Progression of tool usage in project-based IT courses

Justyna Szynkiewicz
Faculty of Social Sciences
Nord university
Steinkjer, Norway
justyna.szynkiewicz@nord.no

Robin I. Munkvold
Faculty of Social Sciences
Nord university
Steinkjer, Norway
robin.munkvold@nord.no

Line Kolås
Faculty of Social Sciences
Nord university
Steinkjer, Norway
line.kolas@nord.no

***Abstract*—The research full paper examines what kind of IT tools (software and hardware) are used in project-based IT courses, where students create digital artefacts like games, apps, audio, VR worlds, web sites etc. The analysis is based on data from 46 IT courses, collected during the spring and fall semesters of 2017 at two Norwegian universities. The analysis of these data has led to a classification of the tools used by students in PjBL and guidelines for ensuring progression of tool usage in course design and study program design. The IT tools were classified into the following categories: Collaboration; Communication; Developer Platforms; Physical Tools; and Project Management. The analysis is done with reference to the existing body of literature on active learning, e.g. constructivism and constructionism, and taxonomies for learning. The study discusses ways in which tools may be used to enhance learning in the Creating level of Bloom's revised taxonomy. The data was further analysed through a "lens of progression", resulting in guidelines for study program designers and course designers, with the goal of quality assuring the tool implementation and usage in project-based IT courses and study programs. The guidelines systematically structure the tool categories with the study program design and recommend an increasing student autonomy throughout the progression of the studies. The guidelines also focus on the implementation of professional tools in IT courses, as tools are influencing students' identity development. The goal of the study is to contribute to the quality assurance of digital tool usage in IT study programs and the project-based courses, where it is important to define pre-requisites and to ensure that students learn to use different digital tools in the right order and that students know the necessary tools for their discipline.**

Keywords—Technology Enhanced Learning, Active Learning, Create, IT Tools, Project-Based Learning

I. INTRODUCTION

Technology is part of our daily life and is being increasingly used as part of the educational system. During several decades, technology use within schools has been promoted and has influenced higher education, but as Kirkwood [1] states, "we should still acquire more information about its effective educational contribution." With a huge number of possible tools available, the question is related to how one should use and utilize these digital tools and resources to benefit the learning process of the students. Technology-enhanced learning (TEL) can imply the use of digital resources and tools within many areas and in different ways. One can use digital tools and resources in the same way as one is using books, but with easier access. One can also use these tools for collaboration, development, analyses, etc. It is, in other words, a complex system, which includes communities, technologies and practices that are informed by pedagogy [1].

The term Technology-Enhanced Learning refers to the application of Information and Communication Technologies for teaching and learning. The International Encyclopedia of Education [2] defines TEL as simply a more recent term for e-learning. E-learning, and so also TEL, is the use of any digital technology in support of teaching and learning [2].

In recent years, learning at university level has moved away from the traditional, teacher-centred style of lecturing [3]. In this model, students play a passive role and most of their time is spent taking notes and asking questions to the instructor. The shift is towards active learning, which is student-centered and, in some cases, enhanced by technology. Active learning increases students' understanding [4], improves their examination results and decreases their failures in STEM exams [5].

Learning from the social perspective is seen as process of becoming [6]. By involvement in learning activities at a University, students are developing an understanding of their future profession. In addition to acquiring disciplinary knowledge, they should gain skills, develop dispositions [7], and form their professional identity [6]. In IT professions, the majority of development work is done in a project mode [8]. Development tools, both software and hardware, that are used in that process can be seen as one of the building blocks of the professional identity.

The Centre for Excellent IT Education (Excited) aims to make Norway a pioneer within innovative IT education and make IT an increasingly attractive study choice for young people. One of the centre's goals is to increase students' motivation through project-based learning focusing on the development of products and services. As part of this, a focus on the term "Learning through Construction" is established, meaning a project-based learning method where the creating of a digital artefact (digital game, app or similar) is central [9]. Active learning is usually more effective than traditional lecturing [10]. While in traditional classroom settings students are passive receivers of information that the teacher provides, active classrooms allow students to be more engaged in the learning activity, and also allowing them to be active creators as part of the learning process. Over the course of the learning process they often develop a digital product while working in groups [9], giving the basics of LtC courses, where one aims to involve students in a practical, hands-on and real-life experience, through active learning.

This paper focuses on TEL in LtC courses in higher education and aims to find and categorize what types of technology is used to enhance learning in project-based IT courses, in order to quality assure the usage of tools in project-based learning during different stages of a study program, by the development of tools progression guidelines for the design of study programs and LtC courses.

II. Theory

Related theory, which is relevant to this study includes theories on learning through making, connecting this learning approach to taxonomies for learning. The theory section will also present previous relevant studies on tool categorization, and theories on disciplinary identity development is presented, and later discussed in relation to the use of professional tools in IT education.

A. Active learning through making

With a focus on the maker movement in education, Blikstein [11] addresses both history and prospects. For more than a century, constructivist research has focused on student centred, interest driven and experimental approaches in education research, referring to how Papert outlined the importance of media and tools in learning, after working several years with a father of constructivism - Jean Piaget. The work of Piaget and Papert is the main building block of the maker movement in education, and Papert trusts that learning has better results if students are engaged in creation of public, shareable artefacts that are meaningful for them [11]. Through maker approaches in STEAM courses, students are making artefacts with integrated computation, as robots, e-textiles, programs and games. In the maker movement, process, product, and creativity go hand-in-hand. Making aims to engage students in their learning experience and make STEAM appealing to broader, diverse group of students [12].

Maker education is an example of active learning pedagogy. An active teaching approach aims to support deeper learning by increasing student motivation, engagement, and confidence. Active students are involved in doing things and reflecting upon them [13]. Active learning can involve project-based, problem-based, collaborative, game-based learning and making.

B. Taxonomies for learning

In 1956, Bloom [14] published a framework for categorizing educational objectives to help teachers to advance the learning practices they employ and enhance students' development in the cognitive domain. Bloom's handbook, with its significant impact in the pedagogical domain, was revised by Anderson and Krathwohl [15] in 2001. The old version of the framework had six categories: Knowledge, Comprehension, Application, Analysis, Synthesis, and Evaluation. Anderson et al [15] changed the categories from nouns to verbs, moreover various levels were renamed: Knowledge to Remembering, Comprehension to Understanding, and Synthesis to Creating. In the new version of the taxonomy, there are six categories belonging to cognitive dimensions including: Remembering, Understanding, Applying, Analyzing, Evaluating, and Creating. The taxonomy defines that understanding entails more complexity than remembering, that applying is more complex than understanding, and so on - the categories increase in complexity.

The Synthesis category was renamed to Creating, additionally upgraded and placed by authors as the highest in the ranking of cognitive complexity [15]. To create, students puts elements together to form a functional whole; students construct new products by mentally recognizing some elements or parts into a pattern or structure not clearly present before [15, p. 84]. When creating, students bring into play elements from many sources and combine them into an original structure related to their previous experiences.

The Create category is associated with three cognitive processes: Generating, Planning, and Producing. In the beginning, a student tries to understand a given task and come up with possible solutions. Secondly, he or she decides which solution to use and makes a plan of action. Afterwards, the student successfully executes their plan by constructing the solution [15].

Another educational taxonomy for the cognitive domain was created by Biggs and Collis in 1982 [16] and named the Structure of the Observed Learning Outcome (SOLO). Like Bloom's taxonomy, SOLO is a classification of learning outcomes based on their complexity. The SOLO model consists of five progressive levels of understanding: Pre-structural, Uni-structural, Multi-structural, Rational and Extended Abstract. The Extended Abstract level of students' performance is the most complex one, where they should have the competence to: design, create, reflect, hypothesize and generalize. Students are able to create new ideas based on mastery of the subject [16].

Both taxonomies place Creation as the highest level of learning outcomes; the process of acquiring knowledge in the most complex way.

In computer science education, Bloom's taxonomy is mainly used to state learning goals, as framework for assessment and to schedule instruction. This study will connect the taxonomies to progression of tool usage in IT education with project-based learning.

C. Previous work on tool categorization

Lukkarinen and Sorva [17] classified the tools of contextualized programming education. Their model presents many software and hardware tools that are available for teachers to assist programming courses using contextualization and media computation. The tool categories are based on whether the context induced with a tool is essentially based on software, hardware, or both. The essentially software-based solutions include e.g. microworlds, games, simulation, AI etc, while solutions with hardware elements include both electrical and non-electrical solutions. Finally, the combined category includes e.g. devices driven with a general-purpose computer and devices programmed with a general-purpose computer. In [18] the division on the categorization of programming tools are made from if the main goal is to learn programming or if the main goal is to use programming in pursuit of another goal.

Khalid et al. [19] did a categorization of students' web tools usage within problem-based and project-based learning, basing their categorization on previous research within the field. One approach was to map the tools with the learning activities, looking at the type of task being done, technique used and tools usage. Another approach was to look at the different phases (group forming, problem formulation, task formation, …) of the PBL project, the activities being done

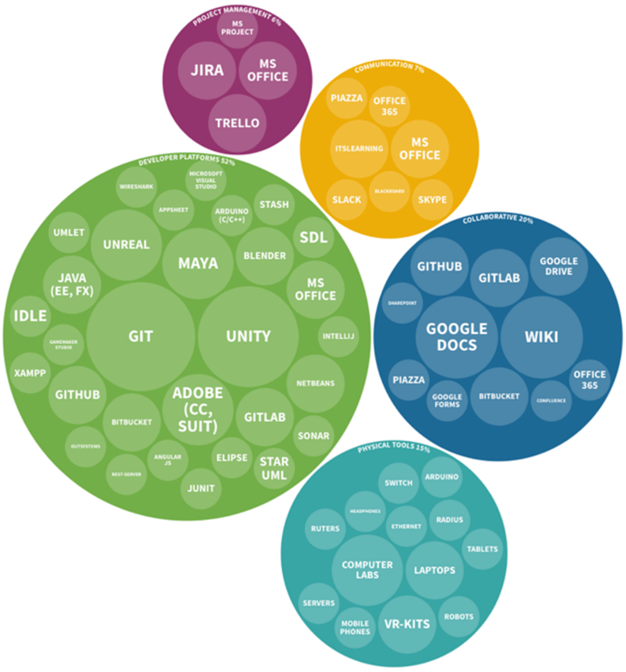

Fig. 1. Tools used in LtC courses, divided into five categories

(brainstorming, literature search, writing, presenting, …) and what tools are used. A third approach was to map common activities (sharing, discussing, communicating, …) with tools being used. Even though Khalid et al's [19] categorization is connected to PBL and PjBL, it is only focusing on web tools, not all digital tools used in project-based learning.

D. *Professional identity development*

Education can be seen not only as process of acquiring knowledge and skills but also as process of becoming [20] and "understanding who we are and in which communities of practice we belong and are accepted" [21, p. 644]. Barnett and Coates [22] proposed inclusion of identity development into curriculum and understanding of the curriculum as knowing, acting and being. Frezza et al. state that "it is essential that they [students], develop not just cognitive knowledge of something but also values related to that knowledge and the ability to express these values in professional action" [7, p. 1].

Conceptualizations of identity is many, in this article we decided to look at identity as a social construct [6] rather than a possession. In that view, identity is negotiated in interactions and constantly in the process of being constructed, negotiated, and reconstructed in everyday practices [23]. We construct identity to make sense of the cultural and social context and to be recognized as a legitimate member in that context [24]–[26].

What is recognized as "real IT" is a social construct that counts some practices as legitimate and others as not. Project-based learning "provides the means to develop their [students] identities as legitimate participants in their future academic communities" [27, p. 19]. Moreover, producing artefacts such as games and apps, which is often part of project work, has been identified as a valid way of participation in the computing discipline [28].

III. Methods

A. *Data collection*

In the project, we collected data that pinpoints 70 IT courses within 14 study programs at 2 Norwegian universities, Nord universities and Norwegian university of Science and Technology (NTNU), which use the LtC approach to different extents. This was achieved by contacting the coordinators of the study programs. In both semesters, we were unable to contact some of the university teachers, leading to only partial collection of data from existing LtC courses. In spring 2017 this amounted to 33 out of 37 existing LtC courses: during the fall semester of the same year, from 13 of 33. In total 46 courses.

The data was collected through 46 structured interviews of teachers of LtC courses, where the teachers, together with a researcher, filled out a questionnaire, including open-ended, multiple-choice and single-choice questions. During the interviews, which were mostly conducted using Skype for business and lasted for around 30 minutes, the researchers made notes on topics not represented in the questionnaire and let the teachers comment on these topics in the end of the interviews.

One of the questions of the questionnaire was about IT tools used in LtC courses, and these data are the basis of this analysis. This paper examines what kind of IT tools (software and hardware) are used in the Learning through Construction projects.

B. *Data analysis*

1) *Classification*

From the basis of pervious research, software tool providers' own definition of their tools, and experts' (teachers, and other researchers at our institution) opinion on the ways in which students use the tools, one researcher made an initial categorization of six categories. A second researcher validated the categorization afterwards (using a spreadsheet template). The initial categorization was based on the opportunities that the tool provides, resulting in placing the tools into several categories, for example Adobe CC was included in three categories: Collaboration, Developer Platforms and Information Processing. In the revised categorization, the researchers classified tools based on the real usage rather than possibilities that the tool provides, which resulted in defining Adobe CC within the Developer Platforms category only.

An inter-rater reliability test was used for reliability check, where two researchers reached an agreement level of 87%. The researchers discussed disagreements and decided to exclude the initial category Information Processing because of its ambiguity. The number of categories decreased from six to five in this process. The IT tools were then classified into the following categories: Collaborative tools, Communication tools, Developer Platforms, Physical Tools, and Project Management tools. Figure 1 presents the distribution of tools in each category whilst table 1 shows the number of tools used in each category. Some of the tools are included in two or more categories, since some of them may be used by students for different purposes. For example, Bitbucket is a developer platform for system control as well as a collaboration tool where users can share code with others.

Most of the disagreements on the categorization were related to a systematic difference of opinion. For instance,

Tool category	Number of tools
Collaborative tools	11
Communication tools	7
Developer platforms	30
Physical tools	13
Project management tools	4

TABLE I. CLASSIFICATION OF IT TOOLS USED DURING LTC COURSES

whether a web development course was to be considered a programming course or a computer science course (the researchers concluded the former). A small number of discrepancies were due to errors in the data gathering process, copy/paste errors, which were easily corrected in this process.

Some of the teachers were more precise when specifying types of IT tools and naming them, whilst others were more general, describing for example presentation and writing tools, text editors, pro tools for sounds, and a variety of freeware educational games, which made it harder to classify them into any of the categories.

2) *Progression as analytical lens*

A higher education study program usually consists of several courses, and it is an obvious goal to make sure that the students learn X before Y when necessary. In study program design there are introductory courses (sometimes without prerequisites), intermediate and advanced courses. However, there are many factors to consider in study program design, e.g. vision, traceability, progression and integration, manning the development project and anchoring decisions, pedagogy, mandatory vs. free choice of courses, implementation, evaluation and marketing [29]. Technology usage should also be consciously planned and designed throughout the study program. This study is using progression as an analytical lens to interpret the data from the LtC courses and will focus on progression of tool usage throughout a study program and in individual project-based courses. This qualitative analysis also includes generic data on student autonomy in the projects, also collected through the teachers' responses.

C. *Limitations*

The data used in this study was part of a larger study, where the focus was on many dimensions of project-based courses, not only tools. The question about the tools was an open-ended question: Which IT-tools (software and hardware) are used in the LtC-project? This is a limitation of the study as some teachers have listed a long list of software and hardware, while other teachers have mainly listed the specific tools needed for the projects. This means that for instance the hardware category is lacking tools like computers and servers, which in most cases are obvious and necessary to actually develop the artefacts of the student projects. Some of the teachers' answers are not specific, e.g. "flexible", "students choose themselves", "depends on the project", and therefore impossible to categorize. It should also be noted that there is no data regarding IT tools from one of 46 courses.

IV. FINDINGS

In IT courses, students create artefacts using technology. IT tools were used in the 45 LtC courses during the spring and fall semesters of 2017. The five categories will first be presented, then the findings concerning tool progression and student autonomy throughout semesters are described.

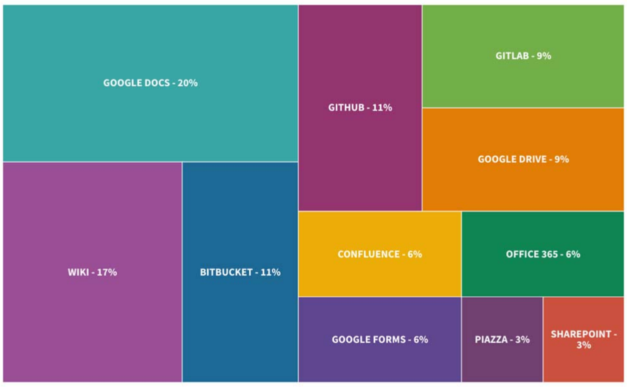

Fig. 2. Collaborative IT tools used in LtC courses

A. *Collaborative tools*

IT tools enhance creative collaborative activities in the context of education. The LtC concept often requires students to work in teams. According to data from 33 mapped courses in spring, the students have the opportunity to work individually in 11 courses; in the rest of the courses they worked in groups consisting of 2 to 10 students.

In this article, the term collaborative tools refer to tools that enable students, and in some cases teachers, to perform a variety of tasks such as sharing, storing and organizing files in addition to co-editing documents.

1 of 5 tools used in all the LtC courses was a collaborative tool. While collaborating, students used IT tools such as: Bitbucket, Confluence, GITLab, GITHub, Google Docs, Google Drive, Google Forms, Office 365, Piazza, Sharepoint, and WIKI. The most popular IT tool which enhanced collaboration in project-based learning processes was Google Docs followed by WIKI, GITHub and Bitbucket. Students used Google Docs in 7 courses (14%). Figure 2 shows to what extent each collaboration tool was used within the courses.

B. *Communication tools*

In this article, IT communication tools are defined as tools that allow students to communicate through short and long messages, or to participate in interactive discussions related to the project work. We also included learning management systems (LMS) into the Communication category, as they are mainly used for teacher communication (like distribution of learning resources), participating in virtual classrooms and informing about assignments, schedule etc.

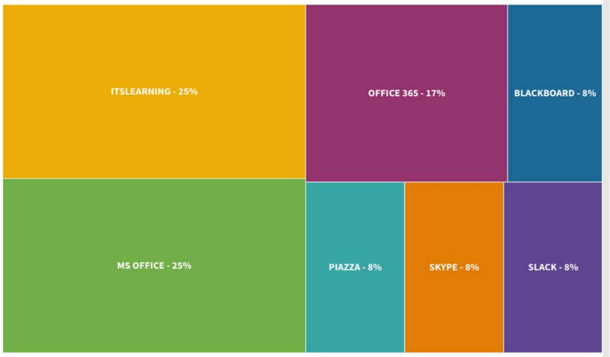

Fig. 3. Communication tools used in LtC courses

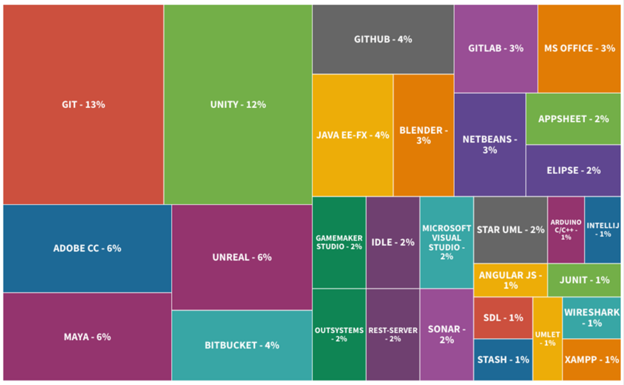

Fig. 4. Developer platforms used in LtC courses

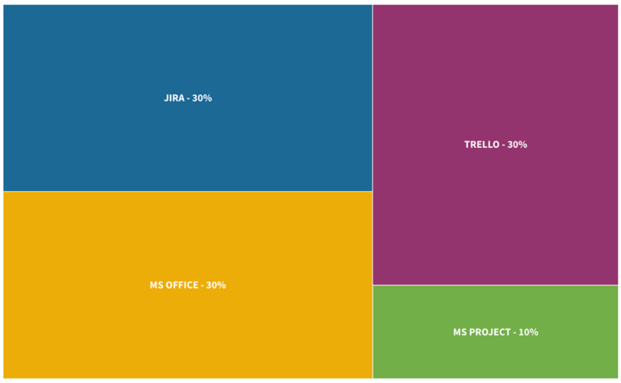

Fig. 6. Project management tools

Only 7% of all tools used in the courses are classified as Communication tools. The data gathered shows that students were using 7 services for communication. ItsLearning, MS Office, Office 365, Blackboard, Piazza, Skype and Slack were the tools that the teachers mentioned. Figure 3 illustrates the percentage distribution of each communication tool in the Communication category.

C. *Developer platforms*

More than half of the tools used by students in the mapped LtC courses belongs to the category of Developer Platforms. These are design tools used to create apps, artwork, games, software, interfaces, websites, videos, and other intangible IT artefacts.

Tools, which have been used relatively often by students to work on projects in LtC classroom settings are GIT, Unity, Adobe CC, Maya and Unreal. Figure 4 presents the percentage distribution of each developer platform software that has been used.

D. *Physical tools*

At both universities, students have access to spaces and tools that support group work and collaborative design. Physical tools used in LtC courses at university are presented in Figure 5. Examples of physical tools are VR-kits, robots and Arduinos, but also laptops, mobile phones and tablets are mentioned. Some of the teachers were precise and specified whether the physical tools used by students were university-owned or private, but not all of them did so. For this reason, the data presented do not distinguish between tools owned privately and by the university. It is quite possible that for all LtC courses, use of tools like laptops or stationary computers is necessary, and for that reason it seems so obvious that some teachers did not mention these tools. Figure 5 presents all the physical tools that teachers mentioned.

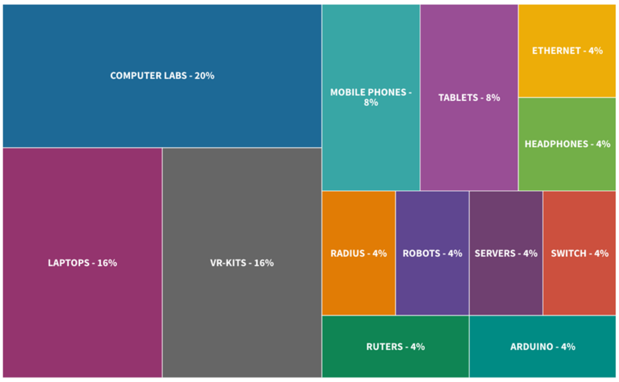

Fig. 5. Physical tools used in LtC courses

E. *Project Management tools*

Project management tools make it easier to organize projects; they help with visibility and control over tasks. Moreover, they enable their users to collaborate remotely and have constant access to all parts of the project. Not only are they often used for management, but also for collaboration and communication. In two LtC courses, students could choose what type of management tools they used, while other IT tools were specified. Students used the project management tools JIRA, MS Office, Trello, and MS Project. Their distribution in the category is presented in Figure 6.

F. *Tool progression throughout semesters*

In the first semester there were five LtC courses, in which students used 27 tools. The majority of the tools were development platforms. During the second semester courses (seven courses), the largest number of tools were in use; 43 tools. Among these, the developer platforms were most used.

The number of tools used decreased greatly in the third semester. Like in the first semester, there were five 3rd semester LtC courses, but in comparison the number of tools used decreased from 27 to 13 tools. In the ten courses during the fourth semester, the students used 40 different tools, again the majority of them in the Developer platforms category. During the fifth semester students used 18 tools, from which 13 were development engines. Interestingly, in the sixth semester in seven out of nine courses, the tools were project dependent or chosen by students themselves. There were no LtC courses mapped in the seventh semester. We see that in 8th, 9th and 10th semester, the number of tools used is lowest and they are often project dependent or chosen by students.

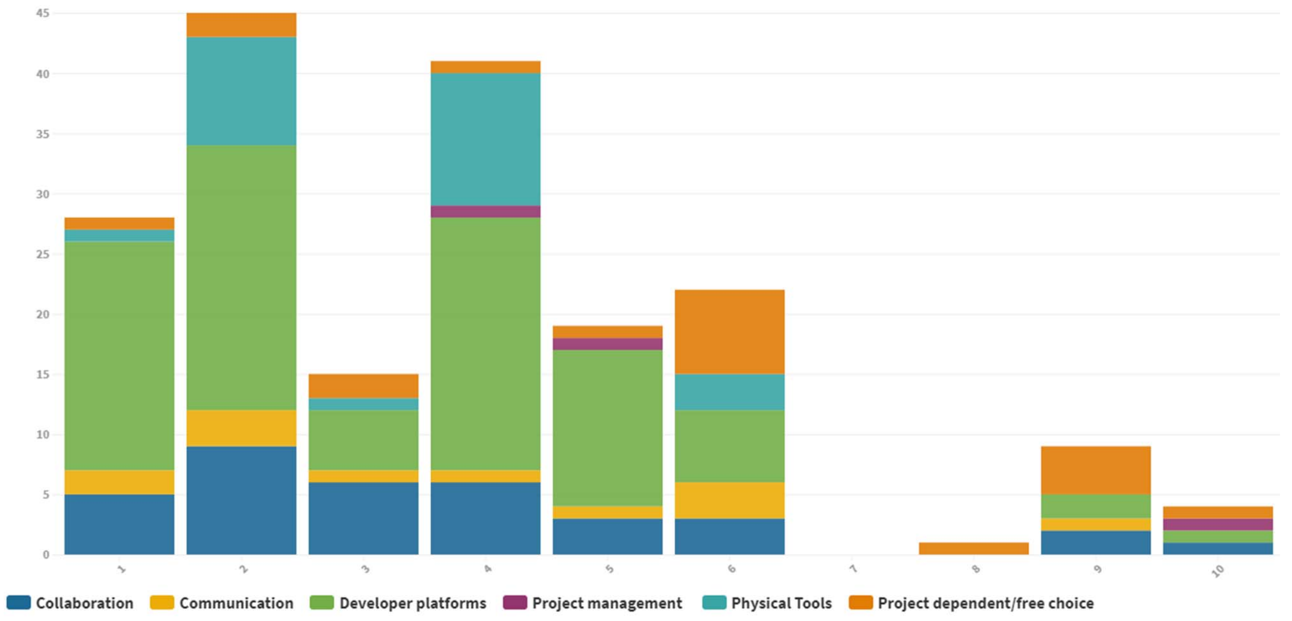

Fig. 7. Tool usage throughout semesters

Semester	# of courses	# of tools	Project dependent	Free choice some tools	Free choice
1	5	27	0	0	1
2	7	43	0	1	1
3	5	13	0	1	1
4	10	40	0	1	0
5	5	18	0	0	1
6	9	15	1	0	6
7	0	0	0	0	0
8	5	5	0	0	4
9	1	3	1	0	0
10	3	0	2	0	0

TABLE II. TOOLS USAGE TROUGHOUT SEMESTER

G. *Students' autonomy with regards to process and deliverables*

Previous work based on the same data, focused on students' autonomy with regards to both the process and the student deliverables in project-based courses. The students are given more freedom further into their education, as there are more unique projects and flexible project processes in the later study years. "Most first and second year courses have a well-specified or somewhat specified project process (concerning what methods to use, steps to follow, deadlines etc.), while during the last study years the project process is flexible (and where the only thing that matters is the final deliverable). However, there are examples of first year courses with unique projects per teams, but where the project process is still well-specified" [30, p. 9].

Project deliverables can range from very strict and well-defined, where instructors have specified in much detail what problem is to be solved and what should be delivered; to a more flexible one, where each team is completely free to decide what to develop as long as it relates to the learning goals [31]. The students' autonomy with regards to the deliverables also changes throughout the study years. "The products of the student projects are typically software prototypes and product documentation/software design, more specifically the main product types are games and web applications. During the two first study years, the project deliverables are mostly well defined by the teacher(s). Later in the study programs, there is a tendency that the students have more freedom concerning what to deliver" [30, p. 11].

H. *Freedom of tool choice*

Through qualitative analysis of the teachers' responses we found that students have a freedom of tool choice in some courses. Instead of listing which tools the students had to use, the teachers from 13 courses explained that the student had freedom to choose which tools to use in the project. In 3 courses, the students could freely choose some of the tools, but not all.

In the majority (16 out of 33) of the spring courses, teachers decided what kind of IT tools students should use. In 13 courses, students had a degree of freedom to choose tools according to their needs and preferences. In two cases, students were only given free choice of a project management tool, while other tools were chosen by the teacher. During the fall semester, four teachers out of 13 said that their students have had the freedom to choose any kind of tools when working on their project. Those that remain specified what kind of tools students should use.

In 10 courses, teachers gave complete freedom to use any kind of IT tools. One of the teachers said "All. There are no limits. Anything from a mobile [phone] to a VR kit and anything in between - anything that can be used to design, program and produce a digital game", another teacher responded, "Whatever the students feel will help them learn."

V. DISCUSSION

Based on the findings, we will focus the discussion on four main topics. First the tools used in PjBL is discussed with regards to Bloom's revised taxonomy and the SOLO taxonomy, then looking at professional tools and their role in the identity development of students. We then argue that it is important to focus on progression also when it comes to tool usage throughout study programs. These three topics are the basis of the guidelines of how to ensure progression of tool usage in study program design and the course design of project-based courses, and the discussion of how the guidelines can be used.

A. Creating as learning process

This study presents technology that aims to enhance learning at university courses with an LtC approach. A common goal for LtC courses is to reach the Create category of Bloom's revised taxonomy [15], as well as the Extended Abstract level in SOLO model by Biggs [16], which can be considered to correspond, since the Creating and Extended Abstract stages are both presented as the most complex levels of educational objectives in their taxonomy. The Create category from Bloom's taxonomy involves that the student put elements together to form a coherent or functional whole, through cognitive processes like Generating, Planning and Producing [15], while at the Extended Abstract stage of Biggs' framework, students should have the competence to create new ideas based on mastery of the subject [32]. Constructing a product during a LtC course is a complex process and requires that students use several cognitive processes; like creative thinking and deep understanding, and the categorization of tools used in LtC courses also shows that students need to collaborate, communicate, manage the project work and construct products through developer platforms.

The creative process in education can be broken down into three phases [15]: Problem Representation, Solution Planning, and Producing; IT tools can be used in each of them. In the problem representation phase, the student tries to understand the task and generate possible solutions. While trying to understand the task and generate various possible solutions, tools from categories like Collaboration, Communication and Project Management may be of use. In the solution planning phase, the student examines the possibilities and devises a workable plan. The designing of a plan to solve a problem is likely to be carried out by students through the use of Collaboration, Communication, Physical Tools and Project Management tools. In the last phase (Producing), in which groups construct artefacts, they will need tools from each category, but in particular Developer Platforms.

The freedom of choice of IT tools differed across the LtC courses. In 10 courses students had total freedom to choose technology, which allowed them to be involved not only in the creation of the deliverables, but also in the process of finding the best tools to achieve their goal. Previous work [30] identified that students' freedom gradually increases with time spent at university. When the outcome is itself an open-ended process, the product not being predetermined at all [32, p. 171], students work at the Extended Abstract end of the Design level.

B. Tools as professional building blocks

According to Vygotsky [33], social interaction involving cooperative or collaborative dialogue promotes cognitive development. Some of the LtC projects allow students to do more than simply interact with peers and teachers, but also to experience working with a customer. Their learning processes can be embedded in situations similar to work environments, thus preparing students for the work situation. For that reason, LtC projects can be placed somewhere in between the worlds of education and work. Most of the IT tools presented are not only used in academic settings, but in a work environment as well.

Students want to use the same tools that professionals use, here the matter of IT-identity comes in play. Students identify with the community of practice by having the same values, speak the same language, thinking the same way and using the same hardware and software. Tools are influencing students' identity development, and Kurnaz and Mahmood [34] aim to predict students' professional identity development by using the Professional Identity Five Factor Scale (PIFFS). One of the factors is knowledge of professional practices and it contains students' perception of what kind of applications, tools and equipment they will use in the future profession. The pedagogically useful tool presented by a teacher could be rejected because it is not the mainstream technology used by professionals in the IT field. Mark Guzdial [35] reflects upon his experiences with students' rejection of pedagogically-useful tools: "I wonder if this explains a phenomenon I've seen several times in CS education: teaching with a non-traditional but pedagogically-useful tool leads to rejection because it's not the authentic/accepted tool. I saw it as an issue of students being legitimate peripheral participants in a community of practice. Identity conflict offers a different explanation for why students (especially the most experienced) reject Scheme in CS1, or the use of IDE's other than Eclipse, or even CS teacher reaction when asked not to use the UNIX command line. It's a rejection of their identity."

C. Progression

In most study programs, students need to learn to use a variety of tools, and students often use additional tools in their learning process, e.g. to write text; to co-write; to calculate; to collaborate. The study program designers need to consider what tools are likely that students know, and what tools they expect the students to know from before. In some cases, it can be useful to communicate with the students to learn which tools they already know. The study program designers further need to consider in which order, and how much time to spend to teach students new tools.

Instead of focusing only on developer platforms, we ground our findings in empirical data on what tools students actually use in IT study programs and from the analysis where five tool categories emerged (see figure 1-6). The categories show that study program designers need to focus on a variety of tools with different focus and impact during the different phases throughout a study program. This is supported by Kolmos [36, p. 6], who emphasizes the importance of giving students tools for project management and collaboration, and also that students have specific needs when doing a comprehensive PBL course for the first time.

Cronholm [29] has defined 9 principles to consider for study program design. Progression is one of these principles, considered together with integration. Progression needs to be considered within several topics of a study program. In this article, the focus on IT tools in project-based courses narrows it down and makes it possible to develop some guidelines for educators, based on the different tool categories presented.

LtC course progression	Study program design – Ensuring progression of tool usage					Which tools are professional tools?
	Collaborative tools	*Communication tools*	*Developer platforms*	*Physical tools*	*Project management tools*	
Expected competence						
LtC course 1 1st semester						
LtC course 2 2nd semester						
LtC course 3 3rd semester						
LtC course 4 4th semester						
LtC course 5 5th semester	Freedom of choice – student autonomy					
LtC course 6 6th semester	Freedom of choice – student autonomy					
LtC phases	**LtC course design – Ensuring progression of tool usage**					
Problem representation						
Solution planning						
Producing						

TABLE III.
GUIDELINES TO ENSURE PROGRESSION OF TOOL USAGE IN STUDY PROGRAM DESIGN AND LTC COURSE DESIGN

D. *Guidelines to ensure progression of tool usage*

Based on the categorization of tool usage in project-based courses, in addition to the analysis focusing on creating as a learning process, progression and professional tools we have developed guidelines in the form of a table (see table III). The table should be used for planning tool usage throughout a study program and courses. The table emphasizes the different tool categories, but also ensures that the study program designers can discuss and reflect on what they expect students to know before arriving to the university and during the course of the studies. The guidelines also suggest that the students need to have freedom of choice when it comes to tool usage in project-based courses in the later part of the study program.

The guidelines (table III) also include a focus on professional tools. That does not mean that the study program designers only should choose professional tools, but that they should make decisions after reflecting upon if any of the chosen tools are professional tools, and how many of the chosen tools are professional tools.

The guidelines suggest that the course designers of project-based learning courses should consider which tools to use for the different phases; problem representation, solution planning and producing [15].

The goal of the guidelines is that study program designers and project course designers can quality assure the choice of tools based on progression and quality assure the choice of professional vs pedagogically useful tools. It is important that the LtC course design in seen in connection with the study program design, as the aim also concerning tool usage is to move from a course-based PBL approach to a systemic PBL approach [36].

The guidelines can also be the basis for discussing if the tools, which students need to learn, are part of the course content and the curriculum, e.g. if or how much of the course is spent teaching students to learn a tool, or if the students need to learn the tool on their own to use it as a tool to learn other parts of the curriculum.

VI. CONCLUSION

In this study, we have explored digital tools which aim to enhance learning in the Creating level of Bloom's taxonomy. Technology-enhanced learning in LtC courses includes a variety of tools; from developer tools for creating products, through project management, collaboration and communication to physical IT devices. LtC projects are not only about creating, but also collaborating, communicating, managing and using physical tools in the design processes.

Through an analysis focusing on categorization of tools and progression of tool usage in IT study programs and courses, we have suggested guidelines of how to ensure progression of tool usage in study program design and the course design of project-based courses. The guidelines also include a focus on the use of professional tools in IT education.

To get more insight into the usage of digital tools, there is a need for more precise data, since some of the teachers' responses were too vague to be useful for this analysis. Nevertheless, the data collected give a good overview of the variety of tools used during LtC courses, which enhances students' ability to successfully reach the Creating stage of Bloom's taxonomy.

ACKNOWLEDGMENT

This study is financed by Excited, a Norwegian Centre for Excellent Education.

We thank the reviewers for their deep engagement and constructive feedback, that we hope to incorporate more of in future work. We thank teachers who participated in the survey, and our colleagues for discussions and recommendation regarding tools categorization.

References

[1] E. Scanlon and J. Taylor, “Is technology enhanced learning an interdisciplinary activity?,” in *Proceedings of the 10th International Conference on Networked Learning 2016*, Lancaster, UK, May 2016, pp. 129–133, Accessed: Apr. 07, 2020. [Online]. Available: http://www.networkedlearningconference.org.uk/abstracts/pdf/P05.pdf.

[2] P. Peterson, E. Baker, and B. McGaw, *International Encyclopedia of Education*. Amsterdam: Elsevier, 2010.

[3] D. Laurillard, “Effective Use of Technology in Teaching and Learning in HE,” in *International Encyclopedia of Education (Third Edition)*, P. Peterson, E. Baker, and B. McGaw, Eds. Oxford: Elsevier, 2010, pp. 419–426.

[4] M. N. Giannakos, J. Krogstie, and T. Aalberg, “Video-based learning ecosystem to support active learning: application to an introductory computer science course,” *Smart Learning Environments*, vol. 3, no. 1, p. 11, Jul. 2016, doi: 10.1186/s40561-016-0036-0.

[5] S. Freeman *et al.*, “Active learning increases student performance in science, engineering, and mathematics,” *PNAS*, vol. 111, no. 23, pp. 8410–8415, Jun. 2014, doi: 10.1073/pnas.1319030111.

[6] E. Wenger, *Communities of practice : learning, meaning, and identity*. Cambridge: Cambridge University Press, 1998.

[7] S. Frezza, M. Daniels, and A. Wilkin, “Assessing Student’s IT Professional Values in a Global Project Setting,” *ACM Trans. Comput. Educ.*, vol. 19, no. 2, p. 9, Feb. 2019, doi: 10.1145/3231710.

[8] G. Sindre, M. Giannakos, B. R. Krogstie, R. Munkvold, and T. Aalberg, “Project-Based Learning in IT Education: Definitions and Qualities,” *Uniped*, vol. 41, no. 02, pp. 147–163, 2018.

[9] R. I. Munkvold, “Game lab: A Practical Learning Approach for Game Development,” in *Proceedings of the 11th Conference on Games Based Learning*, Graz, Austria, 2017, pp. 472–479.

[10] M. Guzdial, “Plain Talk on Computing Education,” *Commun. ACM*, vol. 58, no. 8, pp. 10–11, Jul. 2015, doi: 10.1145/2788449.

[11] P. Blikstein, “Maker Movement in Education: History and Prospects,” in *Handbook of Technology Education*, M. J. de Vries, Ed. Cham: Springer International Publishing, 2018, pp. 419–437.

[12] D. A. Fields, Y. Kafai, T. Nakajima, J. Goode, and J. Margolis, “Putting Making into High School Computer Science Classrooms: Promoting Equity in Teaching and Learning with Electronic Textiles in *Exploring Computer Science*,” *Equity & Excellence in Education*, vol. 51, no. 1, pp. 21–35, Jan. 2018, doi: 10.1080/10665684.2018.1436998.

[13] A. Misseyanni, M. D. Lytras, P. Papadopoulou, and C. Marouli, “Introduction,” in *Active Learning Strategies in Higher Education*, Emerald Publishing Limited, 2018, pp. 1–13.

[14] B. S. Bloom, *Taxonomy of educational objectives: The classification of educational goals, 1st ed.* Harlow, Essex, England: Longman Group, 1956.

[15] L. W. Anderson, D. R. Krathwohl, and B. S. Bloom, *A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom’s Taxonomy of Educational Objectives*. 2001.

[16] J. B. Biggs and K. F. Collis, *Evaluating the quality of learning: the SOLO taxonomy (structure of the observed learning outcome)*. New York: Academic Press, 1982.

[17] A. Lukkarinen and J. Sorva, “Classifying the Tools of Contextualized Programming Education and Forms of Media Computation,” in *Proceedings of the 16th Koli Calling International Conference on Computing Education Research*, New York, NY, USA, 2016, pp. 51–60, doi: 10.1145/2999541.2999551.

[18] C. Kelleher and R. Pausch, “Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers,” *ACM Comput. Surv.*, vol. 37, no. 2, pp. 83–137, Jun. 2005, doi: 10.1145/1089733.1089734.

[19] M. S. Khalid, N. Rongbutsri, and L. Buus, “Facilitating Adoption of Web Tools for Problem and Project Based Learning Activities,” in *Proceedings of the Eighth International Conference on Networked Learning 2012*, Apr. 2012, pp. 559–566, Accessed: Apr. 07, 2020. [Online]. Available: https://vbn.aau.dk/en/publications/facilitating-adoption-of-web-tools-for-problem-and-project-based-.

[20] G. Dall’Alba, “Learning Professional Ways of Being: Ambiguities of becoming,” *Educational Philosophy and Theory*, vol. 41, no. 1, pp. 34–45, Jan. 2009, doi: 10.1111/j.1469-5812.2008.00475.x.

[21] K. Handley, A. Sturdy, R. Fincham, and T. Clark, “Within and Beyond Communities of Practice: Making Sense of Learning Through Participation, Identity and Practice*,” *Journal of Management Studies*, vol. 43, no. 3, pp. 641–653, May 2006, doi: 10.1111/j.1467-6486.2006.00605.x.

[22] R. Barnett, K. Coate, and Society for Research into Higher Education, *Engaging the Curriculum in Higher Education*. Maidenhead, Berkshire, England: McGraw-Hill Education, 2005.

[23] L. L. Pozzer and P. A. Jackson, “Conceptualizing Identity in Science Education Research: Theoretical and Methodological Issues,” in *Sociocultural Studies and Implications for Science Education*, Springer, Dordrecht, 2015, pp. 213–230.

[24] K. L. Tonso, “Student Engineers and Engineer Identity: Campus Engineer Identities as Figured World,” *Cult.Scie.Edu.*, vol. 1, no. 2, pp. 273–307, Sep. 2006, doi: 10.1007/s11422-005-9009-2.

[25] H. T. Holmegaard, L. M. Madsen, and L. Ulriksen, “Where is the engineering I applied for? A longitudinal study of students’ transition into higher education engineering, and their considerations of staying or leaving,” *European Journal of Engineering Education*, vol. 41, no. 2, pp. 154–171, Mar. 2016, doi: 10.1080/03043797.2015.1056094.

[26] D. C. Holland, W. Lachicotte Jr, D. Skinner, and C. Cain, *Identity and agency in cultural worlds*. Harvard University Press, 2001.

[27] R. Elmslie and S. Lewis, “Student identity: transitions through project work,” *International Student Experience Journal*, vol. 4, no. 2, pp. 19–24, 2016.

[28] A.-K. Peters, “Learning computing at University: Participation and Identity. A Longitudinal Study,” Uppsala Universitet, Uppsala, 2017.

[29] S. Cronholm, “Nine Principles for Designing a Study Program,” *Information Systems Education Journal*, vol. 4, no. 31, 2006, Accessed: Apr. 07, 2020. [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:hb:diva-8246.

[30] L. Kolås, R. I. Munkvold, and S. A. Nygård, “Learning through construction in IT courses,” *NOKOBIT – Norsk konferanse for organisasjoners bruk av informasjonsteknologi*, 2017, Accessed: Dec. 18, 2018. [Online].

[31] G. Sindre, M. N. Giannakos, B. R. Krogstie, R. Munkvold, and T. Aalberg, “Project Thinking as a Strategy for University Education,” presented at the MNT-konferansen 2015, Mar. 2015, [Online]. Available: https://www.researchgate.net/publication/281062129_Project_Thinking_as_a_Strategy_for_University_Education.

[32] J. B. Biggs and C. S. Tang, *Teaching for quality learning at university: what the student does*, 4. ed. Maidenhead: McGraw-Hill, Society for Research into Higher Education & Open University Press, 2011.

[33] L. Vygotsky, “Interaction between learning and development,” *Readings on the development of children*, vol. 23, no. 3, pp. 34–41, 1978.

[34] S. Kurnaz and R. M. Mahmood, “Methodology Preview on Predicting Students Professional Identity Using Data Mining Techniques,” in *Proceedings of the Fourth International Conference on Engineering & MIS 2018 - ICEMIS ’18*, Istanbul, Turkey, 2018, pp. 1–6, doi: 10.1145/3234698.3234754.

[35] M. Guzdial, “Facts that conflict with identity can lead to rejection: Teaching outside the mainstream,” *Computing Education Research Blog*, Mar. 31, 2014. https://computinged.wordpress.com/2014/03/31/facts-that-conflict-with-identity-can-lead-to-rejection-trying-to-teach-with-different-tools/ (accessed Apr. 07, 2020).

[36] A. Kolmos, “PBL Curriculum Strategies: From Course Based PBL to a Systemic PBL Approach,” in *PBL in Engineering Education*, A. Guerra, R. Ulseth, and A. Kolmos, Eds. Rotterdam: SensePublishers, 2017, pp. 1–12.