Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

Sondre Stai

Teacher Collaboration in Online Learning

Analyzing an Asynchronous Learning Network in an Online Programming Course for Teachers

Master's thesis in Natural Science with Teacher Education
Supervisor: Monica Divitini, Majid Rouhani
January 2020

NTNU
Norwegian University of Science and Technology

Sondre Stai

Teacher Collaboration in Online Learning

Analyzing an Asynchronous Learning Network in an Online Programming Course for Teachers

Master's thesis in Natural Science with Teacher Education
Supervisor: Monica Divitini, Majid Rouhani
January 2020

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Norwegian University of
Science and Technology

Abstract

The inclusion of computer programming in school curricula has resulted in challenges related to continued education for in-service teachers who are working full-time. Several online courses are being conducted by Norwegian universities for teachers who are learning programming, including NTNU in Trondheim.

In this thesis, I have researched the use of the communication platform Slack in the online course IT6204 Applied Programming for Teachers to get an understanding of how communities of practice emerge in asynchronous learning networks and to identify a set of guidelines for course designers/lecturers of such courses.

The research is designed as a case study with three main sources of data: all messages from the "general" messaging channel used in the course, survey data from course students and interviews with learning assistants. When analyzing the data, I identified three main themes of how Slack is used in the course today: collaboration, reflection and course organization. The data was then used to synthesize a set of guidelines.

Sammendrag

Innføringen av programmering i skolen har medført til utfordringer knyttet til videreutdanning for utøvende lærere. Flere universiteter og høyskoler i Norge tilbyr nettbaserte kurs i programmering for lærere, inkludert NTNU.

I denne masteroppgaven har jeg forsket på bruken av kommunikasjonsplattformen Slack i det nettbaserte faget *IT6204 Anvendt Programmering for Lærere*. Hensikten med forskningen er å få innsikt i hvordan praksisfellesskap kommer til syne i asynkrone læringsnettverk og identifisere noen retningslinjer for fagansvarlige for slike fag.

Forskningen er utformet som et case-studie med tre hovedkilder av data: alle meldingene fra den felles diskusjonskanalen brukt i faget, svar på spørreundersøkelse for studentene og intervjuer med læringsassistenter i faget. I analysen identifiserte jeg tre hovedtemaer for hvordan Slack er brukt i faget i dag: samarbeid, refleksjon og organisering av faget. Dataen ble deretter brukt til å utforme et sett med retningslinjer.

Acknowledgements

Writing a Master's thesis can be frustrating work, but also incredibly rewarding. I have gotten new insights into the field of research, both from writing this thesis and from co-authoring a research article which got published at the CSERC'19 conference in Cyprus.

I would like to thank my dedicated supervisor, Monica Divitini, for excellent supervision and thorough feedback to every question I had. Also, many thanks to my co-supervisor Majid Rouhani, without whom I would not have gotten access to so much empirical data.

I would also like to thank my co-student Niklas Nystad, my journalist friend Morten S. Smedsrud and my mother Anita T. Stai for proof-reading and impactful feedback.

Table of Contents

List of Tables

List of Figures

Abbreviations

ALN - Asynchronous Learning Network

CoP - Comunity of Practice

NTNU – Norwegian University of Technology and Science

IDI – Department of Computer Science

STEM – Science, technology, engineering and mathematics

IT – Information technology

MOOC – Massive open online course

CSCW – Computer-supported collaborative work

Chapter 1

Introduction

"Learning is not an add-on to the role of the professional. It is a habitual activity where the group learns how to learn together continuously" (Hord, 2009, p. 40).

When building a city, it's important to have a thorough infrastructure to make different resources available to all its citizens even though they are geographically dislocated. Similarly, disjointed communities within the educational sector will result in some groups of people having a restricted access to relevant information and prevent holistic, continuous learning. We are dependent on functional professional learning communities to stay up to date with information, innovation and the possibility for sharing experiences with others working on similar problems as ourselves. Learning is both an active and a reflective process, traits which are cooperative by nature and requires interactions with other people.

Teachers have a special responsibility to stay current with new information and innovation, since school's societal mission is to prepare students for the future. In 2013, the Norwegian government appointed a committee to assess the lower school systems' curriculum compared with future demands of employment qualifications and social life. When summarizing their 130 page report, they wrote: "Changes in society and knowledge development are happening in an increasingly greater tempo.. []..it will become more important

in the future that learning in schools creates a foundation for the individual to acquire new skills throughout life" (Ludvigsen et al., 2014). Technology is connecting us broader and faster, and in a changing society it's almost impossible for schools to include all necessary knowledge in their curriculum. Hence, the ability to stay current and learn continuously is of grave importance for the next generations.

One of the greatest contemporary changes in school curriculum which requires new knowledge and continuous development by teachers, is the course renewal. The course renewal includes updated curriculums for all subjects in grade school, lower- and upper secondary school. For this thesis, I will focus on the changes in information technology, specifically the integration of programming in several subjects. The changes are being implemented in the period 2020-2023 and have led to a great need for programming courses for in-service teachers. But how are teachers supposed to learn effectively in online courses when they are working nearly full-time simultaneously? And how can these online courses remedy that the students are geographically disjointed?

1.1 Expected results

In this thesis, I will study the use of asynchronous messaging tools for an online course on programming for teachers and identify a set of guidelines for course designers to facilitate a community of practice in an asynchronous learning network (ALN). Both terms *community of practice* and *asynchronous learning network* will be explained in detail in the background section. In short, an asynchronous learning network focus on anytime collaboration rather than same time collaboration, and a community of practice is a community where the participants in that community share experiences and reflections to achieve collective learning.

1.2 Personal motivation

When I started researching which topic I should choose to study in my master's assignment, my first instinct was to look at the newest and "coolest" technologies for the classroom, specifically virtual reality and augmented reality, and how visualization and immersive experiences could impact learning. I am working part-time as an VR developer at a computer company here in Trondheim and have written about the subject in several courses at NTNU. When conducting interviews in relation to one of my course assignments on this topic, teacher feedback was mainly positive, but they all mentioned the need for special training and continued education on new computer systems as they were already overwhelmed by the rapid integration of new tools in Norwegian schools.

This research altered my thought process when approaching my master subject; instead of researching even more new technologies to be implemented in schools, it seemed like there were some systematic issues to be addressed. How are teachers trained in new technologies, and how can we create a sustainable body of knowledge for teachers to meet the challenges for rapidly changing needs for knowledge on technological subjects. Even though new technologies and their potential are exciting topics of research, they will never

reach the students unless there are systems in place to train teachers and support sustainable learning for educators.

In light of the implementation of programming in several school subjects, a political decision which was already in progress when I started this work, I concluded that it would be interesting to study how teachers are introduced to programming and how they can continue to stay up to date on this subject in the future. The course renewal is the biggest change in school curriculum in many years, but I would not be surprised if even bigger changes emerge in the future. This led me to the online programming course for teachers and the idea of studying its online learning community.

1.3 The case: continued programming education for teachers

Teachers are participating in programming courses as part of their continued education. One of these courses, IT6204 Applied Programming for Teachers, is an online course conducted by NTNU in Trondheim. The course is designed to be flexible both in terms of time management – since the students are in-service teachers – and course content. The teachers who are participating in the course are teachers in different subjects and teach at different levels. One of the tools used by the course designer to maintain flexibility and communication between geographically dislocated participants, is an online communication tool called Slack. Slack is mostly used for written messaging and is highly adopted by the information technology industry. During the course, the teachers were recommended to utilize the Slack communication platform to ask questions, answer each other's questions and share experiences throughout the course's duration. I have gotten access to all messages which were written in the "general" message channel used in the course – a channel where all course participants, learning assistants and lecturers could write, comment and read each other's messages. The messages were anonymized before I got access to them. In addition to Slack, the course lecturers used webinars and the learning man-

agement system Blackboard for communicating with the students, but Slack was the main source of interactive discussions and reflections. I will describe the case more thoroughly in a separate chapter.

1.4 Research inquiry and methodology

In this thesis, I wanted to go in-depth into a relatively small case and look at several different data sources to understand how a community of practice can emerge in an asynchronous environment among teachers who are learning programming. Hence, I chose to conduct a qualitative case study which enabled me to delve into my research topic and use my own experience as an informatics teacher when interpreting the data. I have three main sources of data: an export file of all messages from the general chat channel in Slack used for the course, answers from a post-course survey and interviews with learning assistants from the course in question.

RQ: How can Slack be utilized to facilitate a community of practice in the context of asynchronous online learning for teachers?

In addition to the main research question, I made two propositions to limit the scope of my reserach (which is explained more detailedly in chapter 4):

Proposition 1: What is the learning assistants' role in Slack for this course?
Proposition 2: What are the students' own thoughts on using Slack?

1.5 Life as a researcher

During the last semester, I've had the opportunity to conduct two interconnected works of research, both studying the same case: one of them is the masters assignment which is presented in this thesis, and the other is a research paper. I was invited by a couple of professors at NTNU, together with two PHD students, to participate in writing an article

for the 8th Computer Science Education Research Conference (CSERC '19). The paper, which is called "Programming for teachers: Reflections on the design of a course supporting flexible learning trajectories, was accepted and presented at the conference in Cyprus in November 2019. Working on this paper simultaneously as my own thesis, gave my insights into the lives of full-time researchers and the opportunity to collaborate with dedicated and professional people. I also provided me with an unique possibility for gathering data for my own research. The paper is to be published in the near future, but for now I have added the reviewed version in the Appendix.

Chapter 2

Background

In this section, I will give some background information on communities of practice and relevant theory for learning in online courses.

2.1 Asynchronous Learning Networks

An asynchronous learning network (ALN) are Internet-based learning platforms which focus on ”anytime”-communication rather than same time (Coppola, Hiltz, & Rotter, 2002). This model of collaboration supports continuous collaborative learning, while facing challenges of coordination. Most learners are familiar with the standard lecture format where students are presented with learning materials from a lecturer, but in an ALN the students must find supprt in their peers to a higher degree. Thus, course tutors must redefine their role as an educator from an active presenter to a facilitator of collaboration.

In contrary to same time collaboration, anytime collaboration gives the learners time to think and are not restricted by either geographical constraints or personal prejudice (Zion, Michalsky, & Mevarech, 2005). Zion et al. also states that ALNs contributes to reducing the number of passive students. Hence, it’s important that course designers of ALN-based

courses facilitate student interaction and coninuous, active collaboration between students.

2.2 Defining community of practices

Even though the term community of practice is relatively new, the concept is age-old. Wenger (Wenger, 2011) summarizes the term followingly: "Communities of practice are groups of people who share a concern or a passion for something they do and learn how to do it better as they interact regularly". The phenomenon is relatable to a variety of contexts; from tribes working together to survive, to a group of mathematicians who are trying to solve similar problems. Whenever people are interacting to share experiences regarding a common interest, a community of practice occurs. The concept comprises of three main elements: a domain, a community and a practice.

Musteen (Musteen, Curran, Arroteia, Ripollés, & Blesa, 2018) describes the community as a group of individuals, the domain as the common interest of the community, and practice as regular collective experimentation to increase the community's knowledge and experience of the domain.

The term community of practice was coined to increase our understanding of social learning, and as a framework for analyzing the relationships of individuals with their respective communities and how knowledge are transferred between such communities (Wenger, 2000). How do we know things which we have never researched ourselves? When we start breaking knowledge apart and look at the scientific models, terminologies and concepts which must be accepted and recognized to know something, it is clear that "knowing is an act of participation in complex learning systems" (Wenger, 2000).

Human interactions and how we feel a sense of belonging are not easily described, but we can define some broad categories for belonging to communities. Firstly, engagement; doing things together, creating, participating in shared experiences and interactions. Secondly, imagination; the self-perception of ourselves, our communities and how we fit in the

bigger image. A powerful tool for orienting ourselves in an intricate world of social communities. Thirdly, alignment: a way of interpreting our actions to how they are performed by others so that our local engagements are useful for others, e.g. following acknowledged scientific methods when conducting research (Wenger, 2000).

Gray describes a community of practice in the context of *informal learning*, and emphasize that the collective repertoire which is created through collective learning are used at a later point in time when the participants of the community face new challenges. The goal of such communities is to develop shared practices, and ways of solving problems using the experiences from collective learning (B. Gray, 2004).

2.3 Pedagogy of Digital Learning Environments

Learning environments consist of all factors which surround and influence learners; both physical artifacts such as equipment, access to fresh air etc. and social surroundings such as motivational factors, a sense of belonging and sustainable culture of learning (Skaalvik & Skaalvik, 2018). When John Hattie performed a meta-analysis of factors which contribute to learning outcomes, he found that relations between students and teachers is the most contributing element in a learning environment (Hattie, 2013). Digital learning environments suffer the disadvantage of lacking physical interactions among the participants, which is important to keep in mind when building a community of practice in a digital learning environment.

According to the vastly acknowledged theory of sociocultural constructivism, communication and interactions between students and teachers are vitally important for gradually building an understanding of new knowledge (Chen, Chen, Tsai, 2009). Lev Vygotsky, whose writings are credited as fundamental within theory of sociocultural learning, claims that the culture which surrounds the learner highly affects both what and how the student perceives new knowledge (Vygotsky, 1962). One of the core principals of sociocultural learning is the proximal learning zone, which describes the boundary between what stu-

dents can learn on their own and what could be learned in the right social and cultural setting. In simpler terms; what could be done and learned with assistance today, could be achieved alone tomorrow. Sociocultural learning is important for this article because it requires some type of scaffolding along the path of learning, which in my case is the use of asynchronous discussion tools.

To describe the use of sociocultural perspectives in the digital age, George Siemens created a "new learning theory" which describes learning as a networked activity in a network of nodes, where each learner represents a node (Siemens, 2014). It is highly disputed whether connectivism could be considered a new learning theory or not; a discussion which is out of scope for this thesis. E.g. Verhagen describes connectivism as a pedagogical view instead of a theory (Verhagen, 2006). Nevertheless, one of the key questions which prompted his work on connectivism was: "how can we continue to stay current in a rapidly evolving information ecology?"; a question which is assumedly relatable for teachers who are learning programming for the first time. Similar to communities of practice, connectivism describes learning as a social endeavor where people are connected to different communities. In connectivism, these communities are described as networks, and the individual as an interconnected node. Siemens highlights that one of the keys to maintaining up-to-date knowledge is the ability to see connections across networks of different fields and concepts, while maintaining the node's connection to these networks. The up-to-date knowledge is described as currency, and is to be assessed as the main goal for all connectivist learning activities (Siemens, 2014).

2.4 Implementation of Communication Tools in Online Courses

When implementing an online learning platform, there are three main relationships which must be facilitated (Konstantinou Epps, 2018):

1) Student – Content: How are learning materials made available to students?
2) Student – Facilitator: Interactions between students and lecturer and between students and teaching assistants, such as feedback, questions about course materials etc.
3) Student – Student: Facilitating communication between students to share experiences and ideas during the course, such as interpretation of assignments, reflections on relevance to own practice etc.

Typically, institutes make use of learning management systems (LMS) such as Itslearning and Fronter; in the case of NTNU, the current LMS is Blackboard Learn. In the case of IT6203 and IT6204, Slack was introduced to maintain two of these relationships: student – facilitator and student – student.

2.5 Awareness

Dourish and Bellotti wrote, "awareness of individual and group activities is critical to successful collaboration" (Dourish Bellotti, 1992). When people collaborate in the same physical place, we can effortlessly monitor each others activities. E.g. NTNU's learning management system Blackboard does offer a discussion forum for students, but there are no indications whether other students have read your messages or anyone else is online at the current point in time. As Tom Gross states; "overall it is important to note that awareness is a user's internal knowing and understanding of a situation including other users and the environment that is gained through subtle practices of capturing and interpreting information". Hence, when conducting an online course using an asynchronous discussion tool to scaffold the interactions between students and tutors, it's important to choose a tool which allows awareness of the other participants of the platform.

One commonly used feature to increase the individual's sense of awareness of others in digital communication platforms is outereactions. Nardi and Whittaker describes outereactions as "[]...a set of communicative processes outside of information exchange, in which people reach out to others in patently social ways to enable information exchange."

An example of an outereaction, is an indicator affordance, which tell you whether or not others on the same platform is online.

2.6 Online courses

Online courses have had a surge in popularity the last ten years, but we are still missing a consensus regarding the various pedagogy and learning strategies.

Nelimarkka and Vihavainen conducted a study spanning two years, where they analyzed message data from an asynchronous discussion channel. They were following the students of a single introductory programming MOOC over several terms, but the discussion channel remained the same one. Their observations suggest that communities of practices do naturally emerge in such chat rooms, and that previous students are unconditionally participating in the community even after they have finished the course themselves (Nelimarkka & Vihavainen, 2015); who they describe as alumni participants. The course which is studied in this thesis, IT6204 Applied Programming for Teachers, is not a MOOC, but experiences for how online discussion are conducted in MOOCs could be useful when designing online courses in general.

The importance of a relationship between student and teacher is well documented, e.g. John Hattie's meta-analysis of key factors for learning outcome, where he concluded that a good relationship with the teachers is the most influential factor (Hattie, 2013). Logically, online courses should be no different. For example, Picciani found that the activity of the course lecturer in discussion forums of online courses, increase the students perceived learning and increased the number of messages the students wrote themselves (Picciano, 1998). Increasing student activity is no easy task, and Picciani also stated that students in asynchronous environments must "understand their responsibility for being active contributors to instruction" and that "mature, motivated students are typically better suited for

asynchronous and distance learning". In addition to the relationship between student and teacher, interactions between students have been documented to have a profound effect on online learning (Swan, 2002).

2.7 Learning assistants – facilitators or interlocuter?

Learning assistants have an important role in online courses; how they interact with students and provide support to maintain students' motivation and learning outcome. According to Kizilcec and Halawa (2015), lack of adequate support in discussion forums has been identified as a key factor of low retention rates in online courses. Research also suggest that social motivation in terms of interacting with others can have both a positive and a negative impact on student motivation (Xiong et al., 2015), which implies the importance of assessing how communication in discussion forms are conducted. Ntourmas et al. (2019) studied the role of learning assistants in two MOOCs provided by the OpenEdX platform, and found that they "...did not promote problem-centered learning and collaboration, and they acted more as 'omniscient interlocutors' rather than as facilitators". Facilitation means supporting a group of people towards common objectives, which in the context of this study is the student community's learning goals of increased understanding of programming. Simply replying to direct messages and concrete questions about exercises is not enough to facilitate social interaction between students and build a community of practice.

Chapter 3

About IT6204 – Applied Programming for Teachers

There are six educational institutions in Norway which provide programming courses for teachers, all of which are online. NTNU is one of the institutions and conducts two main courses on programming for teachers: IT6203 Introductory programming for teachers and IT6204 Applied programming for teachers. The data collected for this thesis is in the context of the latter, IT6204. The course is conducted online using a variety of tools: webinars over Skype, Blackboard learning management system, external exercise delivery system and Slack for communication and collaboration.

The prerequisite of IT6204 is that students have a basic understanding of programming. During the course, the students are engaged in more elaborate activities like implementing logic controllers, creating computer games, programming for simulations (e.g. physics and mathematical modelling), and electronics using Arduino, Raspberyy Pi, Lego Mindstrom and Microbit. At the end of the term, each student creates an individual project which is related to their respective subjects as in-service teachers.

There are several challenges to address when designing a course for in-service teachers. Time and flexibility is an important aspect, as students are mostly busy with their daily duties as teachers during daytime, and depend on availability of learning assistants and course materials when they have time to spare for learning programming. Also important, is the variety of prior programming knowledge. Different school subjects have different needs in terms of programming, and their respective students vary in age and proficiency; not to mention that most teachers don't have any formal computing education.

To face these challenges, the course administrators implemented a learner-centered course design as described by Mark Guzdial (Guzdial, 2016). The essential idea is to respect the students and their nuanced needs and motivations for learning programming. To accommodate the different needs in skill level and subject specialization, the course used flexible learning trajectories where students could choose between several different learning paths to fit their individual needs.

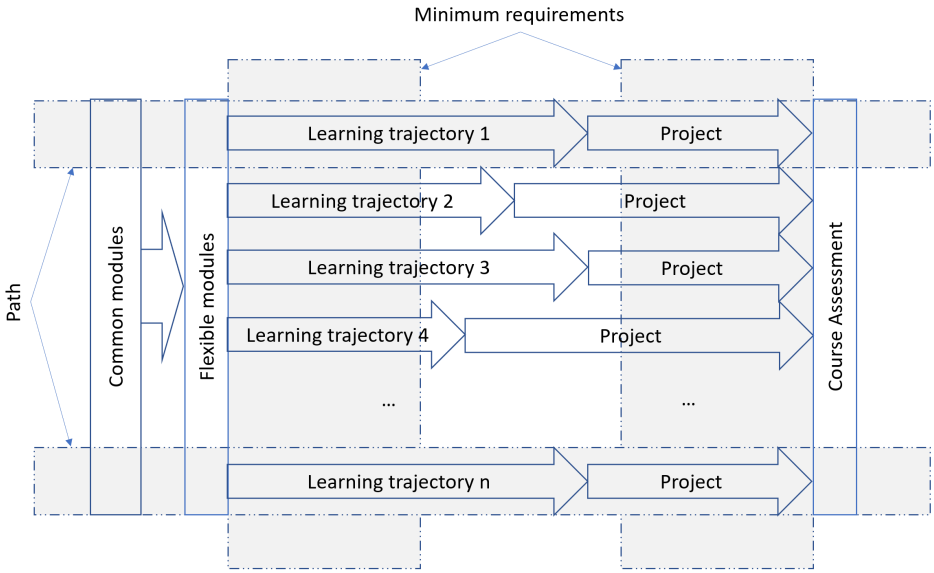


Figure 3.1: Example learning trajectories in IT6204

In his article "Dropout rates of massive open online courses: Behavioral patterns", Onah claims that while MOOCs are becoming increasingly popular, the completion rate of most MOOCs are below 13% (Onah, Sinclair, & Boyatt, 2014). In IT6204, the retention rate was approximately 91%, which vastly exceeds average MOOC retention rates. Albeit, we must keep in mind that most MOOCs don't have a targeted demographic and that the in-service teachers of IT6204 were mostly given extra free time from their workplace to engage in course activities. Nevertheless, IT6204 displays excellent results and one could easily argue that they are doing something right.

In addition to the flexible learning trajectories, the course uses the online communication platform Slack, which is the main focus of this thesis. Slack were used by both students, learning assistants and course lecturer to communicate and collaborate on exercises, discussing programming, asking questions about deliverables and more. In terms of community of practice: the community was the enrolled students, learning assistants and course lecturer, the domain was learning programming for in-service teachers and the practice was sharing experiences and partake in frequent collaborative discussions on the Slack forum (Wenger, 2011).

A more detailed description can be found in Appendix A, the research article about the course which I co-authored (Rouhani, Divitini, Vujosevic, Stai & Olstad, 2019).

Chapter 4

Research design

Robert K. Yin states that "...a research design is more than a work plan. The design's main purpose is to avoid the situation in which evidence does not address the research question" (Yin, 2018). The research design should state the case study's questions, it's propositions (if any), its case, the logic linking the data to the propositions and the criteria for interpreting the findings. Choosing the right methodology for a research project comprise of many factors: is the focus on breadth of depth? Are the statistical numbers or the underlying meaning of data most important? How much of the researcher's interpretations and existing knowledge is relevant? What is the intended end result?

4.1 Stepwise-deductive inductive method

To avoid bias and subjectively pre-existing ideas of how communities of practice emerge in asynchronous discussion forums, the iterative-deductive inductive method as described by Aksel Tjora (2017) is a solid choice when conducting qualitative research. The method is inductive because data is processed and analyzed without using leading concepts or theories of the field, such that new concepts and ideas can emerge. Then, when the inductive process is completed, the deductive process is formed by reversing the steps and check-

ing the data material again using the newly generated concepts. The method becomes iterative by including different types of data generation methods and sources of data material, which aligns with Oats' (2006) description of case studies: "The case is studied in depth, using a variety of data generation methods: interviewing, observation, document analysis, questionnaires". Yin also emphasizes that research is an iterative process where the researcher should base decisions depending on the process of various steps during the study (Yin, 2018). By using this method, I assure systematics and progression through the project duration, and it provided me with tangible goals along the way. Most importantly, the stepwise-deductive inductive method means that the researcher does not use any leading theory as a starting point, but gets the opportunity to create new theory based on the analyzed data (Tjora, 2017).

This study comprises of three main iterations:

1. Slack discussion data: I retrieved anonymized data in JSON format of all messages from the "general" Slack channel which was used during the course IT6204 Applied Programming for Teachers which was conducted in the spring of 2019.
2. Evaluation survey data: At the end of the course, the students where encouraged to answer a post-course evaluation of the course, were I contacted the course lecturer who kindly included my questions in the survey.
3. Interviews with learning assistants of the course: when data from step 1 and 2 were analyzed, I wanted to discuss the course with learning assistants who had first-hand experience with the discussion forums and could potentially supplement my findings with insights which were not apparent from the data.

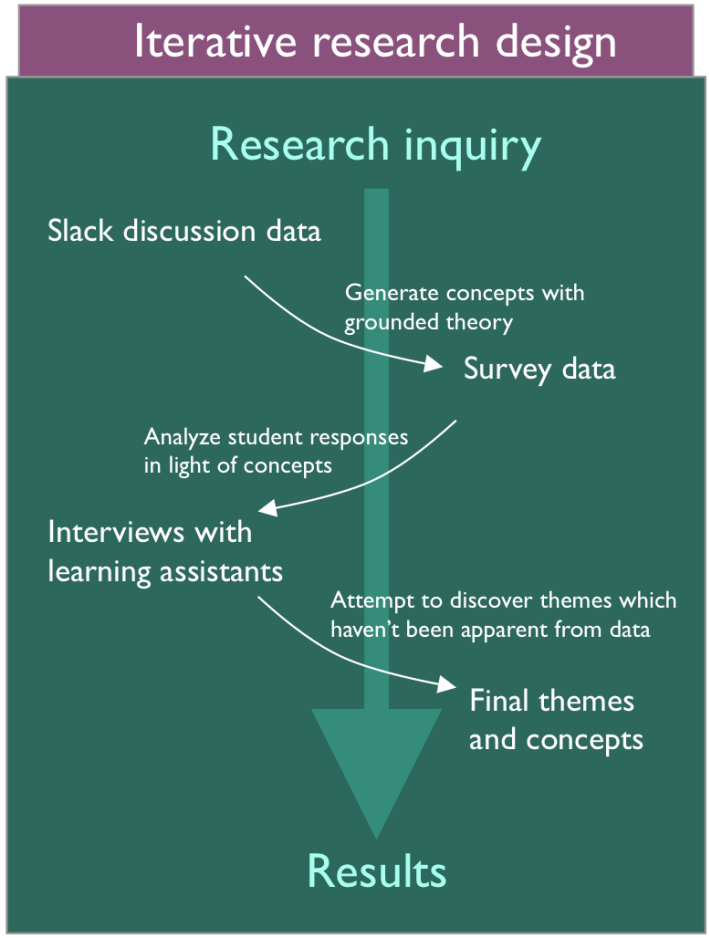


Figure 4.1: Illustration of the iterative research design based

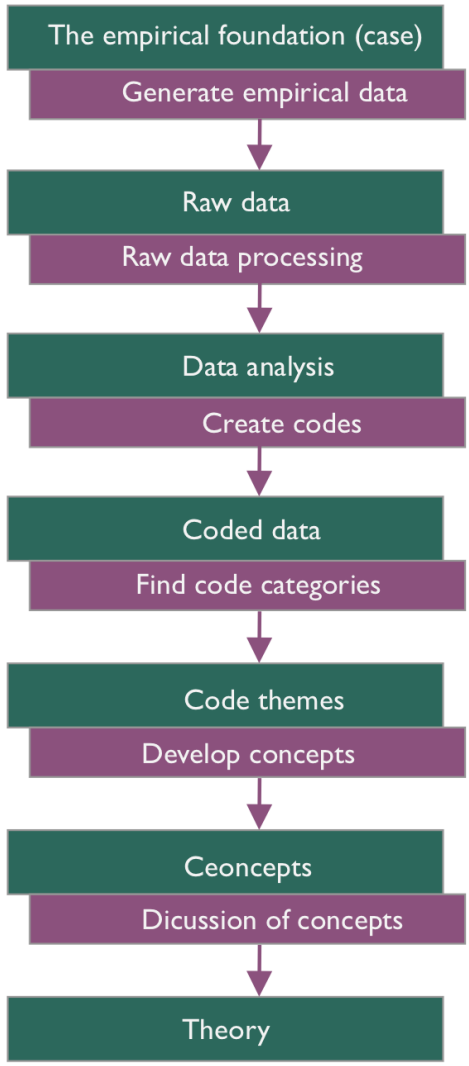


Figure 4.2: Based on Aksel Tjora's theory on qualitative research (Tjora, 2017)

4.2 Study questions

Case study research questions tend to be dominated by "how" and "why" questions, as the researcher intends to explore new areas and find new connections. In this thesis, I seek to investigate the use of collaboration technologies in the context of teaching programming to teachers, and how such technologies can support continued learning, sharing of experiences and building a community of practice; which leads to the following main research question:

RQ: How can Slack be utilized to facilitate a community of practice in the context of asynchronous online learning for teachers?

Some researchers perform a pilot study or some form of field work prior to a case study. In my case, the idea for this project arose when I contributed as a co-author to a research paper called "Programming for teachers: Reflections on the design of a course supporting flexible learning trajectories", which also studied the programming courses for teachers at NTNU, but from a different perspective. The paper was accepted as an experience report at the CSERC'19 conference in Cyprus in November. The work on this project gave me some insights into theoretical positioning of research papers and collaboration processes.

4.2.1 Propositions

To further narrow the scope of my research and to give my analysis some direction, I have supplemented the main research question with some propositions, as recommended by Yin (Yin, 2018).

Propositions:
- What is the learning assistants' role in Slack in this course?
- What are the students' own thoughts on using Slack?

In purely exploratory studies, it isn't necessary to have propositions in your research design, because it might entail to many restrictions for the study's intended purpose (Yin, 2018). In my case, I would argue that some restrictions are needed. The body of knowledge regarding online learning and communities of practice is vast. In addition, I am only analyzing how the community is initially developed and facilitated, not how the participants continue to use their shared practices after the course completion. Hence, the propositions are designed to narrow the focus of my analysis to the usage of Slack during the course, and learning assistants have an important role of facilitating this process as described by both Ntourmas et al. (2019) and Kizilcec and Halawa (2015).

4.3 Case Study Approach

When researching community of practice and student collaboration, I will argue that the qualitative approach provides more flexibility and leeway for in-depth analysis and interpretation of the data material than a quantitative one. In this thesis, I have analyzed all messages from a Slack channel contextual to a continued education course on programming for teachers and interviewed teaching assistants regarding their experiences with the collaboration tools in question. Briony Oates argues that a case study aim to "obtain a rich, detailed insight into the 'life' of that case and its complex relationships and processes" (Oates, 2006). By analyzing all messages sent in the general Slack channel of IT6204 during the a semester, and supplementing that analysis with interviews and surveys, I hoped to obtain insights into the community of practice of teachers learning programming and how the communication platforms were used in this setting. As Oates also states: "This one instance, or case, is studied in depth, using a variety of data generation methods", which is intended to enlighten the depth of a specific case rather than breadth. Studying collaboration in communication tools for teachers learning programming at NTNU, is quite a specific line of enquiry indeed, which convinced me that a case study would be the most appropriate design for my research.

Both surveys and experiments lead to a simplification of the real world and it's complexi-

ties, while a case study allows the researcher to look at the data in its context and take in to account multiple perspectives.

Chapter 5

Data Collection

5.1 Slack discussion data

The main source of data for this project was Slack messages posted by students in the course IT6204, which comprise of 550 messages in total. The messages were originally delivered to me in a unstructured cluster of JSON-files; here is an example message:


```
{
    "client_msg_id":
    "F39458B9-E172-4C0F-9C78-D1D5E5767D6B",
    "type": "message",
    "text": "Noen som kan anbefale hva jeg skal velge av
    Raspberry Pi, Arduino og Lego Mindstorm?
    "user": "UF6RTQ0AU",
    "ts": "1547029870.003900",
    "thread_ts": "1547029870.003900",
    "reply_count": 3,
    "reply_users_count": 2,
```

```
"latest_reply": "1547119577.001100",
"reply_users": [
    "UF5M2UXK2",
    "UF6RTQ0AU"
],
"replies": [
    {
        "user": "UF5M2UXK2",
        "ts": "1547035223.006000"
    },
    {
        "user": "UF6RTQ0AU",
        "ts": "1547101572.001700"
    },
    {
        "user": "UF5M2UXK2",
        "ts": "1547119577.001100"
    }]}
```


This method of exporting data was used in ensure anonymity amongst the students such that I could gain access to the messages without being invited to the Slack channel. I only got a Slack-generated user identification, e.g. UF6RTQ0AU, which is helpful to follow the same student throughout the course duration.

Before I could start reading the messages and doing my initial coding, I needed to convert the JSON format into a chronological and more tangible format for reading purposes. To achieve this conversion, I created the following script which prints out all the messages, the user ID of that message, and the associated replies in sorted chronological order.

```
for ( i=0; i<data.length;i++ ){
    var ts = data[i].ts;
    var alreadyUsedIDCounter = 0;
```

```
    for(j=0;j<usedIds.length;j++){
      if(ts == usedIds[j]){
        alreadyUsedIDCounter++;
      }
    }
    if(alreadyUsedIDCounter > 0){
        console.log(alreadyUsedIDCounter);continue;
    }
    usedIds.push(ts);
    $('#myTable tr:last').after('<tr><td>' + data[i].type
    + '</td><td>' + data[i].text + '</td><td>'
    + data[i].user + '</tr>');
    if(data[i].hasOwnProperty('replies')){
      var replyIds = data[i].replies;
      for(k=0;k<replyIds.length;k++){
        for(l=0;l<data.length;l++){
          if(replyIds[k].ts == data[l].ts){
            $('#myTable tr:last').after('<tr><td>' +
            '<img src="reply.png" />' + 'reply' +
            '</td><td>' +
            data[l].text + '</td><td>' + data[l].user + '</tr>');
            usedIds.push(data[l].ts);
          }
        }
      }
    }
  }
```

5.2 Survey data

At the end of the course duration, the students of IT6204 were given a course assessment survey, mainly conducted by the course supervisor to keep improving the course in the future. I contacted the course supervisor and asked if he would include some of my questions in the course assessment survey, which he was willing to do. With the survey data, I wanted to get a different perspective on the use of Slack then I had already gotten from the Slack data in accordance with the need for breadth in a case study as described by Oates (Oates, 2006). Also, insights from the students themselves would help me answer my second proposition as described in chapter 4. To get more context of the teachers' experiences with the use of asynchronous discussion tools, I composed the following questions:

1. Which digital communication- and collaboration tools (e.g. Slack, Itslearning) do you utilize in your daily working life, and what are they used for?

With this question, I wanted to gain some insights into the teachers past experience with the use of digital collaboration tools. By understanding some of their prior knowledge, I could hopefully gain relevant information for my discussion.

2. After utilizing Slack this last term, what are your thoughts on using similar tools in schools, both with colleagues and/or students?

Since the first question focused on the past, I composed the second question to focus on the future. Their eagerness/willingness to continue using similar tools in their own work could serve as insights into the successfulness of Slack's role during the course.

3. To your experience, are there enough available resources on the use of programming in classroom context?

Slack was a resource for discussion and collaboration to help the students learn programming more efficiently, but also interesting for their use of Slack is the availability of other

resources.

4. What could be done to make large transitions in schools easier for teachers, e.g. the course renewal?

With this last question, I wanted to see if any of the students would make connections between the community of practice that was used to learn programming during the course and continued learning for teachers in general.

Presumedly, adding too many questions to a non-mandatory course assessment survey would result in fewer responses, and since my questions were not directly related to the course assessment, I tried to write as few questions as possible.

5.3 Interviews

Rowley describes research interviewing as a craft (Rowley, 2012), and suggests that novice researchers should perform practice interviews to develop the necessary skills. I am indeed a novice interviewer, but got some practice into both conducting and analyzing interviews during a research methodology course at NTNU. Gray divides qualitative research interviews into five categories: structured, semi-structured, non-directive, focused and informal conversation (Gray, 2004). For this research, I chose to conduct semi-structured interviews, which allows the interviewer to probe for more details and ask follow-up questions to the interviewees' answers. I only interviewed learning assistants of the course IT6204, to gain a different perspective from analyzing the Slack messages and survey responses. Even though I gathered a lot of data from the Slack discussion tool, there were no opportunities to ask follow-up questions about the messages. Also, to get a better understanding of how the Slack channel was used, it seemed important to get the perspectives of staff members, not just the students. Prior to conducting the semi-structured interviews, I wrote a interview guide which could be found in Appendix C and which was approved by The Norwegian Centre for Research Data.

When conducting interviews, especially semi-structured and less structured interviews, the interviewer's bias has to be taken into account (Gray, 2004). For example, I tried to make the interviews be of equal length, as to not play favorites with the participants. When it comes to data security, it's important to use recording devices which is not in any way connected to the Internet, e.g. a phone. For my interviews, I borrowed a standalone recording device, and transferred the recording to NTNU's server, hence avoiding to store any raw data on my personal computer. Strict routines for data security are important for maintain anonymity of the interviewers. When transcribing the recordings, I avoided to have names of participants in the same document as the transcriptions, and simply coded the names with "person A" and "person B".

The interviews were designed to gain additional insights into how Slack was used to facilitate a CoP and help me answer my first proposition as described in chapter 4.

Chapter 6

Data analysis

In this research, I have collected three different types of data: Messages from the Slack discussion platform, student responses to the course's post-course assessment and transcriptions from interviews. Since the research was designed iteratively, the different data collections were analyzed at different points in time. The main source of data was the approximately sixty pages of discussion messages from Slack, which were used to gather knowledge about building a community of practice in asynchronous messaging platforms. The survey data was designed to get insights in the students' perspectives on using the Slack tool at the end of the course. The interviews gave me the opportunity to also gain insights from staff members, more specifically learning assistants from the course. The following subchapters describe the methodology which was used to analyze the data.

6.1 Grounded theory

Qualitative data consists of all non-numeric data from the research (Oates, 2006). To make sense of the data, and to be able to describe how one arrived at any conclusion, researchers must use acknowledged data analysis methods. Since this research was designed as an inductive case study, where the researcher analyzes the empirical data to see which theories

emerge, I chose to use the grounded theory analysis methodology.

Grounded theory was first introduced in 1967, and even though it is a strictly defined method, one should be aware of some of the underlying principles. Firstly, the research data are gathered from continuously changing conditions, not static ones. Secondly, grounded theory rejects both absolute determinism (that all events are completely decided from previous events) and nondeterminism (Corbin & Strauss, 1990). Grounded theory can be used on a variety of data sources, including interviews and written messages which are most relevant for this research. Also important is the view that data collection and data analysis are interrelated processes, and that every discovery must emerge several times through the data to be of relevance to the final conclusions (Corbin Strauss, 1990). This involves a constant comparative method, which means that previously analyzed data must be revisited when new concepts emerge later in the analysis process (Oates, 2006). In the coding process, I utilized a analysis program called NVivo 12. The methodology follows three main steps of coding: open coding, axial coding and selective coding, which will be described in the following subchapters.

6.1.1 Open coding

Open coding is the first step of the grounded theory analysis methodology, and involves the initial steps of finding relevant codes and categories from the data material. One should not look for concepts which can already be found in existing theory, but highlight the concepts which can be found in the material (Oates, 2006). I started with reading through all the material several times to get an overview of the data I had collected. Then, I went over the material again and highlighted words and sentences which were based on my own reflections regarding importance. In the next iteration, the different codes were put in categories. NVivo made it easier to maintain an overview of the coding, for example I could quickly sort out each message for a category to get a quick look at which codes had been added to that category already.

This following tables shows the different codes and their frequency in the data material,

along with a description of each code. Some codes, e.g. “discussing programming” were found in large numbers, but I chose to only select those which were somewhat unique since several students were asking similar questions.

6.1.2 Coding tables

Code	Description	Frequency
Arranging physical meetings	During the duration of the course, there were several seminars etc, and the participants used Slack to arrange physical meetings	7
Sharing learning materials	Discussion of where one can share one’s own learning material and where to find others	10
Discussing programming	Conversations and duscussions about programming	15
Assessment and course materials	Messages about the exercise system and curriculum	11
Equipment	Which equipment the students should acquire and their uses	9
Technical issues	Technical issues related to the different digital platforms used in the course	13
Relating to own practice	Messages where students relate course content to their own work as in-service teachers	25

Table 6.1: Open coding for Slack messages

Code	Description	Frequency
Learning assistant's role in the course	Regarding how the course is organized, mostly in terms of the learning assistants	9
Student messaging habits	How the students use Slack	5
Alternative platforms	Discussing alternatives to Slack and their own experiences with other platforms	7
Activity	When Slack was used and student's threshold to take part in discussions	6

Table 6.2: Open coding for interviews

6.1.3 Axial and selective coding

When the open coding was completed, I started working on the axial coding. According to Corbin and Strauss, the process of axial coding comprises of finding overarching themes for the discovered categories from the open coding (Corbin & Strauss, 1990). Making hierarchies connecting the categories, I found the following themes:

Figure 6.1: Themes from axial coding

After the axial coding comes the selective coding, where the researcher tries to find relationships and abstractions which relate the findings to the research inquiry and previous research (Corbin & Strauss, 1990). Even though the grounded theory methodology starts

out by 'ignoring' existing theories and focuses on the emerging concepts from the data, in the selective coding the researcher tries to make sense of the findings in light of relevant literature. My selective coding resulted in a set of guidelines for the course responsibles which are presented in the discussion section. In addition to the guidelines, I created a relational model of how the three themes are connected.

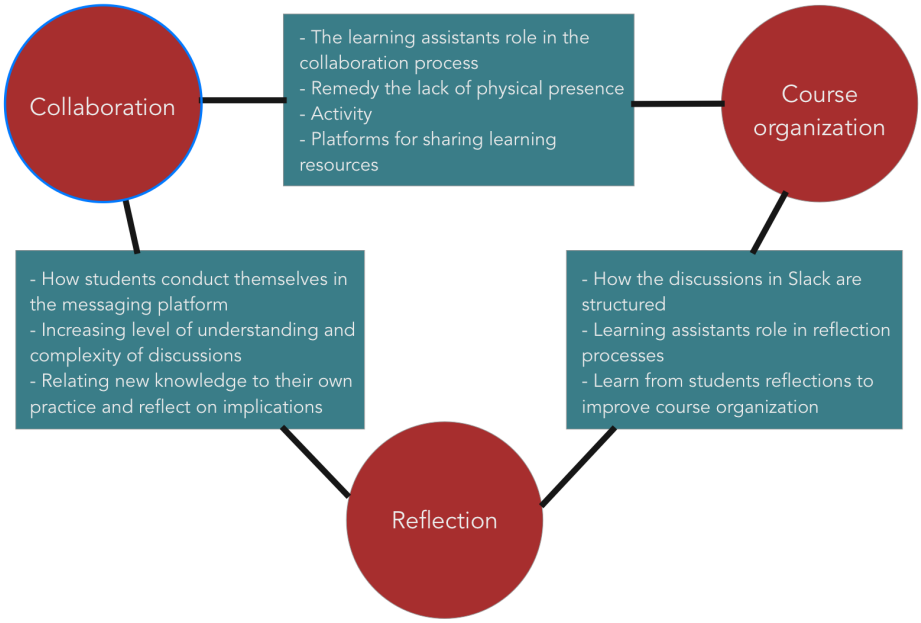


Figure 6.2: Relational model from selective coding

6.2 Validity and Reliability

Tjora defines *validity* as whether the answers which are found during the research process are directly related to the research question (Tjora, 2017). Golafshani (2003) states that validity also include the truthfulness of the research results. Validity can be divided into *communicative* and *pragmatic* validity, where the latter is mostly related to action studies. Researchers have established several traditions for maintaining communicative validity, e.g. presenting research at conferences and in scientific journals with peer review pro-

cesses. The results from my thesis have not gone through a peer review process yet, but part of my research is presented in an article which was accepted at The Computer Science Education Research Conference (Rouhani, Divitini, Vujosevic, Stai & Olstad, 2019). To increase validity, I added propositions to the research question which limited the research scope. There are factors about the community in the Slack channel which are not studied in this thesis, e.g. how the participants for the community use their shared knowledge after the course. Hence, limiting the scope makes the results less interpretative.

Reliability concerns reproducibility and whether the research instrument can be trusted (Yin, 2018). I have taken several steps to increase reliability of this study. Firstly, since reliability relates to consistancy over time (Golafshani, 2003), I made sure to analyze all of the 550 messages from the Slack discussion channel. To confirm that the data is representative for the student mass, I start the findings session by describing Slack's activity compared to the number of students taking the course. If e.g. only 10% of the students where using Slack, the results would be less reliable (and less interesting). Secondly, The interviewees were informed that they could request all data gathers from the interviews in case of any amendments or additional statements. Albeit, it is worth mentioning that grounded theory analysis could vary from researcher to researcher since it allows for using researcher's personal experience when conducting the coding process. Also, data from interviews and survey are dependent on the participants truthfulness.

Chapter 7

Findings

In this section, I will present the findings from my three different data gathering methods: analysis of message data, questionnaire and interviews as well as some general observations. All quotes from the discussion channel is translated from Norwegian by me.

7.1 General observations

There were in total only 50 unique user IDs in the general channel. In figure 7.1 you can see a graph representing number of messages per unique user ID.

Given that lecturers and learning assistants are included in this data, which could account for some of the larger values, the median is more likely to be representative for typical user activity than the average. The median number of messages for an active user amounts to 5 messages or replies. Even though the participation of writing messages was relatively low compared to number of students in the course, that doesn't mean the other students were completely inactive as shown in figure 7.2.

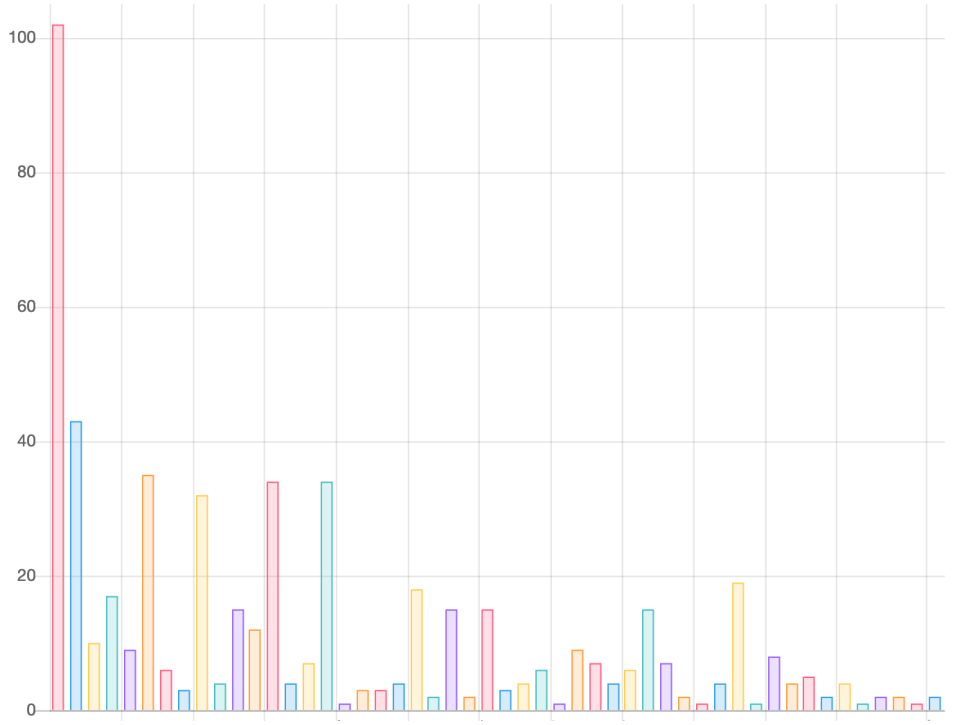


Figure 7.1: Diagram generated using graph.js, det code can be found in Appendix B. X-axis represente different users, and the y-axis shows number for messages per user.

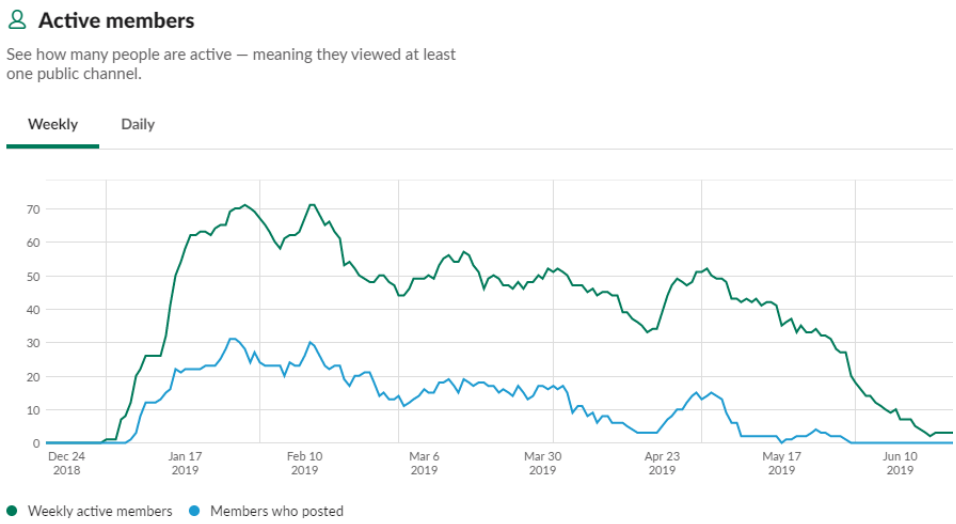


Figure 7.2: Diagram of student activity in Slack

As we can see from this auto-generated activity graph, most students were reading and observing the Slack conversations quite regularly.

The selective coding of my data analysis resulted with three overarching themes: Collaboration, course organization and reflection. In addition to the Slack messages from the course IT6204, I also got the results from two surveys related to the course IT6203. It's worth noting that many participants were engaged in both courses, so the surveys and Slack conversations are at least partially related in terms of who responded. For anonymity reasons, I could not compare the names myself, but I have gotten a confirmation from the course instructor.

7.2 Collaboration

Collaboration is a wide term which comprises of many different areas of interactions, co-operation etc. During my data analysis, I divided the collaboration-related messages into three main themes in accordance with number of occurrences: Discussing programming (The most frequent topic), arranging physical meetings and body of knowledge.

7.2.1 Arranging physical meetings

The key motivation for using Slack during this course was to enable students to interact with each other even though they were geographically disjointed. Nonetheless, one of the recurring topics of conversation was planning physical meetings and asking other participants regarding attendance to seminars and workshops. For example, one participant wrote:

> "Hi, I was allowed to attend a 'Lær Kidsa Koding' seminar in Trondheim on Monday the 11th of Mars. Will I meet any of my co-students there?"

Where some other student replied:

> "Yes, I am also signed up for that seminar."

A third student shimmed in:

> "I am also traveling to Trondheim the 11th of Mars, Looking forward to it! [smiley]"

And the first student concluded:

> "Great, I hope we can meet and exchange experiences."

So even though the Slack channel's original purpose was to replace the need for physical meetings, the students actually used the platform to plan physical meetings and let their peers know which seminars they would be attending.

Interestingly, students also used the general Slack channel to coordinate Skype meetings with lecturers, e.g.

> "Is it possible to get a Skype-conversation with any of the lecturers? I need a short conversation about the project assignment, and I think it would be easier than writing back and forth".

7.2.2 Body of Knowledge

In the "description of challenges" portion of this thesis, I described two main challenges which are relevant to my research enquiry; continuing professional development and developing a collaborative community of practice. In both cases, access to information and platforms for sharing experiences and reflections are relevant. A relevant question arises; does the Slack platform suffice or does it need to be supplemented or replaced by other software? In addition to the aforementioned request for voice chat, there were requests about where to share their projects with other students. One student said:

> "I have a question regarding the project assignment. I have a feeling there will be created plenty of interesting classroom teaching activities during this course. Is it possible for us to share these activities with each other when the course is finished? Is that something others would be interested in? "

Some other student suggested:

> "[].. Maybe we can share them here on Slack? I can't think of a better place to share"

It is possible to share files on the Slack platform, but is it the right platform for this purpose? This point will be revisited in the discussion section.

Another example from a student:

> "In assignment 5, we are supposed to create a game which can be used as a classroom teaching activity. I don't feel creative enough to generate new ideas for a game which can be used in mathematics and physics. Does anyone have any good ideas?"

In this example, the student is requesting ideas and inspiration for their own game project, also projecting a need for a platform to find other student's project to get inspired. One student felt so deprived of sharing opportunities that he/she chose to implement his/hers own platform:

> "My idea was to start with an empty web server and show how to build these types of learning resources such that the project assignment for this course ends up being the first step in the process of creating a new website with learning resources for electronics classes."

Due to the lack of platforms for sharing, some students decided to use the Slack channel to share guides with their co-students. E.g.

> "When it comes to Raspberry, I'm using the following when I'm programming. 1. I'm using OS version without graphics (only command prompt) 2. I install Samba on the Raspberry 3. From the PC, I connect to Samba as a network station which enables me to open program files on Raspberry from a text editor on the computer. I think Python works fine as an editor, also on C4. Compiling runs from command prompt by the help of Putty from windows or by SSH from Linux. I've not explained how to install the different elements, since their usage varies and because it's easily found on Google. In regards to programming, Gordon has a good library for dealing with ports on Raspberry."

Even though one could disagree with the statement that Python is an editor, the intention is clear: to share experiences with his/hers peers in form of a numerated guide. This guide is contextual to the course material and a different category of information sharing compared to the aforementioned classroom learning materials, but still in the theme of creating a body of knowledge.

7.2.3 Discussions on programming

As intended by the course lecturer, the Slack channel was frequently visited by students who wanted to discuss programming. In addition to learning assistants answers, the stu-

dents answered each others questions and collaborated on finding solutions to the assignments.

As an example, one student pondered:

> "I understand that I don't need to round off the time, but sometimes there are a lot of decimals when the speed is calculated. I'm just wondering if there is a way to limit number of decimals."

And got the reply:

> "Yes, you can use a function called Math.roundWithPrecision(a,b).
> For example Math.roundWithPrecision(2.5567,2) is rounded off to 2.56. See this screenshot: ..[file]"

The term screenshot occurred nine times throughout the general channel, and was a reoccurring method of sharing solutions to exercises. Some discussions were quite simple and could probably have been resolved through a search engine, e.g.

> "What is the difference between from pygame import* and import pygame?"

Which got the following reply:

> "Practically there are no differences between 'from pygame import' and 'import pygame', since both commands import the whole module. I liked the following explanation: [link]."

Other discussions were more complex and combined programming and their respective discipline:

> "You can't express V = blah blah or n = blah blah. T and P can be expressed in such a way, but to find V and n you must use the solution to Newton Raphson. That means finding better and better values for your desired outcome, generally:
>
> ```
> Xny = Xold f(Xold)/f'(Xold) f()=(P+(n^2a/V^2))(V-nb)-nRT
> f'()=P-an^2/V^2+2abn^3/V^3
> ```
>
> for volume v0 is initial value found by using equation of state, and you must choose desired accuracy, e.g. 0.0001"

And

> "I have some questions in regards to code/syntax when using numpy and other math libraries. Hope some of you can help [smiley]. This issue occurs frequently when I'm doing tasks on modelling and numerical methods. Firstly: We can create an empty list/table with N elements by using numpy.zeroes(N) and then assign values. How do I find the index of a certain value in such a list?...[]".

Where someone replied:

> "If I understand correctly, you get a list with indexes where a[index] ¡ 10 if you use np.where(a¡10). If you have the error code you get, it would be simpler to help you

When I analyzed the data, I tried to separate discussions on programming from general questions and answers, but it turned out to be difficult since simple questions could turn into long discussions with answers from several other Slack participants.

7.3 Course organization

The second overarching theme which emerged when coding the Slack messages was course organization. Students found the Slack platform to be a natural place to ask questions regarding exercise requirements, webinar recordings, what equipment to purchase etc. I have divided this theme into three categories: assessment and course material, equipment and technical issues.

7.3.1 Assessment and Course Material

This category was to be expected and does not comprise of many controversies; most questions of this nature are in relation to the different exercises and the final exam, but interestingly there were several students asking questions regarding Skype login information; messages regarding Skype also emerged in the "Collaboration" coding, though in the context of arranging extra conversations with lecturers or learning assistants. I coded these messages in this category because Skype was used for the course seminars, which is interpreted as an extension of course materials in this context. For example, one student asked:

> "What login name should I use for Skype for business? And login address?"

Some students were having issues related to getting hold of the course textbook:

> "Regarding curriculum: I have ordered the course textbook through the school I work at, but it seems like the delivery timespan is fairly long. How important is it to get hold of the book right away?"

And for some students, they clearly reflected on the issues of working full time while conducting their responsibilities as students:

> "[]... I haven't even ordered it [referring to the course textbook]. It's on my to do list when all the term grades have been calculated [smiley]. But I found that it's possible to get a free trial of the online version of the book, at least a starting point"

7.3.2 Equipment

In IT6204, the focus of the course was application of programming, and as a result the students needed micro controllers, wires and different types of electronic extensions to create their systems. The student group of IT6204 is fairly diverse, and many do not have a computer background beyond IT6203, where the focus is on fundamental principles of programming, not its application. Naturally, the Slack channel contained many questions on which equipment to buy. E.g.

> "Can someone recommend what to choose between Raspberry Pi, Arduino and Lego Mindstorm? I don't have any experience with electronics/programming except the course of last semester"

Also in regards to equipment, the students were collaborating in terms of planning to share equipment with peers within geographic proximity.

> "Argh! Was supposed to get Micro:bit from my school through the county ordering system by Atea. This was claimed prior to Christmas, and the order was sent today. Next week I'm in Hamburg with students. I have already checked with Vitenfabrikken and Kjellco, but not there either. Someone in Stavanger I can borrow from?"

And:

> "The school ordered Micro:bit on my behalf three weeks ago. It still hasn't arrived... I don't have the options of borrowing from nearby firms. I live in Hamar, anyone there who possess it?[referring to peers]"

In addition to parallels to the "planning physical meetings" category, I also found a connction to the "body of knowledge" category:

> "I have gotten started with Lego Mindstorm EV3, and have something called 'core set'. What kind of robot can be built using this, and where do I find instructions?"

As we saw with the example of a student who needed inspiration for a computer game,

once again a student is requesting inspiration and access to a body of knowledge. Both IKT I Praksis and Kidsakoder websites have descriptions of Lego Mindstorm projects.

7.3.3 Technical Issues

The course IT6204 make use of several communication platforms, and one of the reoccurring themes in the Slack messages was reporting on technical problems. These messages aren't highly relevant for my discussion, so I will briefly give some examples:

> "I can't see the videos on Blackboard. Have tried both Chrome, IE, Edge... The site becomes more grey, and I must close it again using a button at the bottom of the page to do anything. Do I have to look them up on youtube?"

As we will see when studying the course's retrospective questionnaire, students were generally not accepting of the Blackboard system and preferred using other available platforms.

> "I can't submit my video on Blackboard. When I click 'send', the button turns grey and nothing happens. "

And

> "The Blackboard app is unfortunately completely terrible. It displays an entirely different UI for the exercises than the website."

Several students also reported on issues related to Skype login:

"I can't login to Skype for the seminar. Get the error: $_{y}ouarenotconfiguredtoconnectwiththeserver.Contactthesu$

And

> "Will webinar 2 be posted on Blackboard? I am having troubles with my computer network, can't seem to login to Skype"

7.4 Reflections

This coding category relates to topics where students reflect on knowledge application in the context of their own practice.

As mentioned earlier, the students in the course are themselves educators within a wide range of disciplines, both standard subjects such as mathematics and languages, but also from practical disciplines such as electronics and physical machine operations. It was interesting to read about the latter's reflections regarding programming and their own community of practice. One teacher said:

> "We who are lecturing electronic courses and automation usually use equipment and technological solutions which are similar to those found in the industry and which complies with the criteria of durability and security from "Maskinforskriften" and NEK-EN 60204-1. Teachings in programming have basis in IEC 61131-3, and our school are using industry robots from ABB and a small number of CNC machines."

The same student also reflected:

> "For some of the electronic courses, programming will be integrated as part of a systematic whole. Programming will in practicality, and according to the course plan, be approximately the last five percent of the teachings, when working on completing an entire system."

Both statements describe issues related to the introduction on programming for a specific high school specialization. Some reflections regarding own practice also related to age difference instead of curriculum differences; while most teachers seemed to teach at either lower- or higher secondary school, e.g. one teacher wrote:

> "[]. . . does anyone have any recommendations for what learning material (LegoMindstorm etc) which is most applicable for someone working in primary school? Would like to direct my project towards such material"

And some other student reflected:

> "The way I interpret the book, the author [referring to Mark Guzdial] argues why one should teach programming in the different age groups, but I don't understand where a process would fit in that picture".

Some students were eager to know more about how the curriculum relates to the industry in which their own students will be working one day:

> "Is there a possibility to choose extra material and variations of the course material which is more similar to those in the automation industry?"

And

> "I called a firm and asked what Python is used for in the industry, and was told that

> Raspberry Pi have many key features for prototyping and such, that Python have a math library which is commonly used and that there are libraries for input and output"

One challenge which is not discussed thoroughly in the course renewal plan, is platform-specific technologies, which some teachers briefly commented on:

> "Hi. Does any of you use iPad to solve some of the assignments, or does anyone do programming on iPad? What apps are you using? Pros/cons? So far, I have only used a computer, but I work at an iPad School and realize that I have to familiarize myself with iPad usage for programming."

In the context of relating programming to own practice, several teachers reflected on which topics would be most suitable for implementation of programming exercises. One teacher said:

> "I can't imagine that I will be able to use programming when teaching quantum physics to my pupils, but electricity and technology in the physics curriculum seems appropriate for e.g. Arduino. I can also imagine using Python to draw graphs, but GeoGebra works fine for this purpose already. Since the new curriculums are not yet decided, it's difficult to predict what would be applicable later on ..[]"

And another teacher chimed in on the discussion:

> "Yes, programming is useful for numerical solution of differential equations, but at the same time they [referring to dif.eq.] are being removed from R2 [advanced mathematics in higher secondary school], but they might re-emerge in Physics 2. I would have loved to focus on something I know for sure will be part of the curriculum after the renewal".

This dilemma is also difficult for course lecturers who are teaching programming for teachers, because they have the same difficulties of predicting the final curriculum after course renewals next year. At the same time, some teachers were eager to explain solutions they had successfully implemented in classroom context:

> "I have created a learning activity for Physics 1, where the pupils are gathering data using Tracker and draw position- speed- and acceleration graphs in Python for the movement. The activity has already been used in two classes, graphs on PowerPoint were used to have interesting conversations about the difference between numerical and analytical derivation, and how number of data points affected the outcome. The entire learning activity therefore comprised of, simply put, showing graphs on PowerPoint, programming together with the students regarding some predetermined movement, and then get the pupils to do it themselves with their own data".

7.5 Students' feedback

When the IT6204 course was approaching its end, the course organizers distributed an evaluation survey to get feedback from the students regarding possible improvements and general remarks on the course. Before the survey was distributed, I contacted the course lecturer and asked to get a couple of my own questions included in the survey. Out of the 80 students who enrolled in the course, 22 answered the evaluation survey. One of the questions was closed and used the Likert scale for answer options (Oates, 2006), while the other three questions were open questions.

Question 1 (Likert scale): I have used Slack during this course and felt it was useful.
Answers:

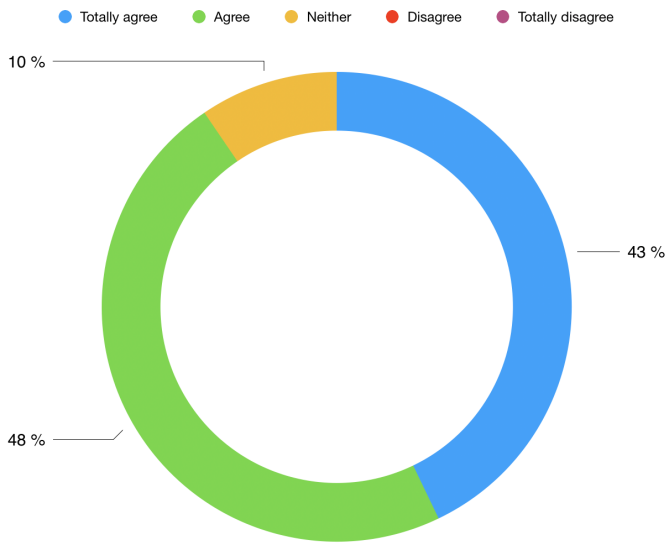


Figure 7.3: Diagram of student responses to evaluation question 1

Question 2: Which digital communication- and cooperation tools (e.g. Slack, Itslearning) do you use on a daily basis at work?

Question 3: After using Slack this term, what are your thoughts about utilizing similar tools in schools, both for collaborating with colleagues and communicating with pupils?

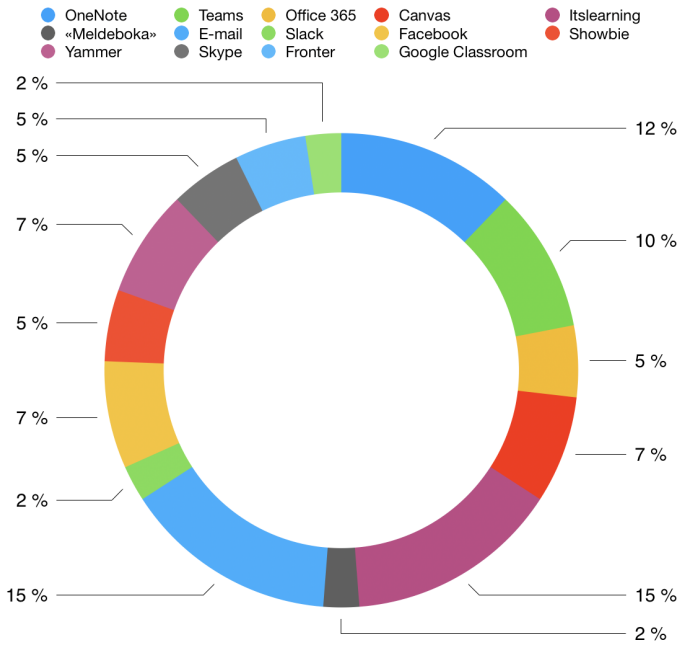


Figure 7.4: Collaboration tools used by teachers in their daily work life

12 of the respondents expressed positive statements towards implementing Slack in schools, but their motivation regarding field of application varied. Some respondents argued that they were positive, but only if it replaced some of the current platforms, e.g.:

> "I think it could work fine. But as a replacement, not an addition to the rest. The new communication- and learning platforms we have in our county work poorly, especially when it comes to direct communication with students."

While some other respondent even mentioned the idea of using Slack as a replacement for blogs:

> "Absolutely worth thinking about. I'm going to teach "Technology and Research" this fall, could try it out there. Previously, we have used blogs in that course."

7 respondents were negative to the utilization of tools such as Slack in schools, but their arguments were almost solely in terms of not wanting yet another system to deal with. Several of the negative answers pointed out that it would be unnecessary in addition to the Teams platform. E.g:

> "I'm thinking it would be a little too much to have a tool as Slack in addition to Teams. Especially for those who are insecure about using digital tools."

And:

> "Okay, but we can't have too many tools on different platforms. E-mail, schoollink, teams and potentially Slack will be many to deal with."

Question 4: Do you feel like there is enough available resources which explain how programming could be used in classroom context?

It might seem strange to have text answers to a yes/no question, but my reasoning was that I would get more replies if the respondents could choose between a simple "yes" and some elaboration. Indeed the most recurring answer was "yes" (6 instances), but most answers contained some elaboration to their reasoning. E.g.:

> "There are many good resources out there when it comes to specific learning activities and projects teachers can give their students, but maybe less resources in regard to didactics of programming. I also wish for more examples of how student works and learning can be evaluated in a good way".

And:

> "Yes, you need to search and put in an effort, but it's fully possible to find information. Especially if you are willing to use English websites and resources."

Relying on English resources was a reoccurring theme:

> "Yes, I find what I need online, but I want more internet resources / good explanation- and inspiration videos in Norwegian to make it easier for my students."

There were some ambiguity in the answers, for example, one respondent wrote:

> "We have gotten access to many good advices and materials for future learning"

While another respondent wrote:

> "I don't feel like I have gotten to know such resources in this course".

Which could be seen in relation to another answer:

> "Yes, if one could only find them! It was nice with the link collection which is located on Blackboard somewhere".

One respondent also pointed out the challenges in relation to their respective field:

> "There are many good learning activities for use in the classroom, but it could be difficult to find material which is related to a specific subject".

Question 5: What could have been done differently to make it easier for teachers in large processes of change, e.g. the course renewal?

Several respondents pointed out the need for enough time to learn:

> "Most importantly, we need working hours to work with this change. We only have 24 hours a day. We are not lacking ability to change, but willingness to use all of our spare time to learn new things."

And

> "To get enough time to work properly with the subject. When the days are too busy, this doesn't become a priority".

One teacher also pointed to the need for collective learning:

> "More collective education in the workspace. Doesn't need to be with examination, but e.g. guest lecturers, workshops.."

And another one wrote about teachers' motivation on the subject:

> "Many teachers have no interest in programming, which means the interest won't show in the classrooms either. A separate course on programming could probably work better."

Chapter 8

Discussion

In this section, I will discuss the findings and make sense of their implications and synthesize a set of guidelines for course designers of online courses for teachers.

8.1 The Community

During the analysis, I identified three main themes of how the Slack channel was used during the course: collaboration, reflection and course organization. As described by Wenger (2011), a community of practice comprise of a group of individuals who share a domain and conducts a practice by regular interactions with the other participants. The Slack channel was used regularly during the entire course duration, but the community can be described to have more than one domain. On one hand, you can look at the participants as in-service teachers who needs to learning programming and how to relate programming to their respective courses as described by the course renewal. On the other hand, they are also students who are trying to pass a course, which was mandatory for many of them.

A shared practice did emerge during the course, and the students were specifically asking the other students about programming related to their own practice as in-service teachers.

This was especially apparent during the projects period, where the students made learning activities for schools in relation to their own practice. They were interested in each others projects and one student even made it his course project to create a website for sharing the classroom activities created by the other students.

Even though the community shared a domain in regards to learning programming for use in classroom activities, they also have individual problems related to using programming for their respective subjects.

8.2 Learning assistants role

When the students were asked whether they found Slack useful or not, the response was overwhelmingly positive (see figure 7). Even though the analysis of student activity showed that only about half of the students did post a message of their own, most students were actively following the discussions (see figure 6). We can also see a gradual decrease in activity, both for students who were posting messages and for students who were observing. In the interviews with learning assistants, I asked about their thoughts on activity on Slack, and they pointed out that students gradually went from asking questions in the public channel to sending private messages to the learning assistants. As observed by Ntourmas (Ntourmas et al., 2019), learning assistants have an important role in facilitating and promoting social interaction between the participants of a digital community, and Coppola, Hiltz and Rotter (2002) have similar conclusions regarding asynchronous learning networks. The learning assistants in the course also stated that they were not given any kind of guidelines or training in how to conduct themselves in Slack, and one can easily imagine that the students prefer just getting the right answer from asking a learning assistant directly than spending time discussing problems with their less experienced peers. However, I would argue that this model of interaction between learning assistants and students is not in accordance with sociocultural learning theory or useful for building lasting learning communities among the students. As stated by Musteen, a community of practice requires regular collective experimentation to increase the community's knowledge

(Musteen et al., 2018). When the discussions are gradually replaced by messaging certain individuals, we are no longer talking about collective learning, building a community nor contributing to an ALN with continuous interactions. This leads me to two guidelines for the course designer:

- Have a seminar with learning assistants at the beginning of the course where you discuss how learning assistants and lecturers should conduct themselves in the asynchronous chat, as to facilitate collaboration instead of simply answering students' questions individually
- Also, encourage students to use the shared discussion forum instead of asking learning assistants individually

Let's imagine one of the students are back to working full-time as a teacher and is preparing some learning materials using programming, and that some problem arises. Now that the course is ended, the teacher can't just send a private message to their previous learning assistant and expect to get any help for free! For teachers to stay current in the paradigms of technology and programming, I believe they need to build communities among themselves and share knowledge along the way. As we can see from figure 5, one of the users had a dominant presence in the Slack forum with over 100 messages, compared to the median. From the contents of the messages, it seemed like this user was either a learning assistant or a lecturer. It is worth considering for the course designers whether such activity from a non-student will have a positive or negative effect on the general discussions.

8.3 Collaboration

There were numerous interesting discussions in the Slack messaging channel where students shared problems and concerns and found solutions together. However, when I identified number of messages for a unique user, the median number of messages was only 5. In a way, this is a very good result: students were very happy about the use of Slack and found it useful, but the activity level suggests that there is still room for improvement. When I asked the learning assistants how we could increase activity in the Slack channel, they sug-

gested making the discussions a mandatory part of some exercises. As Picciani pointed out, student activity in asynchronous learning environments tend to favor mature and motivated students who take responsibility of instructional messaging (Picciano, 1998). An important factor to consider regarding the students motivation in this course, is that most student are obliged to take it. I have not gathered motivation-specific data in this research, but it's a perspective to consider for the course designer.

The last exercise in the course is a project where students are creating their own learning material for use in classroom contexts. From my findings I identified a theme which I called "body of knowledge". By this, I mean platforms and technologies which enables teachers to share learning materials with each other, and several of the students requested such a platform.

Slack is not intended as a platform for large-scale sharing of resources. Obviously, you can share files which other participants can download, but you can't create structures of learning resources with e.g. searchable terms. I contacted the course lecturer regarding this dilemma, and he explained that the course participants were recommended to use either:
- https://iktipraksis.iktsenteret.no/laeringsopplegg
- https://kidsakoder.no/

Kidsakoder contains many great guides on creating programming projects with numerous different technologies, e.g. Scratch, Micro:Bit etc, but it's seemingly not possible to publish your own learning activities without contacting the administrators directly. "IKT I Praksis" on the other hand, offers a web-based interface for publishing learning activities in the context of teaching computing. From the Slack data, some students were seemingly not aware of these resources. Which leads me to the next suggested guideline for the course designer:

- Clearly specify how and where students can share learning materials with each other,

both for course collaboration and as a body of knowledge for use in schools after the course.

8.4 Reflections

As mentioned in the findings section, one teacher reflected on the different learning platforms used in schools, and stated that their schools are using iPads for teaching programming. I have previously, while working as a research assistant at NTNU, contacted The Norwegian Directorate for Education and Training (Udir) regarding the exact same dilemma. Some school have iPads for all their students, while others have Chromebooks; the classical personal computer seems to be vanishing from Norwegian schools. Udir replied to my enquiry with *"[]... whether there will be any guidelines for specific platform is not yet decided... []"*, which implies that teachers must individually find solutions for their school's chosen platform. I believe this is a good perspective for designers of programming courses for teachers; most teachers are dependent on using online tools instead of downloadable software for specific platforms.

In Mark Guzdial's book "Learner-Centered Design of Computing Education: Research on Computing for Everyone" – which was part of the curriculum for IT6204 – he discusses situated learning in the context of community of practice and describes how new learners of a community often are eager to absorb the knowledge which is actually used in a professional context (Guzdial, 2016). Interestingly, his observations do align with some of the Slack messages e.g. the student who had called businesses and asked about which programming language was most useful in the industry today. This points to an interesting perspective on the course curriculum of IT6204; while the course is aimed at giving teachers knowledge for use in classroom context, the teachers are also interested in obtaining industry-relevant knowledge.

Guzdial also points to the fact that one is supposed to make a selection of which learning community to join before one is proficient in the community's domain. It's impossible

to know whether you like being an expert in programming before you have tried it, but you can know if you identify yourself with the participants of a community. As Dewey also points out, it's important that knowledge in schools does not become a self-contained entity, but that students can contextualize new knowledge and relate it to their own experiences from the real world (Dewey, 1902). Hence, teachers also need to contextualize these concepts before presenting them in schools, which leads me to a new guideline for the course designer:

- *Contextualize the course curriculum and help the teachers/students to understand the needs from the industry related to their respective fields.*

From the retrospective evaluation survey of the course, a couple of students suggested that the course needs more didactics; how these concepts are presented in schools in relation to learning goals and school curriculum. To meet the needs of all the different subjects which the students are teaching across a wide age range, it would mean a significant increase in workload for course lecturers to facilitate such course contents and points to the importance of a functioning community of practice.

8.5 Summary of guidelines

Complete list of guidelines presented in the discussion:

- Clearly specify how and where students can share learning materials with each other, both for course collaboration and as a body of knowledge for use in schools after the course.
- Have a seminar with learning assistants at the beginning of the course where you discuss how learning assistants and lecturers should conduct themselves in the asynchronous chat, as to facilitate collaboration instead of simply answering students' questions individually
- Also, encourage students to use the shared discussion forum instead of asking learning assistants individually

- Consider having a separate channel for administrative questions, as they take up a large part of the discussion forum
- Consider including the writing of discussion messages in the forum as part of one or more exercises
- Contextualize the course curriculum and help the teachers/students to understand the needs from the industry related to their respective fields.

Chapter 9

Closing words

9.1 Future research

This thesis have been restricted to activity in an asynchronous discussion tool when analyzing a community of teachers who are learning programming. But what happens to the community after the course has ended? Nelimarkka and Vihavainen (2015) did a study where they used the same discussion channel when conducting an online course over several course periods and found that previous students voluntarily continued to participate in discussions. Albeit, this was for a MOOC, so it would be interesting to see if the same would happen in a closed online course.

In terms of a community of practice, it would also be interesting to do research on how the participants of the community use their knowledge outside of the discussion forum. In my findings, I found that several teachers were uncertain about where to share learning resources, both at the beginning of the course and at the end. Does community-building in asynchronous environments stimulate the participants to continue to keep in touch and share experiences? Since the students in the course are in-service teachers, one can easily imagine the benefits of sharing learning activities regarding programming, sharing experiences from these activities and conduct collective reflections.

9.2 Conclusion

My findings show that the asynchronous learning network was popular among students, and resulted in a community which discussed programming and reflected on programming in terms of their own practice as in-service teachers. To further facilitate teacher/student collaboration in this course, I have identified a set of guidelines for the course designer. The results points to the importance of how lecturers and learning assistants conduct themselves in the discussion; when they more frequently helped students individually, the activity in Slack declined, which was also confirmed in the interviews with the learning assistants from the course.

Conducting the research presented in this thesis has been a privilege, an influential learning experience and a lot of work. Even though the course renewal represents a huge change in school curricula and demands for continuous learning for teachers, I suspect we will see bigger changes in the future. I believe communities of practice and asynchronous learning networks will be important to scaffold these processes now and in the future.

Bibliography

[1] Chen, Y., Chen, N.-S., & Tsai, C.-C. (2009). The use of online synchronous discussion for web-based professional development for teachers. *Computers & Education, 53, 1155–1166.*. https://doi.org/10.1016/j.compedu.2009.05.026

[2] Coppola, N. W., Hiltz, S. R., Rotter, N. G. (2002). Becoming a virtual professor: Pedagogical roles and asynchronous learning networks. *Journal of Management Information Systems, 18*. (4), 169–189. https://doi.org/10.1080/07421222.2002.11045703

[3] Corbin, J., Strauss, A. (1990). *Grounded Theory Research: Procedures, Canons, and Evaluative Criteria. Qualitative Sociology*. (Vol. 13). Retrieved from https://link.springer.com/content/pdf/10.1007

[4] Dewey, J. (1902). *The Child and the Curriculum.*. Chicago: University of Chicago Press.

[5] Dourish, P., & Bellotti, V. (1992). Awareness in Collaborative Writing Systems. *Cscw,*. 107–114. Retrieved from http://dourish.com/publications/1992/cscw92-awareness.pdf

[6] Golafshani, N. (University of T. (2003). Understanding Reliability and Validity in Qualitative Research. *The Qualitative Report, 8*. (4), 597–606.

[7] Gray, B. (2004). Informal Learning in an Online Community of Practice. *Journal of Distance Education, 19*. (1), 20–35.

[8] Gray, D. E. (2004). *Doing Research in the Real World.* . SAGE Publications.

[9] Guzdial, M. (2016). *Learner-Centered Design of Computing Education.*. Morgan & Claypool.

[10] Hattie, J. (2013). *Synlig læring - for lærere.* . Cappelen Damm AS.

[11] Hord, S. M. (2009). Professional learning communities. *Journal of Staff Development, 30*. (1), 40–43. https://doi.org/10.4324/9781315164564-3

[12] Kizilcec, R. F., & Halawa, S. (2015). *Attrition and achievement gaps in online learning.*. L@S 2015 - 2nd ACM Conference on Learning at Scale, 57–66. https://doi.org/10.1145/2724660.2724680

[13] Konstantinou, G., & Epps, J. (2018). Facilitating online casual interactions and creating a community of learning in a first-year electrical engineering course. *Proceedings of 2017 IEEE International Conference on Teaching, Assessment and Learning for Engineering, TALE 2017, 2018-Janua*. (December), 128–133. https://doi.org/10.1109/TALE.2017.8252317

[14] Ludvigsen, S., Elverhøi, P., Ishaq, B., Rasmussen, J., Sundberg, D., Gundersen, E., … Rose, S. (2014). *Elevenes læring i fremtidens skole.* . Oslo. Retrieved from https://www.regjeringen.no/contentassets/e22a715fa374474581a8c58288edc161/no/pdfs/nou201420140007000dddpdfs.pdf

[15] Musteen, M., Curran, R., Arroteia, N., Ripollés, M., & Blesa, A. (2018). A Community of Practice Approach to Teaching International Entrepreneurship. *Administrative Sciences, 8*. (4), 56. https://doi.org/10.3390/admsci8040056

[16] Nelimarkka, M., & Vihavainen, A. (2015). Alumni Tenured participants in MOOCs: Analysis of two years of MOOC discussion channel activity. *L@S 2015 - 2nd ACM Conference on Learning at Scale*. 85–93. https://doi.org/10.1145/2724660.2724671

[17] Ntourmas, A., Avouris, N., Daskalaki, S., Dimitriadis, Y. (2019). Teaching Assistants in MOOCs Forums: Omnipresent Interlocutors or Knowledge Facilitators. *Transforming Learning with Meaningful Technologies, 11722.* . https://doi.org/10.1007/978-3-030-29736-7

[18] Oates, B. J. (2006). *Researching Information Systems and Computing. Inorganic Chemistry*. (Vol. 37). https://doi.org/10.1016/j.ijinfomgt.2006.07.009

[19] Onah, D. F. ., Sinclair, J., Boyatt. (2014). DROPOUT RATES OF MASSIVE OPEN ONLINE COURSES: BEHAVIOURAL PATTERNS MOOC Dropout and Completion: Existing Evaluations. *Proceedings of the 6th International Conference on Education and New Learning Technologies (EDULEARN14),*. 1–10.

[20] Picciano, A. G. (1998). Developing an Asynchronous Course Model at a Large, Urban University. *JALN,* . 2(1), 1–14. https://doi.org/10.24059/olj.v2i1.1926

[21] Rouhani, Majid; Divitini, Monica; Vujosevic, Vojislav; Stai, Sondre; Olstad, Hege Annette. (2019) *Design of a programming course for teachers supporting flexible learning trajectories.*. The 8th Computer Science Education Research Conference (CSERC '19) . Open University; Larnaca, Cyprus. 2019-11-18 - 2019-11-20.

[22] Rowley, J. (2012). Conducting research interviews. *Management Research Review, 35*. (3–4), 260–271. https://doi.org/10.1108/01409171211210154

[23] Siemens, G. (2014). Connectivism: A Learning Theory for the Digital Age. Retrieved from http://er.dut.ac.za/handle/123456789/69

[24] Skaalvik, E. M., Skaalvik, S. (2018). *Skolen som læringsarena*. (3rd ed.). Universitetsforlaget.

[25] Swan, K. (2002). Building Learning Communities in Online Courses: the importance of interaction. *Education, Communication and Infromation,* . 2(1), 23–49. https://doi.org/10.1080/1463631022000005016

[26] Tjora, A. (2017). *Kvalitative forskningsmetoder* . 3rd ed.). Oslo: Gyldendal Norsk Forlag AS.

[27] Verhagen, P. (2006). Connectivism, A New Learning Theory? https://doi.org/10.15405/epsbs.2017.05.02.41

[28] Vygotsky, L. (1962). *Thought and language. Behavioural Neurology*. (Vol. 5). https://doi.org/10.3233/BEN-1992-5106

[29] Wenger, E. (2000). Communities of Practice and Social Learning Systems. *Organization, 7*. 2), 225–246.

[30] Wenger, E. (2011). Community of Practice: a Brief Introduction. https://doi.org/10.2277/0521663636

[31] Xiong, Y., Li, H., Kornhaber, M. L., Suen, H. K., Pursel, B., & Goins, D. D. (2015). Examining the Relations among Student Motivation, Engagement, and Retention in a MOOC: A Structural Equation Modeling Approach. *Global Education Review, 2*. (3), 23–33. Retrieved from http://ger.mercy.edu/index.php/ger/article/view/124

[32] Yin, R. K. (2018). *Case Study Research and Applications: Design and Methods* . Los Angeles: SAGE Publications.

[33] Zion, M., Michalsky, T., Mevarech, Z. R. (2005). The effects of metacognitive instruction embedded within an asynchronous learning network on scientific inquiry skills. *International Journal of Science Education, 27*. (8), 957–983. https://doi.org/10.1080/09500690500068626

Appendix

9.3 Appendix A: The article which was published at the CSERC'19 conference

The article can be found here:
https://www.ou.nl/documents/49898/761136/CSERC_19_collected_papers_smaller.pdf/d1eed513-4ecf-edcb-a328-104d9167be4b

Programming for teachers: Reflections on the design of a course supporting flexible learning trajectories

ABSTRACT

How to design an online flexible learning trajectory course where students are in-service teachers with varied level of programming knowledge, interests, and different application need? This paper presents the design of such a course for teachers on applied programming. The main learning objective of the course is to provide in-service teachers with insight into how programming can be used to create digital solutions. The course is practically directed and emphasizes programming as a constructive and creative tool. The course is aimed at teachers in secondary schools. The paper describes the main design choices of the course. Based on the experience with the course, the paper reflects on the challenges to design courses that do not support a single learning path for all the students, but rather aims at providing a context where students can identify and follow the learning path that is best fitting for their competencies, interests, and needs of the local practices.

CCS CONCEPTS

• **Applied computing** → **Education**; **E-learning**; *Distance learning*; • **General and reference** → Design; • **Social and professional topics** → *Computer science education*; Adult education.

KEYWORDS

Online course, Programming for teachers, Flexible learning trajectories, Applied programming, Learner-Centered Design, Continued Education

ACM Reference Format:

. 2019. Programming for teachers: Reflections on the design of a course supporting flexible learning trajectories. In *The 8th Computer Science Education Research Conference, 18-20 November 2019, Larnaca, Cyprus.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Teaching programming in schools is challenging, and can also be overwhelming if programming is not part of your educational background. The demand for incorporation of programming into the curriculum of various subjects in schools is on the rise, and teachers need some additional education and guidance regarding that. The course is aimed at in-service teachers who represent the group that has the most difficulties with this topic as they usually did not have any kind of formal programming education in their careers thus far. Some of the challenges that are discussed in this paper are: Students are in-service teachers who teach at different levels (primary, secondary, and upper secondary level), different needs for application of programming to varying levels of education, flexible course versus class size and assessment, "learning programming is hard" and "programming for all".

In this paper, we reflect on how these challenges in the everyday practice of teachers influence the design of in-service training for teachers. The discussion is based on the experience with the design on an online course: *Applied programming*.

The paper is organized as follows: Section 2 presents related work and positions the paper in the context of current teacher training in the area of programming. Section 3 presents the overall design of the course, and Section 4 discuss the results on the course based on the instructor's experience and feedbacks from participants. Lastly, section 5 outlines some implications for course designers and instructors.

2 RELATED WORK

"Computational methods and models give us the courage to solve problems and design systems that no one of us would be capable of tackling alone" [12]. Wing argues that computational thinking should be included in the lower education curriculum, which indeed is in coherence with the report *Students' Learning in the Future School* by the Norwegian Directorate of Education and Training [7].

A study on student engagement in online discussions [1], concluded with the need for including pedagogical course designs in closer relation to online learning. In the course which is described in this paper, Applied Programming for Teachers, we have tried to implement didactic features of online learning by using a learner-centered design for the course structure [2], which is explained further in section 3.

To understand the place that this course takes, we need to understand the variety of courses that are delivered in an online format. Massive Open Online Course (MOOC) is a widely adopted type of course and most dominant form found today. They are aimed towards participation on a large scale with availability to anyone with access to the internet. Various reports on the effectiveness of this type of course have been made. It has been reported that adoption and drop-out rates vary, in some cases 50% or more of the participants cease their activity in the first weeks with the decline to 16% in the following weeks [4]. Also, various reasons have been reported as reasons for high drop-out numbers, some of them include lack of motivation, poor contact with the staff, long times to receive answers to queries, too constrained syllabus, and similar [6, 13, 14]. As MOOCs usually do not have targeted demographics, the assessment of the course effectiveness becomes hard other than reporting competition numbers.

CSERC '19, 18-20 November 2019, Larnaca, Cyprus

ACM ISBN 978-1-4503-9999-9/18/06…$15.00
https://doi.org/10.1145/1122445.1122456

In recent years, several projects for teaching coding to teachers have emerged [6],e.g., Switzerland has an ongoing educational reform that requires them to conduct continued education for teachers in regards to programming and computational thinking [5]. In the context of Norwegian schools, Simula Research Laboratory[1] has been working with teachers about how programming can be implemented in classroom situations with an interdisciplinary focus similar to the course described in this paper. In contrary to Simula's face-to-face and seminar-based approach, the Applied Programming for Teachers course has adopted a web-based structure, which according to Hadjerrouit [3] offers increased learning benefits when combined with a learner-centered structure.

3 DESCRIPTION OF METHOD AND CASE

The purpose of this study is to promote how to design a course with flexible learning trajectory for students who are in-service teachers. In the following sections we will give a brief description of the challenges and how to design such a course.

3.1 Background

The message from the Norwegian government, [9] says: *"The curriculum will be renewed so that it reflects the current school life and the challenges children and young people face today."*.

Norwegian Directorate for Education and Training (Utdanningsdirektoratet) is working on renewing all the curricula in primary and secondary education, which will be implemented from 2020 onwards. The purpose of renewing *"The knowledge promise"* is to make children and young people able to meet and find solutions for todays and future challenges. They will develop relevant expertise and good values and attitudes that affect the individual, in a society characterized by greater complexity, high diversity, and speed change"[10].

To support this, the Centre for Continuing Education and Professional Development at the University offers several courses within Information and Communication Technology (ICT) Programming. These are online courses that provide teachers with insight into how programming as a subprocess of the more significant problem-solving methodology is used to create digital solutions. Using a programming language can create a solution to a problem. The program is practically directed and emphasizes programming as a useful and creative tool. The focus is on how the programming subject can be communicated to students with a focus on creativity and collaboration in task solving. The program qualifies for teaching in programming at levels 8-13 (and earlier).

The target audience is teachers who need programming skills and insight into the possibilities of coding, design, and modeling of software-based solutions. These courses give guidance of programming in schools and other subjects and activities where programming is used to support learning.

Lectures are web-based, but the emphasis is placed on social and interactive learning with weekly activities such as online lectures and regular compulsory work requirements (exercises). Lectures include interactive learning materials and videos made available to students, and online collaboration and guidance are conducted in social spaces (Slack[2] and Blackboard[3]).

The teaching is based on the curriculum consisting of both textbook [2] and online resources. In teaching, we use both text and block-based programming tools. Through the obligatory exercises, the students will try out new academic and subject didactic knowledge in their teaching.

3.2 Challenges

Through the process of designing this course and further reflection we have identified some challenges which we briefly explain in this section.

Students are in-service teachers who teach at different levels (primary, secondary, and upper secondary level). Our focus is teachers who teach in level 8-13, but we also try to meet the needs of lower-level teachers. We achieve this by including games and block-based programming.

Programming at primary and secondary schools will require other forms of programming didactics than those who wish to apply for upper secondary school programming. This requires a flexible course content that will meet the needs of all students in order to be able to implement programming in their respective areas.

Different needs for application of programming at different levels of education. Teachers from the upper secondary level teach in many different subject areas (automation in technical subjects, natural science, and history/languages, etc.). At lower levels, the students have other needs and issues related to programming and its applications. This challenge requires highly flexible course content, and students should be able to choose the direction which is relevant to them.

Flexible course versus class size and assessment. There is a need to increase the number of students taking this course. Defining a course content and having a form of assessment where the workload does not escalate when the number of students increases, is, therefore, a challenge that must be dealt with in the right way.

Learning programming is hard. *"We have significant empirical evidence that learning to program is harder than teachers might predict."*[2]. One crucial aspect is motivation and a strong desire to learn programming. *"Critical to success in learning computing is wanting to learn computing"*[2]. Learning programming languages might be more comfortable for some people, and more difficult for others. *"Becoming a good programmer is incredibly difficult, and it does not happen quickly"*[11].

The aim of this course is, therefore, to teach programming to all students at Norwegian schools in a way that increases their interest and commitment. Through this course, our task is to transfer this engagement with the students (who are in-service teachers) in a way that enables them to see the benefits of applying programing in their area while also being able to increase their student's interests. How to learn programming concepts, thus becomes an essential aspect of this course.

Programming for all [2] There is a broad range of reasons for making computing education available to everyone. These are

[1] https://www.simula.no/news/simula-educates-teachers-programming

[2] https://slack.com/

[3] https://www.blackboard.com/about-us/index.html

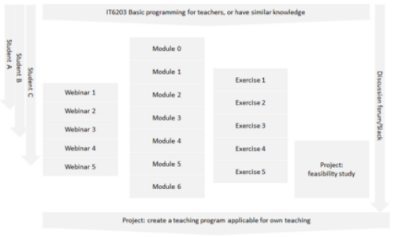


Figure 1: Organization of IT6204 Applied programming for teachers

Webinars are complementary online lectures and are meant to help students get a broader view of core topics. Also, students will have a chance to ask questions directly or discuss different subjects related to the course. All webinars are recorded and made available through the learning platform.

Module 1 - Introduction, why and how everyone should learn programming. There is a high demand for education in computing and information technology. Many students (both children and adults) are aware that they will need programming in the future. This module presents arguments for ubiquitous programming education while allowing for individual learning motivation and application of use. What do we mean when we talk about everyone having to learn to program? How can we create an educational program that works for everyone? In this module, we look at the use of a learner-centered design of computing education (student-centered) approach to reach a broad audience. Several reasons for teaching programming to everyone are discussed, and we study how the various cases lead to different choices of learning objectives and teaching methods.

Module 2 – Block based programming. This module deals with applications of programming using block-based programming. All students are encouraged to complete this module even though its content is more applicable to primary/secondary schools. The chosen technologies for this module are *micro: bit, code.org*[8], *Scratch*[9], *and Pocket Code*[10].

Module 3 - Principles, constructions, and structures in modern programming. In this module, students extend their programming knowledge on selected topics (relevant for natural science) to an advanced level. Also, they will learn more about algorithms and how to apply programming in natural science.

Module 4 - Understanding the software's function in electronics and robotics. In this module, we apply the application of programming in electronics and robotics. Several techniques (microelectronic technologies) are introduced, but it is up to students to choose which of these he/she wants to immerse in. Techniques introduced are relevant to teaching at all school levels.

[8]https://code.org/
[9]https://scratch.mit.edu/
[10]http://robotixedu.com/phiroresources/introduction-to-pocket-code.html

Module 5 - Game programming. Making games allows us to be high-interest, engaging, teaches foundational and transferable skills [11]. Using games in teaching can, therefore, be very motivating for the students, but it is crucial that the level is right so that they are not overwhelmed by information and technical difficulties. Game programming can be so advanced that it governs the overall IT development, both in terms of graphics and artificial intelligence. On the other hand, it can be done very quickly, and in this module, we make a game with Scratch (block programming).

Module 6 - Programming languages, tools, methodology, and testing. In this module, we address some key topics relevant to professional software development. The purpose is to familiarize students with the software development process itself, methods and techniques that are used professionally to develop computer systems. We have introduced the most central topics in this lesson and enclosed documents that deal with the topics in greater depth for those who are particularly interested in increasing their general competence in software development.

Exercise. Exercises follow the flexibility of the modules. Students choose relevant tasks from the exercises based on the topics they have select to immerse in.

Project. Student assessment will be based on exercises and project delivery. When a student has completed a minimum set of exercises, they can start to work on the project. It is, however, possible to start a feasibility study for the project before mandatory exercises have been completed.

The project is about to create a teaching program that the student can use in his/her class to teach programming.

Project delivery requirements:

- Scope: It is expected that a minimum of 40 hours per person will be worked on with the project (pre-project / previous exercises not included). Students can work individually or in groups of a maximum of three persons.
- A simple project report per person shall be provided containing hourly consumption per activity and reflections on self-learning/results.
- Project description (preliminary project): A step by step description of how the student intends to implement the teaching program
- A complete teaching program that includes how the student plan to teach programming in the classroom. This implies:
  - Program code and screen dumps
  - Video of a maximum of five minutes showing a demo of running program or robot.

Discussion forums A forum where students can discuss with each other and trainers of the course is an important part. Slack[12] is used as a tool for this purpose.

3.5 Course exercises

In this flexible course, we have designed exercises that correspond to the flexibility of the topics. The aim is that every student can have benefits of the course regardless of which level the student teaches and what level of programming knowledge he/she has. To achieve

[11]https://codakid.com/why-coding-games-is-the-best-way-to-teach-kids-computer-programming/
[12]https://slack.com/about

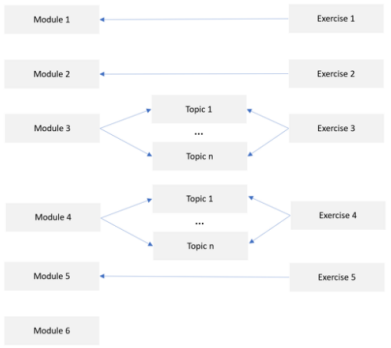


Figure 2: Flexible exercises

this, we have defined a minim set of tasks in each exercise. Each task can either be mandatory or optional. Figure 2 shows the design of exercises and their connection to topics in each module. Modules 1, 2, and 5 in the course are mandatory for all students. There is no flexibility in these modules, and therefore, tasks specified are not optional, and students must resolve the minimum set of tasks to be able to pass. However, in module 5 (Game programming), the student is supposed to create a game. The technology and complexity behind the games that students creates can be flexible. Module 3 and 4 are the main modules for application of programming and covers many areas. The specified exercises are also reflecting the same flexibility. In exercises 3 and 4, there are some mandatory tasks. Also, the students need to resolve a minimum set of optional tasks. They can select tasks depending on what topics in the module they have immersed in.

In Module 6, we address some key topics relevant to professional software development. The module is optional, and no exercise has been defined for it. Based on feedback (see figure 6 question 33), 52% of the students find this module useful. This is expected since the module is supposed to address topics only for those students who want to gain advanced knowledge to have a better understanding of the overall picture.

3.6 Course assessment

The main requirement for completing the course is that students conduct a project, as explained in the previous section. This project has two sub-deliveries. The first delivery is a feasibility study (sub-delivery one) where the students specify what programming concepts they will cover and how this can support/increase the understanding of the profession for their students. The first sub-delivery will be evaluated, and feedback is given to students before they are allowed to continue with sub-delivery two, which is the actual

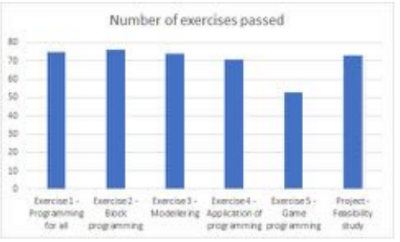


Figure 3: Overview of passed exercises

implementation of the teaching program they have specified in the first sub-delivery.

A prerequisite for starting the project is that students have to deliver a minimum of 70% of exercises in advance. Each exercise is organized in a way that reflects the diversity and flexibility of the course content. This means that students need to resolve a minimum set of questions in each exercise, but they can choose what optional questions to resolve.

Mandatory exercises force the students to study the learning materials which are relevant and interesting to students, which in turn make the basis for their project.

Figure 3 shows number of exercises passed. A minimum of 4 exercises in addition to "project feasibility study," had to be delivered. Exercises 3-4 were designed in a way that students could select a minimum set of tasks that were relevant with regards to complexity and application in their subject area.

73 students got a grade after delivering the mandatory project in the course. 80 students started the course in January. This gives a retention rate of approximately 91%. This result should be seen in the context of an online course (7,5 ECTS) where there is a clear commitment between teachers and schools to give teachers some free time to take the course. Therefore the result is excellent, but should be seen in this specific context.

4 RESULTS

4.1 Flexible Learning Trajectories

Figure 4 is showing schematic view of how flexible learning trajectories are implemented in this course. A learning path consist of common mandatory modules; flexible modules and a project delivery.

The different areas for applications of programming within electronics and natural science/mathematics in the course have been defined as follows: micro: bit, block-based (Scratch), Pocket Code, Raspberry PI, Arduino, LEGO Mindstorms, Game programming (Minecraft, PyGame), Modelling (natural science and mathematics). In table 2, we have an overview of what modules students have been using during their project.

Some examples of learning trajectories that students have followed in this course:

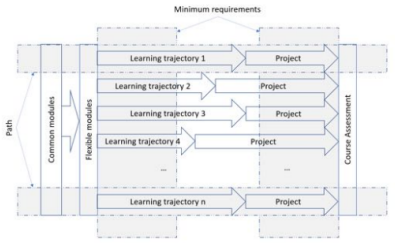


Figure 4: Schematic view of flexible learning trajectories

Example 1:

- Level
  - Upper secondary school
  - Mathematics
- Modules
  - Module 1: programming didactic
  - Module 2: Block based programming
  - Module 3: Application of programming within natural science
  - Module 4: micro:bit, robatics
- Topics
  - Modelling, Mathematics
  - 3D printing
  - micro:bit
- Project: Measurement of acceleration, speed and distance using programming

Example 2:

- Level
  - Upper secondary school
  - Technical and industrial production
- Modules
  - Module 1: programming didactic
  - Module 2: Block based programming
  - Module 3: Principles, constructions and structures in modern programming
  - Others (topics not covered in the course): C++, Programmable logic controller (PLC)
- Topics
  - Python
  - C++
  - Programmable logic controller (PLC)
- Project
  - Part 1: A basic course in "Use and programming of PLS for VG1 electrical subjects"
  - Part 2: Course in sequence control with PLC. Adapted to class level VG2 electric energy plus especially interested students in VG1 electrophysics.

Table 1: Application of programming in student projects

Application areas	Language								
	C++	Python	KRL	Python/C++	Block/Python	HTML/JavaScript	Block	Block/JavaScript/C++	Block/C++
App Lab							1		
Arduino	**10**								
Blue-Bot							1		
Games							1		
iPad							1		
iPad/Pythonista		1							
KUKA-robot			1						
Lego Mindstorm							**11**		
micro:bit							**21**		
micro:bit, bitbot							1		
micro:bit, Arduino									1
MineCraft							1		
PLS/Arduino								1	
PyGame		1							
Python		**9**			2				
Python/C++				1					
Pythonista		1							
Raspberry pi		2							
Scratch/Ras.Pi					1				
Spyder		2							
TinkerCad							1		
Website						1			
Totalsum	**10**	**16**	1	1	3	1	**39**	1	1

  - Part 3: Basic course in text-based algorithmic programming using C ++ with emphasis on electrotechnical calculations. This course will probably fall outside the limits of the ordinary VG1 Electrical Subjects, but it should be able to be used within "Vocational Education Specialization »where the students work from the curriculum goals for later VG3 studies. (Automation and Computer electronics.)

Example 3:

- Level: Elementary school
- Modules
  - Module 1: Programming didactic
  - Module 2: Block based programming
- Topics: Block based programming
- Project: Programming BlueBot to demonstrate programming concepts

Examples 1 and 2 are both on upper secondary level, but subject area and complexity they have chosen are different. Example 3 is showing a path which fulfills the minimum requirements, but is directed towards elementary school.

Table 2: Projects related to modules

	Project
Module 2, 3	2
Module 3	14
Module 3,4	16
Module 4	34
Module 4, Others	2
Module 5	3
Others	2
Totalsum	**73**

The most popular applications of programming in the project delivery has been the use of micro: bit, Arduino, and LEGO Mindstorms. Projects based on block programming are 53% (39 projects), Text-based projects are 40% (29 projects) and 7% (5 projects are mixed).

Relation between modules and student projects are also shown in table 2. Modules 3 and 4 deal with application of programming within different areas covered in the course and most projects are related to these modules. There are 2 projects in "Others" modules which are not covered in the course, but still accepted and will be considered to be added next time the course is executed.

There are both students with little or no programming knowledge and students who are advanced programmers. We have covered a wide range of topics that students can choose to immerse in. Still, some observation has been done where students have been asking for new areas that were not covered in the course (e.g., automation, programming of PLC (programmable logic controller) and 3d printing). More research needs to be done to identify all relevant subject areas in primary-, secondary- and upper secondary schools in regards to the inclusion as mentioned above of programming in specific curriculums. These areas need to be included in (or considered) the learning trajectories of the course. We have also observed that students with little or no programming knowledge and students with advanced programming knowledge have expressed that this flexible course has been useful to them since they can choose a learning trajectory that is relevant for them.

An advantage of having a class where a different level of knowledge, objectives, and interests for applications exist, is that we can use the class as a learning resource environment. We utilized Slack to enable students to interact with each other and observed a high level of participation amongst students of different skill levels. They share their knowledge, which in turn may increase the level of programming skills.

We are experiencing that there is a considerable demand for passing the programming courses, and we are expecting this demand to increase the next few years (the reason is explained in section 3.1). The main challenge of increasing the number of students in the assessment process regarding a parallel increase in workload for staff members. To accommodate the demand for more students, we need to reassess how compulsory assignments are implemented in the course, and how we can increase efficiency and general scalability.

The implementation of flexible learning trajectories in this course allow teachers to study topics which are applicable to their respective school subjects and how programming can be used to convey knowledge in a variety of topics. Reflection notes provided by the students as part of their project work show that their engagement has increased considerably during the project period, and they have discovered new areas where they can teach programming for their students. They also welcome the fact that the results of the project can be used immediately in their teaching classes, which means that they put more effort into the implementation of the project.

4.2 Course evaluation

At the end of the course, we sent a questionnaire to the students for completion. They were informed that anonymized data from the survey could be used in research related to the subject. A total of 22 students filled in the questionnaire while the approximately same number of students decided not to complete the questionnaire (stopped at first question). The result is shown in figure 5. A detail graph is shown in ??. In addition to these results, we also asked students to deliver a reflection note as part of their project deliveries. These reflections evaluate both this course and their project results, which are discussed in session 4.3.

Questions 39 and 40 are related to the project, and respectively, 71% and 67% are happy with their project delivery.

Questions 34 through 38 are related to course exercises. Most of the students (>65%) agree that exercises have been relevant and were not hard to resolve. However, we have got feedback from some students saying that exercise 3, which is related to the application of programming within natural science and mathematics (modeling), has been too hard to resolve. In this exercise, we assumed that all students have a minimum understanding of essential mathematical functions like sinus, cosine, etc.

Questions 28 through 33 are related to actual topics in each module and how relevant they were for students. Results are slightly weaker (52%-68%) but still within the range of acceptability. In module 1, we discussed programming didactics. We focused on questions like what is computational thinking, why should everyone learn to program, what are the challenges, etc. To gain a better understanding of these topics, we used the textbook Learner-Centered Design of Computing Education [2]. For this module, the satisfaction rate is 64%. Most students (68%) are also satisfied with module 2 (Block-based programming). This is also reflected in table 1 where most students have selected block based (micro: bit) as their project. The satisfaction rate for module 5 (Games programming) is lower (52%). The number of students that have delivered is lower for this exercise (see figure 3) compared to others. 24% of students have answered "Neither nor" to this question, which could mean that they have not done the exercise.

Questions 3 and 4 are related to using communication channels. 90% of students are satisfied using Slack. A total number of 1540 messages have been registered in Slack during the course. Figure 7 shows daily activities. The activity level of Slack confirms the high rate of satisfaction in the course. In addition to Slack, some students preferred to have direct communication with instructors/teaching assistants using e-mail instead of or in addition to Slack. Questions have been answered as quickly as possible to ensure that students get clarification in the shortest possible time. We would like to analyze and discuss use of communication tools used in this course

Figure 5: Course evaluation - Overview

in a separate paper. Therefore, we do not go into more details on use of Slack here.

Questions 5 and 6 are related to the use of Skype as a webinar tool. Only 50% of students were able to connect successfully to the webinars. The rest had either technical issues or was not able to join due to other reasons. At this point, we do not know how many of these students had technical issues, but based on comments during the webinars, we have been told that either they could not see the picture or they could not hear the sound. However, 64% are satisfied with the recorded version of webinars.

Questions 7 through 16 are related to the contents of the webinars. The first webinar had a focus on introducing the course and giving practical information. The last webinar focused on the project (all sub-deliveries). The other three webinars discussed different topics relevant to teachers. The satisfaction rate is lower compared to other areas of the course and the main reasons for this may be related to these conditions: Firstly, many students had technical issues connecting to these webinars (See comments on questions 5 and 6). Secondly, students have responded that they wanted a better connection between topics discussed in the modules and the topics in the webinars.

4.3 Reflection notes

In addition to results from the questionnaire discussed above, we have received a reflection note from each student who passed the course as part of their project delivery. These reflection notes are discussing two subjects: Firstly, how this course has increased their knowledge about applying programming in their subject-area and their more profound understanding of programming concepts. Secondly, students have created their own "teaching program" as the project delivery. Some of them have been able to run this program in the class for their students and have reflected on the results.

In this paper, we do not analyze all the results from these reflections, but can point to few reflections which confirms that the project has been a process for students to gain a better understanding of programming and how to apply it in their subject-area:

"What I first and foremost appreciated about this project is that it motivates me as a student and teacher as I spend time on a teaching program that is aimed at my practice...I spend time on this through the study program, but at the same time, I get a program that I can use myself in my practice. I believe that we have produced a teaching course that is very beneficial for the competency goals in the elective course programming in secondary school."

Several students are giving this type of feedback which confirms that it is motivating for students who are in-service teachers to work on a project where results which are rooted in the competence goals can be directly used in their class.

"Programming engages, it creates engagement and collaboration. In the teaching situation, we find that the students talk a lot together, and there is good work noise in the classroom. We see that this motivates the students very much. However, we also see that it requires a lot of knowledge and expertise from the educators, and sees that it is important to be well prepared. It is important to have clear learning goals and criteria for each session. The fact that the students have written a log along the way and have had to submit a description of their programming project has meant that the work the students have done has been more thorough..."

This is another type of reflection where students observe an engaged and motivated class. Another student reflects on how it went when they executed their teaching program development during the project in their class: "The main impression I am left with after the completed project period is that the task was comprehensive and demanding, but feasible, fun, and very educational. It differentiated well and gave all students, regardless of their skill level, the challenges that suited them. Some students were very self-sufficient and needed little help with the task. They showed high competence in terms of problem-solving and coding. Others needed a great deal of help with the coding itself. It was intense weeks for both students and teachers, but everyone came to the goal, and everyone felt that they were getting something they thought was impossible beforehand..."

Each student had delivered a project report where they are reflecting on how it went during project execution and when they executed their teaching program in their class. We have chosen to quote a few of these in this paper to show the type of results and what students think about the project in this course.

Some students with little or no programming knowledge experienced that the introductory course was hard, while it was easier for them to understand the programming concepts in the applied programming course.

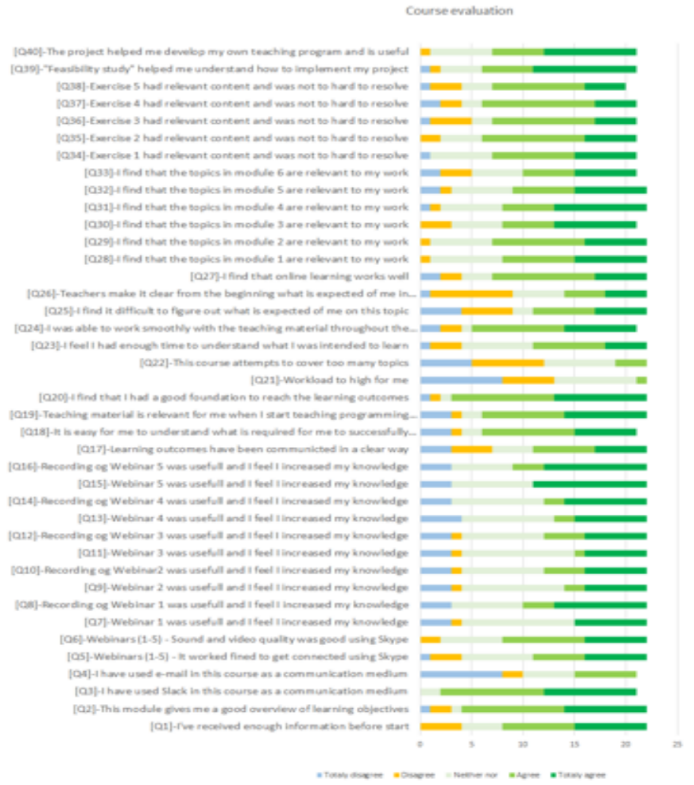


Figure 6: Course evaluation - Detail

5 LESSONS LEARNED

Multiple dimensions of flexibility in this context mean providing a fully flexible course for teachers in elementary and secondary schools in a way that helps them achieve the learning goals. E.g., selecting relevant content from the learning material and have the flexibility of choosing their topic in the project for creating their own 'teaching program'.

Teachers are usually postgraduate students with varied backgrounds and experience in computing education. One dimension is to consider this variety of backgrounds, and another dimension is that teachers are going to teach computing education to students on different levels. Teaching programming to all introduces other topics like programming didactic and gaining the understanding that computing education is important and necessary for the future.

By a variety of backgrounds as one dimension, we mean that students have different levels of programming knowledge. Some students are experienced programmers, while others do not have any programming knowledge at all. Also, teachers are teaching

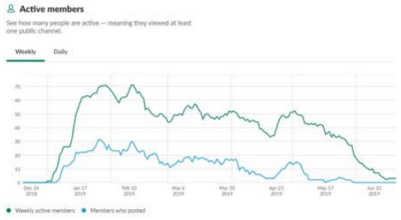


Figure 7: Slack

in a wide variety of subjects. Some examples are Mathematics, Art-, design and architecture, Media and communication, Music, dance and drama, Sports and physical education, Electricity, and electronics, Technical and industrial production, etc. [13]. This variety of background knowledge of programming and subject areas in the same classroom may lead to the design of a flexible learning trajectory course. Projects delivered by students (see table 1) reflects variety of programming applications and reflection notes in section 4.3 indicates that this flexibility is desirable. We believe that in a class-context with these varieties, a flexible learning trajectory course is a good choice.

The course was designed to provide flexible learning trajectories in some areas and mandatory in other areas of the curriculum. The first part of the course was mandatory for all students. The main goal was gaining the understanding that computing education is for everyone. This prevented students from spending much time discussing why they should learn programming and contributed to increased motivation and engagement for the rest of the course.

Module 1 and exercise 1 deal with this. Delivery of exercise 1 in figure 3 shows high student participation. the students have engaged a lot in these questions and, based on the feedback, this part of the course has created increased motivation for further learning.

Individual trajectories were planned for other topics of the course since the level of knowledge on computing education, their interest and local practice (this is an element that is different from courses for regular students) are very different. It is challenging to create enough learning trajectories to cover both students interests and what is required to know and teach in their respective subjects. However, having in mind that the aim is to increase students' interest and engagement in this field makes it easier to design the course in a way that most teachers find it useful.

Teachers need to have a clear understanding of the learning outcomes for their students. They need to have a clear understanding of the requirements of the new curricula provided by Norwegian Directorate for Education and Training in primary and secondary education. Based on this, and support from the course instructor, they can select a learning trajectory that is most appropriate for their subject.

[13] https://www.vilbli.no/en/en/no

Having a course where students can select a learning trajectory among many, makes it challenging for course instructors and teaching assistants (TA). Many different areas for the application of programming need to be supported in the course. To manage this, we have given each TA a separate area to focus on. Each TA is responsible for one or more areas of applied programming. Another challenge has been cooperation among students. Since they have different areas of interest, it can be challenging for them to get the support they need. We have introduced the use of Slack as a tool for discussions in addition to instructors and TA's actively supporting the students. Providing a set of links to relevant online Internet resources helps students find the answers to some of their questions.

5.1 Course webinars

Webinars have been used in this course as complementary lectures and a way of direct communication between students and instructors.

These types of lectures are appreciated by students, but need to be planned and executed carefully. Finding a time that suits all participants can be challenging (see section 4.2). In addition, students are spread around the country and use different types of equipment which can create technical issues with connection. When a proper solution to these issues are available, webinars can be useful in such a model.

Based on the results of the course evaluation, a tighter connection between Webinars and topics of the course is desirable. This will help students get a better understanding of each module and they will be in a better position to select a proper learning trajectory in the course.

5.2 Teaching assistants

Several TAs have been used to manage the correction of exercises in addition to answering questions and making clarifications. A course supporting flexible learning trajectories requires TAs to have a wide range of competencies and skills. Finding these types of resources could be challenging, especially if the number of students in the course increases, and the need for employment of TAs increases proportionally. To resolve this, we shared the responsibility of different topics to different TAs. E.g., some had a responsibility to answer questions/make a clarification to Arduino related discussions and exercises while others had responsibility for Raspberry Pi, etc.

We find the use of TAs with different backgrounds in programming to be appropriate in such a model.

5.3 Implications for course re-design

As the approach is learner-centered, we need to tackle the problem as a design-based-research project and use feedback from the first cohort to redesign the course where needed.

A flexible course design model as explained in this paper does not scale up well when number of participants increase. Huge number of students will result in proportionally additional work for instructors to do assessment.

Another implication in such a flexible course design is for students to find the correct learning trajectory. In-service teachers with little or no programming knowledge may need assistance to

find the right path and this challenge will increase as number of students increases in the course.

Some topics for application of programming seem to be more relevant then others (see table 1). Re-design of the course must focus on these areas which will result in reducing the flexibility of the course.

6 CONCLUSIONS AND FUTURE WORK

Designing a course that supports flexible learning trajectories introduces many challenges that need to be addressed. At the same time, it opens opportunities to increase student learning outcomes in a course where in-service teachers have varied level of programming knowledge, interests, and different application need. A learner-centered design process for computing education[2] is the basis for the design of a flexible programming course for in-service teachers. In the case of this course, this design helped creators to reach their initial idea and has provided a good structure for flexible learning trajectories. Some areas of future research on this topic are:

(1) A full evaluation of communications channels used in the course (Slack email)
(2) How to help students to reflect on their learning needs and choose the right learning trajectories
(3) How to help students to reflect on their teaching practice
(4) How to promote cooperation among students towards the establishment of a Community of Practice (CoP)

In this paper, we have explained how the course has been designed and discussed our initial reflections on results, challenges and possibilities.

REFERENCES

[1] Bodong Chen, Yu-Hui Chang, Fan Ouyang, and Wanying Zhou. 2018. Fostering student engagement in online discussion through social learning analytics. *The Internet and Higher Education* 37 (apr 2018), 21–30. https://doi.org/10.1016/j.iheduc.2017.12.002

[2] Mark Guzdial. December, 2015. *Learner-centered design of computing education : research on computing for everyone;*. Morgan & Claypool Publishers. 147 pages.

[3] S Hadjerrouit. 2005. Learner-Centered Web-Based Instruction in Software Engineering. *IEEE Transactions on Education* 48, 1 (feb 2005), 99–104. https://doi.org/10.1109/TE.2004.832871

[4] Andrew Ho, Justin Reich, Sergiy Nesterko, Daniel Seaton, Tommy Mullaney, Jim Waldo, and Isaac Chuang. 2014. HarvardX and MITx: The First Year of Open Online Courses, Fall 2012-Summer 2013. 1 (2014). https://doi.org/10.2139/ssrn.2381263

[5] Anna Lamprou, Alexander Repenning, and Nora A Escherle. 2017. The Solothurn Project - Bringing Computer Science Education to Primary Schools in Switzerland. *In Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education* November (2017), 218–223. https://doi.org/10.1145/3059009.3059017

[6] Fotis Lazarinis, Christoforos V. Karachristos, Elias C. Stavropoulos, and Vassilios S. Verykios. 2018. A blended learning course for playfully teaching programming concepts to school teachers. *Education and Information Technologies* (2018), 1237–1249. https://doi.org/10.1007/s10639-018-9823-2

[7] Sten Ludvigsen, Pia Elverhøi, Bushra Ishaq, Jens Rasmussen, Daniel Sundberg, Eli Gundersen, Kjersti Kleven, Mari Rege, Helge Øye, Sigve Indregard, Tormod Korpås, and Sunniva Rose. 2014. *Elevenes læring i fremtidens skole.* Technical Report. Oslo.

[8] Bonnie A. Nardi. 1993. *A Small Matter of Programming.* The MIT Press.

[9] Government Norwegian. accessed April 17, 2019. *Government message St. 28 (2015–2016).*

[10] Utdanningsdirektoratet. 2019. *What is the knowledge promise.*

[11] Utdanningsdirektoratet. 2019. *Why Don't More People Work As Programmers.*

[12] Jeannette Wing. 2006. Computational thinking. *Commun. ACM* 49, 3 (2006), 33–35. https://doi.org/10.1145/1118178.1118215 arXiv:=

[13] Di Xu and Shanna Smith Jaggars. 2013. Adaptability to Online Learning: Differences Across Types of Students and Academic Subject Areas. *Economics of Education Review* 37, 54 (2013), 46–57. https://doi.org/10.7916/D82N59NB

[14] Di Xu and Shanna S. Jaggars. 2014. Performance Gaps Between Online and Face-to-Face Courses: Differences Across Types of Students and Academic Subject Areas. *The Journal of Higher Education* 85, 5 (2014), 633–659. https://doi.org/10.1353/jhe.2014.0028

9.4 Appendix B: Code snippet for diagram showing message statistics

```
  let graphData = [];
let usedElements = [];
for ( let i = 0; i < users.length; i++){
    let current = users[i];
    let cnt = 0;

    let usedCheck = false;
    for (let j = 0; j < usedElements.length; j++){
        if (current == usedElements[j]){
            usedCheck = true;
        }
    }
    if(usedCheck){
        continue;
    }
    usedElements.push(current);

    for (let k = 0; k < users.length; k++){
        if(current == users[k]){
            cnt++;
        }
    }
    graphData.push(cnt);
}
console.log(graphData);
let sortedGraphData = graphData.sort(function(a, b){return a-b});
let half = Math.floor(sortedGraphData.length/2);
let median = sortedGraphData[half];
console.log("Median is " + median);

  var ctx = document.getElementById('myChart').getContext('2d');
  var myChart = new Chart(ctx, {
      type: 'bar',
      data: {
          labels: uniqueUsers,
          datasets: [{
              label: '# of Votes',
              data: graphData
              ],
              borderWidth: 1
          }]
```

```
        },
        options: {
            scales: {
                yAxes: [{
                    ticks: {
                        beginAtZero: true
                    }
                }]
            }
        }
    });
```

9.5 Appendix C: Interview guide

Questions for learning assistants (semi-structured):

- Did you have any experience with Slack prior to IT6203/IT6204? If yes: in what context?

- In your opinion, how does Slack differ from the Blackboard forum?

- Did the usage of Slack change in IT6204 compared to IT6203? How?

- How did you usually help students; textually, through voice conversations on Skype, something else?

- What is the main drawback of not meeting the students physically?

- If you could change Slack in any way, what would you do?

- Do you have any experience using Discord?

- Discord is quite similar to Slack, but it comprise of more features, like voice conversations and screen sharing. Do you think that would have been useful as a learning assistant in these courses? Why?

- Did you get the impression that the students created a community for sharing and helping each other, or was it mostly the teaching assistants answering questions?

9.6 Appendix D: NSD registration form

Meldeskjema 294179

Hvilke personopplysninger skal du behandle?

- Navn (også ved signatur/samtykke)
- Lydopptak av personer
- Andre opplysninger som vil kunne identifisere en fysisk person

Type opplysninger

Du har svart ja til at du behandler andre opplysninger som vil kunne identifisere en person, beskriv hvilke

De intervjuede personene vil bli omtalt om læringsassistenter i emnet IT6204 Anvendt Programmering for Lærere ved NTNU våren 2019. Det var 5 læringsassistenter og jeg planlegger å intervjue 2 av dem.

Skal du behandle særlige kategorier personopplysninger eller personopplysninger om straffedommer eller lovovertredelser?

Nei

Prosjektinformasjon

Prosjekttittel

Building a Community of Practice in an Online Programming Course for Teachers Through Asynchronous Discussions

Dersom opplysningene skal behandles til andre formål enn behandlingen for dette prosjektet, beskriv hvilke I samtykkeskjemaet ber jeg intervjueobjektene godta at anonymiserte transkripsjoner lagres fram til sommeren 2020 for et potensielt oppfølgingsstudie. Dette er i tilfelle jeg skal skrive en artikkel til en konferanse e.l. i etterkant av masterinnleveringen.

Begrunn behovet for å behandle personopplysningene

Ettersom studiet omhandlet emnet IT6204 Anvendt programmering for lærere ved NTNU, er det nødvendig å skrive i publikasjonen at intervjuobjektene var læringsassistenter i emnet for at deres betraktninger skal være relevant for oppgaven.

Type prosjekt Studentprosjekt, masterstudium

Kontaktinformasjon, student

Sondre Stai, sondre.stai@ntnu.no, tlf: 93878249

Behandlingsansvar

Behandlingsansvarlig institusjon

Norges teknisk-naturvitenskapelige universitet NTNU / Fakultet for informasjonsteknologi og elektroteknikk (IE) / Institutt for datateknologi og informatikk

Prosjektansvarlig (vitenskapelig ansatt/veileder eller stipendiat)

Monica Divitini, divitini@ntnu.no, tlf: 91897790

Skal behandlingsansvaret deles med andre institusjoner (felles behandlingsansvarlige)?

Nei

Utvalg 1

Beskriv utvalget Læringsassistenter i emnet IT6204 våren 2019 ved NTNU i Trondheim

Rekruttering eller trekking av utvalget

Kom i kontakt med utvalget via fagansvarlig i IT6204

Alder

20 - 30

Inngår det voksne (18 år +) i utvalget som ikke kan samtykke selv?

Nei

Personopplysninger for utvalg 1

- Navn (også ved signatur/samtykke)
- Lydopptak av personer

• Andre opplysninger som vil kunne identifisere en fysisk person

Hvordan samler du inn data fra utvalg 1? Personlig intervju

Grunnlag for å behandle alminnelige kategorier av personopplysninger

Samtykke (art. 6 nr. 1 bokstav a)

Informasjon for utvalg 1

Informerer du utvalget om behandlingen av opplysningene?

Ja

Hvordan?

Skriftlig informasjon (papir eller elektronisk)

Informasjonsskriv

Tredjepersoner

Skal du behandle personopplysninger om tredjepersoner?

Nei

Dokumentasjon

Hvordan dokumenteres samtykkene?

• Manuelt (papir)

• Elektronisk (e-post, e-skjema, digital signatur)

Hvordan kan samtykket trekkes tilbake?

Jeg har oppgitt min kontaktinformasjon til utvalget, så de kan enkelt komme i kontakt med meg og trekke samtykket.

Hvordan kan de registrerte få innsyn, rettet eller slettet opplysninger om seg selv?

De registrerte kan få tilsendt datamateriale på e-post som passordbeskyttet PDF eller møtes personlig og revidere innholdet.

Totalt antall registrerte i prosjektet 1-99

Behandling

Hvor behandles opplysningene?

Maskinvare tilhørende behandlingsansvarlig institusjon

Hvem behandler/har tilgang til opplysningene?

Student (studentprosjekt)

Tilgjengeliggjøres opplysningene utenfor EU/EØS til en tredjestat eller internasjonal organisasjon?

Nei

Sikkerhet

Oppbevares personopplysningene atskilt fra øvrige data (kodenøkkel)?

Ja

Hvilke tekniske og fysiske tiltak sikrer personopplysningene?

- Opplysningene anonymiseres
- Opplysningene krypteres under forsendelse
- Endringslogg
- Adgangsbegrensning
- Adgangslogg
- opplysningene krypteres under lagring

Varighet

Prosjektperiode

01.11.2019 - 26.01.2020

Skal data med personopplysninger oppbevares utover prosjektperioden?

Nei, data vil bli oppbevart uten personopplysninger (anonymisering)

Hvilke anonymiseringstiltak vil bli foretatt?

- Koblingsnøkkelen slettes
- Lyd- eller bildeopptak slettes

Vil de registrerte kunne identifiseres (direkte eller indirekte) i oppgave/avhandling/øvrige

publikasjoner fra prosjektet?

Ja

Begrunn

Det blir ikke oppgitt navn eller annen persondata i publiseringen, men intervjuobjektene beskrives som læringsassistenter i emnet IT6204 Anvendt programmering for lærere våren 2019 ved NTNU i Trondheim. Alle læringsassistentene blir ikke intervjuet, men om noen skulle få tak i navneliste for læringsassistentene i emnet vil man kunne spekulere i hvem av dem som ble intervjuet.

9.7 Appendix E: NSD assessment of registration form

29.10.2019 - Vurdert

Det er vår vurdering at behandlingen av personopplysninger i prosjektet vil være i samsvar med personvernlovgivningen så fremt den gjennomføres i tråd med det som er dokumentert i meldeskjemaet med vedlegg den 29.10.19. Behandlingen kan starte.

MELD VESENTLIGE ENDRINGER

Dersom det skjer vesentlige endringer i behandlingen av personopplysninger, kan det være nødvendig å melde dette til NSD ved å oppdatere meldeskjemaet. Før du melder inn en endring, oppfordrer vi deg til å lese om hvilke type endringer det er nødvendig å melde:

https://nsd.no/personvernombud/meld_prosjekt/meld_endringer.html

Du må vente på svar fra NSD før endringen gjennomføres.

TYPE OPPLYSNINGER OG VARIGHET

Prosjektet vil behandle alminnelige kategorier av personopplysninger frem til 26.01.20.

LOVLIG GRUNNLAG

Prosjektet vil innhente samtykke fra de registrerte til behandlingen av personopplysninger. Vår vurdering er at prosjektet legger opp til et samtykke i samsvar med kravene i art. 4 og 7, ved at det er en frivillig, spesifikk, informert og utvetydig bekreftelse som kan dokumenteres, og som den registrerte kan trekke tilbake. Lovlig grunnlag for behandlingen vil dermed være den registrertes samtykke, jf. personvernforordningen art. 6 nr. 1 bokstav a.

PERSONVERNPRINSIPPER

NSD vurderer at den planlagte behandlingen av personopplysninger vil følge prinsippene i personvernforordningen om: - lovlighet, rettferdighet og åpenhet (art. 5.1 a), ved at de registrerte får tilfredsstillende informasjon om og samtykker til behandlingen - formålsbegrensning (art. 5.1 b), ved at personopplysninger samles inn for spesifikke, uttrykkelig angitte og berettigede formål, og ikke behandles til nye, uforenlige formål - dataminimering (art. 5.1 c), ved at det kun behandles opplysninger som er adekvate, relevante og nødvendige for formålet med prosjektet - lagringsbegrensning (art. 5.1 e), ved at personopplysningene ikke lagres lengre enn nødvendig for å oppfylle formålet

DE REGISTRERTES RETTIGHETER

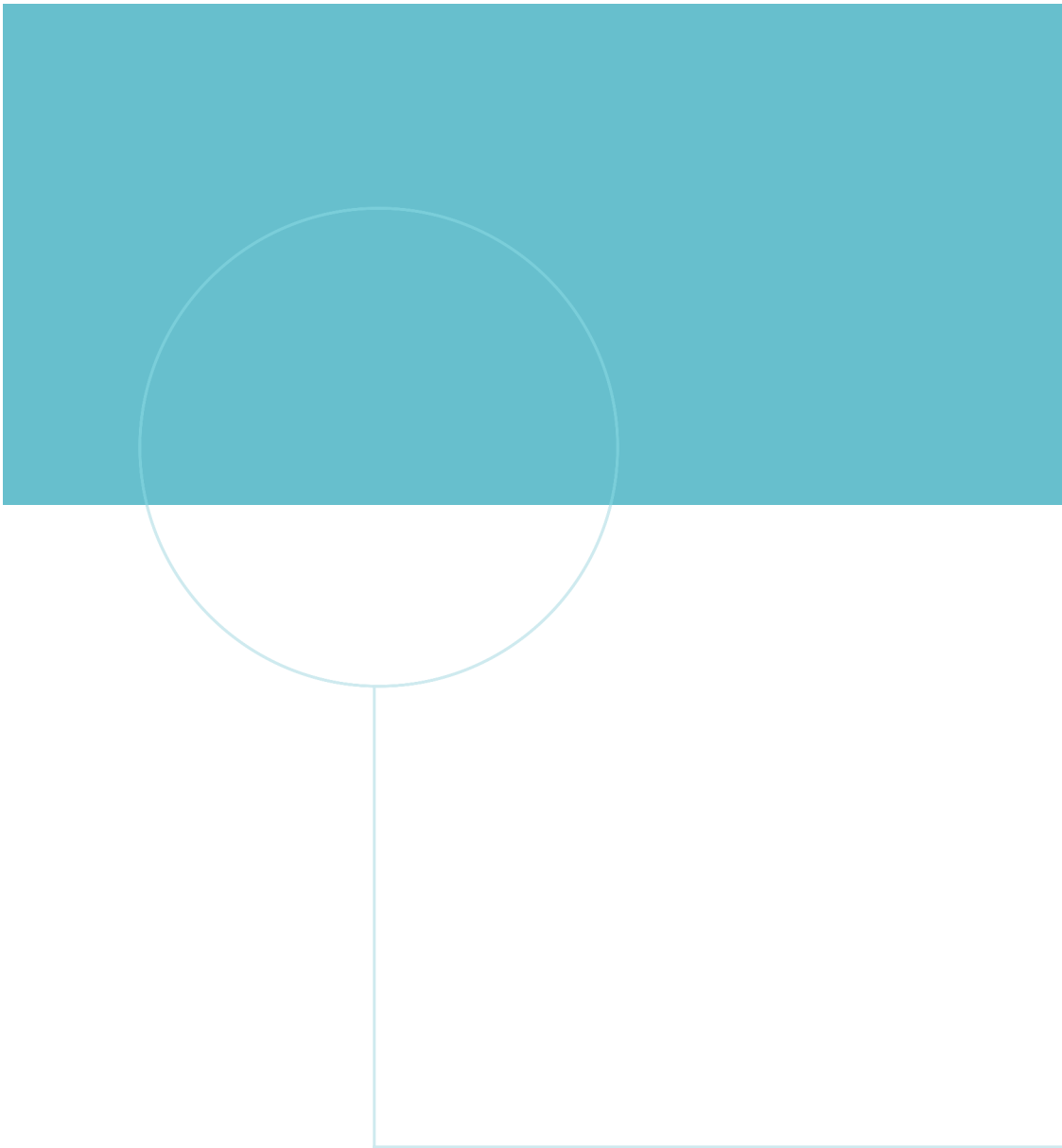
Så lenge de registrerte kan identifiseres i datamaterialet vil de ha følgende rettigheter: åpenhet (art. 12), informasjon (art. 13), innsyn (art. 15), retting (art. 16), sletting (art. 17), begrensning (art. 18), underretning (art. 19), dataportabilitet (art. 20). NSD vurderer at informasjonen om behandlingen som de registrerte vil motta oppfyller lovens krav til form og innhold, jf. art. 12.1 og art. 13. Vi minner om at hvis en registrert tar kontakt om sine rettigheter, har behandlingsansvarlig institusjon plikt til å svare innen en måned.

FØLG DIN INSTITUSJONS RETNINGSLINJER

NSD legger til grunn at behandlingen oppfyller kravene i personvernforordningen om riktighet (art. 5.1 d), integritet og konfidensialitet (art. 5.1. f) og sikkerhet (art. 32). For å forsikre dere om at kravene oppfylles, må dere følge interne retningslinjer og/eller rådføre dere med behandlingsansvarlig institusjon.

OPPFØLGING AV PROSJEKTET

NSD vil følge opp ved planlagt avslutning for å avklare om behandlingen av personopplysningene er avsluttet. Lykke til med prosjektet!



NTNU

Norwegian University of
Science and Technology