelin, 2008) to which a progressive formative evaluation process is attached (Nuninger, 2024). The aim is to encourage involvement and collective intelligence for team-based learning based on a shared project. The learning-by-doing approach calls for an active posture on the part of the students and support (Grzega, 2005). The trainer explains the concepts, guides without solving problems, but corrects to ensure production conformity providing feedback (Hattie & Timperley, 2007). Nicol & Macfarlane's (2006) 7 principles enable tutors to support self-directed and reflective learning: clarify expectations, promote self-assessment, provide quality feedback, encourage communication and positive mindsets, and enable improvement.

[a] https://orcid.org/0000-0002-2639-1359

Nuninger, W.
Performance Indexes for Assessing a Learning Process to Support Computational Thinking with Peer Review.
DOI: 10.5220/0013361100003932

In *Proceedings of the 17th International Conference on Computer Supported Education (CSEDU 2025) - Volume 2*, pages 815-822
ISBN: 978-989-758-746-7; ISSN: 2184-5026

1.1 CT and Chosen Curriculum

According to Shutes et al. (2017), computational thinking is "*the conceptual basis needed to solve problems effectively and efficiently (i.e., algorithmically, with or without the help of computers) with solutions that can be reused in different contexts. (It is) a way of thinking and acting, which can be demonstrated through the use of specific skills, which can then become the basis for performance-based assessments of numerical skills*". This reflects varied learning goals and priorities that go beyond the knowledge object (Baron et al., 2014). The deployment of artificial intelligence only reinforces the need for data, safety and networking in a world in transition. The underlying expectations are the use of digital technology for learning activity, work and compliant digital production. The Digital Competence European framework for citizen (DigComp 2.2) meets such requirements through 21 capabilities, divided into 5 domains (Vuorikari et al, 2022) given in Table 1. The targeted skill levels for engineers range from professional to expert (grades 4-8), covering intermediate, advanced and specialist levels. The curriculum covers hardware, networks, software, and data representation, along with functional analysis, algorithm description language, and application development methods for clean code and easy-to-maintain solutions (Martraire et al., 2022), such as test-driven development (iterative unit testing and refactoring). The emphasis is not on coding languages, but on development processes, i.e. the concepts of abstraction covering data and performance, generalization for digital transfer and decomposition through valid algorithms as sets of efficiently assembled instructions.

Table 1: skill repository and chosen CAT curriculum.

DigComp 2.2	Learning by doing and by using
Information data literacy	file format, meaningful naming, structure of list, mob-programming
Communication Collaboration	LMS, share storage space (cloud), Agile mindset and RAD
Digital content creation	Pair- and Peer- programming Refactoring and clean coding
Safety	Unit testing and code review
Problem solving	Top-down functional analysis, test driven development, pseudocode

1.2 Formative Digital Production

At the heart of our proposal to support the computational and algorithmic thinking (CAT) is a digital production (Figure 1). It is a formative problem-based project that motivates Kolb's learning cycle (Kolb & Kolb, 2005): concrete experience, reflective observation, abstract conceptualization and active experimentation in a collective environment.

The progressive assessment process SCPR (Self and Cross Peer Review) attached to it is also formative, motivating questioning, feedback, decision-making and skills transfer (Nuninger, 2024; Thomas et al., 2011). The challenge engages students in the training, then the collective supports their learning (Falchikov, 2005; Sadler, 2010) thanks to: self-evaluation to give meaning and learning autonomy for personal effectiveness; cross-feedback for trust and action, then empowerment in a deeper learning act; and peer review as an active observer through shared, fact-based assessment.

The project's phasing invites students to discover the coding environment on their own, then to program in pairs, sharing the workstation display (one does the input while the other controls, both self-assessing). They are obliged to carry out unit tests to clarify the need, and then to review the code of the other batches. Finally, mob- and peer- programming involve assembling batches in an agile spirit with a view to reaching a common consensus (delivery) prior to the confrontation with competing teams.

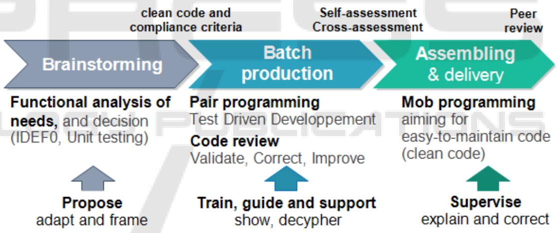

Figure 1: digital production training process with attached SCPR (above) and tutor's role (below).

1.3 Goals and Focus

In this paper, we review the enriched standard pedagogical scenario that integrates a formative digital production and progressive Self and Cross Peer Review process (SCPR) in the context of its use to support computational thinking. From 2021 to 2023, it is deployed in the full-time initial training (FIT) for engineers who are not computer developers. Section 3 then recalls the data collected for the three groups studied. The focus is put on the *counter-performance index* (cpi) and the two score variables ($SFpf_n$: p*ositive feeling*; $SFed_n$: *expressed difficulty*) introduced by Nuninger (2024). Following the results in section 4, section 5 focuses on the effect of the device on the *learning profile*. This is reinforced by unit testing. The final section concludes the paper.

2 PEDAGOGICAL DEVICE

The CT module learning outcomes for FIT students meet the requirements of chartered engineers and DigComp2.2 (intermediate to advanced levels). They must be skilled digital users, "*able to use problem-solving methods to define specifications and collaborate with experts to effectively lead digital or data-related projects, while remaining aware of the constraints and limitations associated with integrating digital innovations into organization and communication*". The underlying competencies are personal effectiveness and social intelligence. The chosen curriculum covers the concepts listed in the second column of Table 1. The teaching approach is based on the use of digital tools and the production of digital solutions within the framework of hybrid and blended-oriented courses (Figure 2). The 36-hour teaching unit program is divided into two equal parts, finishing with the one devoted to the collective rapid application development project. Upstream, 15 hours of online self-training begin on the Digital Support for Guided Self-Study (Nuninger, 2017), structured as a sequence of activities synchronized by completion tests on Moodle. The aim is to involve students, to develop their learning autonomy and organization, and to facilitate their grasp of the chosen coding environment Scilab prior to the project. The formative project focuses on the production of an application based on the initial code supplied, and the function packages to be developed. The data structure is imposed with the call sequences for conventional processing (read, write, add, delete, search...). In 2021, unit testing was introduced, and test-driven development reinforced this requirement in 2023. The project generates 3 outputs:

- **a set of skills with a level of expertise** built with the team and confronted with the group;
- **individual reviews on experience** that facilitates learners to take a step back;
- and **project productions** that value the work.

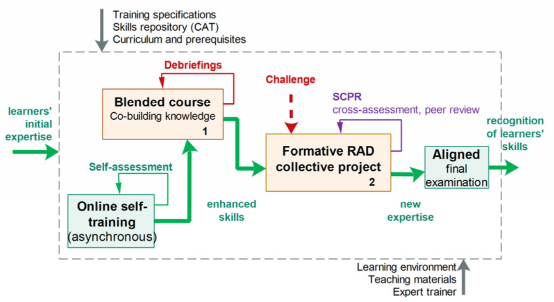

Figure 2: Standard hybrid blended-oriented course.

Validation of learning outcomes is based on the average of project and final exam marks (out of 20). Social intelligence is not directly assessed, but does have an impact on the results of the project-based practical work. A positive effect of the activity should be reflected in the final individual mark. The aligned final exam is based on the following evaluation criteria sorted by increasing level of difficulty:

- mastery of standards (IDEF0, pseudo-code);
- assignment, read/write, iteration, alternative;
- understanding of lists and structures (pointers);
- able to develop a digital solution by assembly;
- able to debug by data control (test unit);
- basic proficiency in Scilab for clean code.

3 EXPERIMENTATION

We are interested in the three FIT groups that started a 3-year chartered engineer training in biotechnology and agri-food sector in 2021, 2022, and 2023. The CAT module occurs in the second semester of the academic year starting in September. The mean age is around 20.4 years (20 in 2022), with age ranges from 19 to 24 years (19 to 23 in 2022). In recent years, the number of students has fallen (-34% with respect to 2021) with the ratio of women to men dropping from 6.4 to 2.2, and a change in the distribution of previous training paths (Table 2). Two senior teachers are involved in the teaching unit each year (Table 3), with the course leader (SL1) and a permanent trainer (SL3) who has replaced the temporary substitute since 2022.

Table 2: Group characterization with training path.

Groups	FIT2021	FIT2022	FIT2023
flow ; women %	50 ; 86%	45 ; 74%	33 ; 67%
2 y. technology	7	7	1
2 y. Bach. of Sc.	18	9	6
preparatory path	17	15	12
Preparatory class	8	14	14

Table 3: Groups Project by Trainers (Student Numbers).

Senior lect.	FIT2021	FIT2022	FIT2023
SL1 leader	2 gr. (24)	2 gr. (21)	2 gr. (22)
SL2 substitute	2 gr. (26)	-	-
SL3 permanent	-	2 gr. (24)	1 gr. (11)

3.1 Data Collection and Processing

The data are collected primarily for educational purposes, as explained and carried out in Nuninger (2024). Pre- and post-module questionnaires provide the qualitative data from which score variables are constructed to complement the quantitative data from the assessments. The compulsory personal evaluation questionnaire at the end of the project makes it possible to identify *clear positive* and *clear negative* feelings, and the absence of opinions. The data are anonymized once the various sources have been linked, but contextualized by group and trainers. Incomplete data that cannot be reconstructed, data relating to specific situations, or data in insufficient numbers are rejected for this study. Redundant variables are not pre-selected for the regression study. Data processing is done using Excel, Scilab, and R. Satisfaction survey respondent rates vary from 47% to 58% (max 29) depending on group and year. This is why we mainly compare the years 2021 and 2023, focusing on full groups and SL1' groups (Table 3) to limit biases related to professional practice and style, and insufficient or missing data in 2022.

3.2 Score Variables

The score variables (Table 4) introduced by Nuninger (2024) are the normalized min-max sums over the interval [1,2] of the measures (yes/no) relative to 4 sets of descriptors at the beginning (B) and/or finish (F) of the course. *Difference variables* (prefix Δ) show changes in learner responses throughout the course. The first pair (pp, ee) describes the ***learning profile***, while the second (pf, ed) reflects the ***final feelings*** after completing the course. By construction, only a finite number of values are possible (Table 5).

Table 4: Descriptors sets and score variables (letter S).

Refers to	Descriptors
Pedagogical preference ($SBpp_n$; $SFpp_n$)	pp1: listen to the lesson pp2: prepare and ask in class pp3: teamwork
Evaluation experience ($SBee_n$; $SFee_n$)	ee1: personal assessment ee2: assessed colleague ee3: confronted in a team
Positive feeling ($SFpf_n$)	pf1: motivating to be evaluated pf2: rewarding to evaluate others pf3: peer review is useful
Expressed difficulty ($SFed_n$)	ed1: difficult to evaluate oneself ed2: difficult to evaluate others

Table 5: normalized score variable (subscript *n*).

Score	$SFed_n$	$SFpf_n$, SB/Fee_n, SB/Fpp_n
Final value (possible configurations)	1 (1) 1.5 (2) 2 (1)	1 (1) 1.33 (3); 1.67 (3) 2 (1)

3.3 Counter-Performance Index (CPI)

Table 6 explains the meaning of the *counter-performance index* (cpi) based on its range, defined as the ratio of the normalized scores of final *difficulty expressed* ($SFed_n$) to *positive feelings* ($SFpf_n$). An *expressed difficulty* does not necessarily mean a non-positive experience, as both can be experienced in the same way (=1). cpi only compares the two scores, without judging the cause. The cpi normality study is based on 6 classes centred on class 3 (Table 7) with a mean width of 0.25 ranging from 0.075 to 0.5 (nested mean method), taking into account the permitted values of the index and the [min, max] number of respondents in the groups to be compared ([7; 29]).

Table 6: Meaning of cpi range and values in [0.5; 2].

Range	cpi=$SFed_n/SFpf_n$	Underlines the
> 1	2; 1.5; 1.2; 1.125	difficulty felt; >$SFpf_n$
= 1	$SFdf_n = SFpf_n$	equilibrium
< 1	0.5; 0.6; 0.75; 0.9	positive feeling; >$SFed_n$

Table 7: Meaning of chosen cpi classes (width, values).

cpi class	width	values (configurations)
1: more positive	0.250	0.5 (1); 0.6 (1); 0.75 (2)
2: felt positive	0.175	0.9 (2)
3: balance	0.075	1 (2), i.e., no difference
4: felt difficult	0.175	0.125 (1)
5: more difficult	0.325	1.2 (1) ; 1.5 (2)
6: much more	0.5	2 (1)

4 RESULTS

4.1 Evolution of Assessed Levels

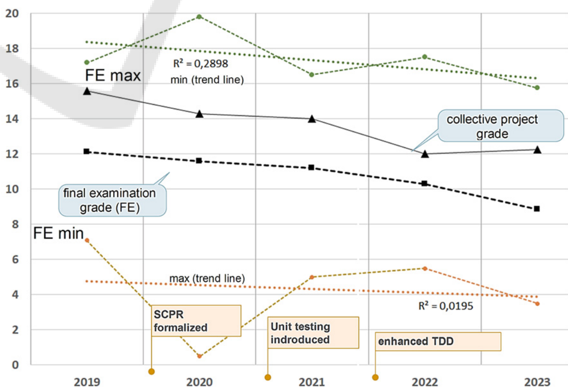

Figure 3: Evolution of project and final exam grades (/20).

Analysis of the scores reveals a difference in final levels, influenced by group building and trainers' style, but which is difficult to specify due to changes over the period. The gap between final exam grade narrows compared to 2019 and 2020 (Figure 3). This suggests, firstly, an effect of the pedagogical proposal, which enables a greater transfer of expertise within the groups, as noted by one student in 2023: “*we learnt at different speeds, but the exchanges within the group enabled us to progress faster*”. Secondly, the impact of trainers' professional style, as highlighted in 2023: “*I appreciate that you tried to motivate us. You listened to our difficulties and didn't let us give up at the beginning*” or “*we were less helped during the project than during the course”.*

4.2 Positive Feedback and Barriers

Based on open-ended questions, we have identified *clearly positive* and *clearly negative* opinions; others are neutral (Table 8). In 2023, for all trainers and groups, there are fewer *clearly negative* responses, while the rate of *clearly positive* responses increases. Table 9 presents the main obstacles grouped according to 6 main dimensions identified:

1. Documentation (understanding of goals);
2. Environment (code syntax, lexicon);
3. Beginner (level, heterogeneous group);
4. Communication (group organization)
5. Allocated time (want more);
6. Commitment (late, starting a priori).

Table 8: *clearly positive* versus *clearly negative feelings* expressed (over the respondents; others have no opinion).

FIT	Full	SL1	SL2/SL3
2021	12-36 % (20)	8-38% (9)	15-35% (11)
2022	20-61 % (46)	10-71% (21)	28-52% (24)
2023	7 -86 % (28)	6-89% (18)	10-80% (10)

Table 9: Obstacles expressed by respondents (sorted).

Barriers	2021	2022	2023
No opinion	**88%**	**38%**	**19%**
1. Documentation	8%	17%	16%
2. Environment	0%	13%	22%
3. Beginner	4%	15%	16%
4. Communication	0%	8%	9%
5. Allocated time	0%	2%	13%
6. Commitment	0%	6%	6%

The first 2 obstacles are linked to the challenge, which consists in expressing the customer's needs and appropriating the initial codes supplied. The next 2 reflect the influence of the group (heterogenous) and the previous training paths which also explain the obstacles linked to the chosen coding environment (Scilab compared to Python, which some students were familiar with). The last two reflect the difficulties encountered due to lack of time, but also the desire to complete digital production successfully.

4.3 Qualitative Feedback

In 2021, personal assessments show a more positive trend than in previous years, but remain focused on individual behaviour (fear, lack of commitment, objectivity) and the teacher's responsibility in the final grade despite an activity recognized as federating (Nuninger, 2024): “*Despite initial difficulties, I got back on track, completed the work (and became) more efficient in testing to find errors in others' code.”; “we're proud of the work we've done in the time available”; “we were creative, (but had) difficulties in coding”; “It interested me (but) I would have liked more step-by-step lessons”; “enriching experience but a complex organization”.*

In 2023, satisfaction levels are higher and show a real sense of perspective on work, results and individual and collective responsibility: “*I had coding experience but I adapted to Scilab”; “despite difficulties, novices were motivated and committed, developing programming logic with unit tests”;* “*we did a good job. There are still mistakes but we've improved*”; “*The project gave meaning to the course*”; “*The project is rewarding, motivating and requires us to communicate well*”; “As *a novice, I'm proud of my progress”.*

But some negative feedbacks remain such as: “*I didn't enjoy the experience too much, perhaps because of a lack of interest in IT and a lack of involvement. I didn't feel I had learned much”, “I found it frustrating to rely on others' functions to progress. Their errors and omissions caused delays (much like in the professional world, you might say)”.*

The school's satisfaction survey highlights these aspects (Figure 4), showing increased confidence in expertise identified in individual assessments despite a decline in final results (Figure 3).

		FISE 2021	FISE 2022	FISE 2023
	Respondents to school survey* (rate)	**26** (52%)	**21** (47%)	**7** (21%)
	to TU final survey (rate)	not contacted	**44** (98%)	**26** (79%)
Overview	**Adapted pedagogy***	- (30%*)	- (34%*)	- (33%*)
	Overall satisfaction (*school survey)	**35%** (32%*)	**56%** (11%*)	**-** (33%*)
	I made progress (*school survey)	- (36%*)	**84%** (26%*)	**92%** (50%*)
Learning outcomes	**Code basics and pseudo-code**	-	**95%**	**85%**
	Functional analysis (IDEF0)	-	**91%**	**85%**
	Code performance, clean code (TDD)	-	82%	69%
	Understanding of the coding Env.	-	82%	81%
	Assembling, Debugging skills	-	75%	**88%**
	No understanding of these concepts	**-**	**5%**	**8%**

** school survey otherwise teacher surveys ; data not available : -*

Figure 4: school surveys and individual reports analysis.

4.4 Counter-Performance Index

The beneficial aspect of the proposed pedagogical device increases further between 2021 and 2023 (Figure 5) with a cumulative frequency rate rising from 41% to 50% for cpi inferior or equal to 1 (Table 10). The asymmetrical shape of the histograms shows two distinct populations. For the SL1's groups, this was the case in 2021, but not in 2023. The distribution is normal with a cumulative frequency rate dropping from 50% to 36% (Table 11). This conclusion differs for the groups followed by SL2 and SL3, but the numbers are insufficient to draw any further conclusions other than the impact of trainers.

Table 10: *counter-performance index* in 2021 and 2023.

Full FIT group in	2021	2023
Respondents (rate)	29 (58%)	18 (54%)
beneficial:1, 2 (<1)	24%	33%
balance:3 (=1)	17%	17%
difficult:4, 5, 6 (>1)	59%	50%
mean (std)	1.32 (0.47)	1.16 (0.33)

Table 11: *cpi* for the SL1's groups in 2021 and 2023.

SL1's groups	2021 (SL1)	2023 (SL1)
Respondents (rate)	12 (50%)	7 (50%)
beneficial:1, 2 (<1)	33%	27%
balance:3 (=1)	17%	9%
difficult:4, 5, 6 (>1)	50%	64%
mean (std)	1.83 (0.41)	1.22 (0.38)

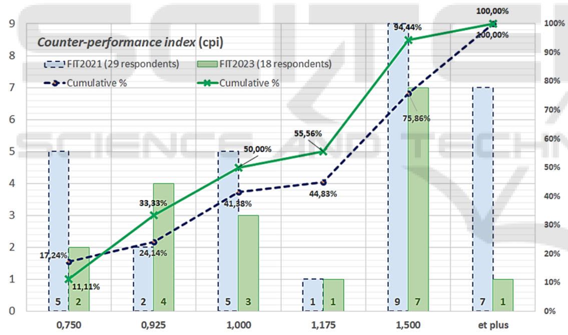

Figure 5: *cpi* histograms and cumulative frequencies for full groups in 2021 and 2023 (29 and 18 respondents).

4.5 Performance Index and Grade

The analysis of end-of-course *learning profiles* ($SFed_n$, $SFpf_n$) against final grades is tricky due to the varied possible response configurations (due to score variables definition). By plotting the cpi in descending order, we can identify 3 to 4 meaningful areas in 2023 (Figure 6):

- cpi>1: a high *expressed difficulty* does not prevent a positive experience but probably favoured a higher final exam mark (area 1);
- cpi<1: a high *positive feeling* is favoured by a lower *expressed difficulty* (area 2),
- then the decay of the two score variables lowers the final score, while increasing them tends to improve the final score (area 3).

For the groups supervised by SL1, in 2023 (middle curve) the final mark decreases along with the 2 score variables (area 2, cpi<1), and even more so with increasing difficulty expressed (area 4). This was not the case in 2021 (top curve): indexes are decorrelated from the final grade.

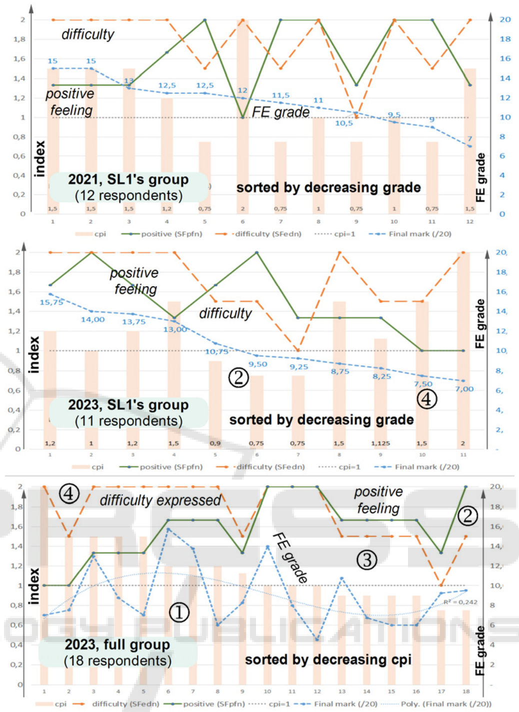

Figure 6: cpi, $SFed_n$, $SFpf_n$ and grade (/20) for full group in 2023 (bottom) and SL1's groups in 2021 and 2023 (above).

4.6 Key Variables Influencing Indexes

Among the p pre-selected inputs using linear squared correlation coefficient, our recursive identification process retains the relevant ones to describe (cpi, $SFpf_n$, $SFed_n$); i.e., minimum J_{np} (sum of residuals divided by (n-p); n being the number of respondents) ranging between 0.16 and 0.38 in 2021 and 2023. In 2023, *positive feeling* ($SFpf_n$) depends on the student's age but not in 2021, then of Δee1 (difference in *self-evaluation experience*). The difference of the *cross-evaluation experience* (Δee3) remains the relevant input of the *expressed difficulty* ($SFed_n$), in addition to the difference of *pedagogical preference* for lecture (Δpp1). In 2021, *counter-performance index* depended above all on the evolution of *pedagogical preference* with respect to *lecture*

(Δpp1) and then to *teamwork* (Δpp3), but in 2023, it's primarily the previous training path, then Δpp1.

4.7 Influences on Grades

The principal component analysis (PCA) shows that project and final exam grades are not influenced by gender, but might depend on students' previous training paths, and therefore on the groups assigned by the school, on which the project groups depend. In 2023 (Figure 7), the examination grade depends solely on the project grade (0.43 correlation). In previous years, however, the grade also depended on age and/or previous training paths, with correlations exceeding 0.25. This was also true for project grade in 2021, but not since 2022, indicating a real benefit of the project in improving student expertise in CAT.

PCA for	Final Examination grade			Collective Project grade		
	2021	2022	2023	2021	2022	2023
Fin. Exam. gr.	1	1	1	0,35	0.32	0.43
Col. Project gr.	0,35	0.32	0.43	1	1	1
Prev. Tr. Path.	0.25	0.38	0.11	0.47	0.09	0.08
Age	-0.29	-0.26	0.01	-0.21	-0.09	0.10

Figure 7: PCA focus on examination and project grades.

5 DISCUSSION

The 3-index set analysis indicates that the *expressed difficulty* results from students' increased awareness of learning outcome expectations and their effort required to resolve the project, while the *positive feeling* arises from the project's success and their sense of personal evolution. Students involved in the project have higher levels for both indicators and a better chance of passing the final exam (grade higher than 10/20). Conversely, a low level of *expressed difficulty* may reflect a false sense of mastery. The proposal positively impacts learning performance and satisfaction (Nuninger, 2024; Schein, 2013).

In 2023, at the end of the course, the net decrease (corrected for the increase) in students' preference for the classic course (pp1) is -6% for a preference expressed at 93% at the beginning (-14% in 2021, for 88%) in favor of active learning (pp2-3). In 2023, 72% of students recognize their experience of cross-assessment and peer review, compared with 64% in 2021 for similar starting values (56% and 57% respectively). Rates rise to 83% in 2023 and 75% if self-evaluation is included (62% and 68% at the beginning). The net evolution of self-assessment experience is +17% in 2023 (+11% in 2021). Peer interaction enhances pedagogical understanding and raises awareness of evaluation (Topping, 2009).

Early teaching of unit testing is beneficial, as it provides feedback (Scatalon et al., 2019). Students' personal assessments show increased knowledge and confidence in their computer skills. Debugging expertise has improved, although the final level is lower than expected. The reasons put forward are the persistent decline in the level of students recruited, the impact of the health crisis and the increased difficulty of the final exam with the integration of unit testing. Unit testing is challenging and increases cognitive load, especially for beginners due to the limited time available (Garousi et al., 2020).

Our pedagogical approach does not always mitigate the effects on students' grades of their previous training, the school-imposed groups, and the chosen pairings. In 2021 and 2022, previous training path is less dependent on age, but in 2023, the correlation is strong (-0.42, while -0.18 and -0.01 the previous years), confirming better-targeted recruitment. Observation during the sessions reveals the generational evolution. In 2023, students no longer focus on the grade, but really express a wish to understand, do and succeed in the challenge. It is the combination of age (*positive feelings*) and previous training (*cpi*) that contributes to acceptance of the pedagogical approach adopted and commitment, with the risk of disappointment (*difficulty expressed*). The trainer-tutor plays an essential role to compensate for the heterogeneity of learners' profiles, but is limited by classroom constraints (Sadler, 2010). One student points out: *"the amount of help given to the groups should be more evenly distributed to ensure fairness. It's hard to get all the groups with different concerns on the same track"*.

In 2024, to understand how students approach digital production and develop computational thinking (learning profile) during the project, first an online Kanban aims to collect the following metrics: time spent on tasks with version tracking (cycle time), time elapsed before task validation (execution time) and throughput (performance and productivity). Second, abstract syntax trees can help compare the algorithms of imposed function versions within a group and between different groups based on clean code criteria. Third, we are currently prototyping an automated data collection solution in the coding environment Scilab to identify coding processes. A first experiment took place in November with a group of apprentices to assess their level of acceptance of data collection and identify any difficulties in integrating the extension into Scilab. Main constraints lie in GDPR, data safety and storage, and GUI. Our device's instrumentation will enable comparison with other studies on computational thinking, even those using different coding environments like Thonny (a

Python IDE for beginners) with additional data-collection plugins (Marvie-Nebut & Peter, 2023). The final objective is to propose indicators for monitoring, guiding, and evaluating remotely (DLE).

6 CONCLUSION

The proposed standard teaching scenario focuses on skills through blended-oriented lessons and a formative digital production to develop computational thinking. The peer review process reinforces reflective learning. Despite the complexity of unit testing, the approach improves understanding of algorithms and their design, debugging skills and a willingness to validate solutions, helping future engineers gain perspective. According to data collected between 2021 and 2023, difficulty is strongly influenced by students' previous training path, in line with their age and social intelligence. The cognitive load of beginners can only be mitigated by more time devoted to them during the sessions and the professional style of the trainer-tutors; a parameter that has not been explored. The 3-index set (*counter-performance index, score variables of final expressed difficulty,* and *positive feeling*) demonstrates the effect of the device on learning and postures, and helps in learning profile analysis. However, it is not sufficient to fully analyze the learning processes of computational thinking.

To this end, larger student flows are required to overcome the limitations of this work, but the proposed training scenario is stable. The priority is to instrument the Scilab coding environment, then to identify students' coding processes in computational thinking, and to determine learning profiles using relevant contextualized indicators.

REFERENCES

Baron, G.-L., Drot-Delange, B., Grandbastien, M., & Tort, F. (2014). Computer Science Education in French Secondary Schools : Historical and Didactical Perspectives. *ACM Trans. Comput. Educ.*, *14*(2), 11:1-11:27. https://doi.org/10.1145/2602486

Falchikov, N. (2005). Improving Assessment through Student Involvement: Practical Solutions for Aiding Learning in Higher and Further Education.

Garousi, V., Rainer, A., Lauvås, P., & Arcuri, A. (2020). Software-testing education : A systematic literature mapping. *Journal of Systems and Software*, *165*, 110570. https://doi.org/10.1016/j.jss.2020.110570

Gervais, J. (2016). The operational definition of competency-based education. *The Journal of Competency-Based Education*, *1*(2), 98-106. https://doi.org/10.1002/cbe2.1011

Grzega, J. (2005). Learning By Teaching: The Didactic Model LdL in University Classes.

Hattie, J., & Timperley, H. (2007). The Power of Feedback. Review of Educational Research, 77 (1), 81–112. doi: 10.3102/003465430298487

Kolb, A. Y., & Kolb, D. A. (2005). Learning Styles and Learning Spaces: Enhancing Experiential Learning in Higher Education. Academy of Management Learning & Education, 4(2), 193-212.

Martraire, C., Thiéfaine, A, Bartaguiz, D., Hiegel, F. & Fakih, H (2022) Software craft (French) Ed : Dunod, 289 p., ISBN : 978-2-10-082520-2

Marvie-Nebut, M., & Peter, Y. (2023). Apprentissage de la programmation Python : une première analyse exploratoire de l'usage des tests. In J. Broisin, et al. (Éds.), *Actes de l'atelier Apprendre la Pensée Informatique de la Maternelle à l'Université. APIMU 2023* (p. 1-8). https://hal.science/hal-04144206

Nicol, D. J., & Macfarlene-Dick, D. (2006). Formative assessment and self-regulated learning: A model and seven principles of good feedback practice. Studies in Higher Education, 31(2), 199–218. doi:10.1080/0307 5070600572090

Nuninger, W. (2017). Integrated Learning Environment for blended oriented course: 3-year feedback on a skill-oriented hybrid strategy. HCI Int, 9-14 July. In: Zaphiris P., Ioannou A. (Ed.) *LCT,* 10295, Springer, Cham. doi: 10.1007/978-3-319-58509-3_13

Nuninger, W. (2024). Hybrid and Formative Self and Cross Peer Review process to support Computational and Algorithmic Thinking. In *CSEDU 2024* (Vol. 2). SCITEPRESS (Science and Technology Publication, Lda). doi:10.5220/0012712800003693

Sadler, D. R. (2010) Beyond feedback: developing student capability in complex appraisal, Assessment & Evaluation in Higher Education, 35:5, 535-550, doi: 10.1080/02602930903541015

Scatalon, L. P., Carver, J. C., Garcia, R. E., & Barbosa, E. F. (2019). Software Testing in Introductory Programming Courses: A Systematic Mapping Study. *50th ACM Technical Symp. on Computer Sc. Educ.,* 421-427. https://doi.org/10.1145/3287324.3287384

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142-158. https://doi.org/10.1016/j.edurev.2017.09.003

Raelin, J. A. (2008). Work-based Learning. Bridging Knowledge and Action in the Workplace. (New and revised edition). San Francisco, CA: Jossey-Bass.

Thomas, G., Martin, D. & Pleasants, K. (2011). Using self- and peer-assessment to enhance students' future-learning in higher education. Journal of University Teaching & Learning Practice, 8(1), 1-17.

Topping, K. (2009). Peer Assessment. Theory into Practice, 48(1), 20–27. doi:10.1080/00405840802577569

Schein, E. H. (2013). Humble Inquiry: The Gentle Art of Asking Instead of Telling". Berrett-Koehler Publishers

Vuorikari, R., Kluzer, S. & Punie, Y. (2022), DigComp 2.2: EU 31006 EN, doi:10.2760/490274, JRC128415.