

TITELBLAD

PROGRAMMERING I GRUNDSKOLEN:

Et casestudie om erfaringer fra pilotprojektet Coding Class og læringspotentialer for
programmering i undervisningen

Navn: Pernille Risør Elving

Studienummer: 20151713

Uddannelsessted: Aalborg Universitet

Uddannelse: It, Læring og Organisatorisk Omstilling

Opgavetype: Kandidatspeciale

Afleveringsdato: 31. Maj, 2017

Opgavetype: A

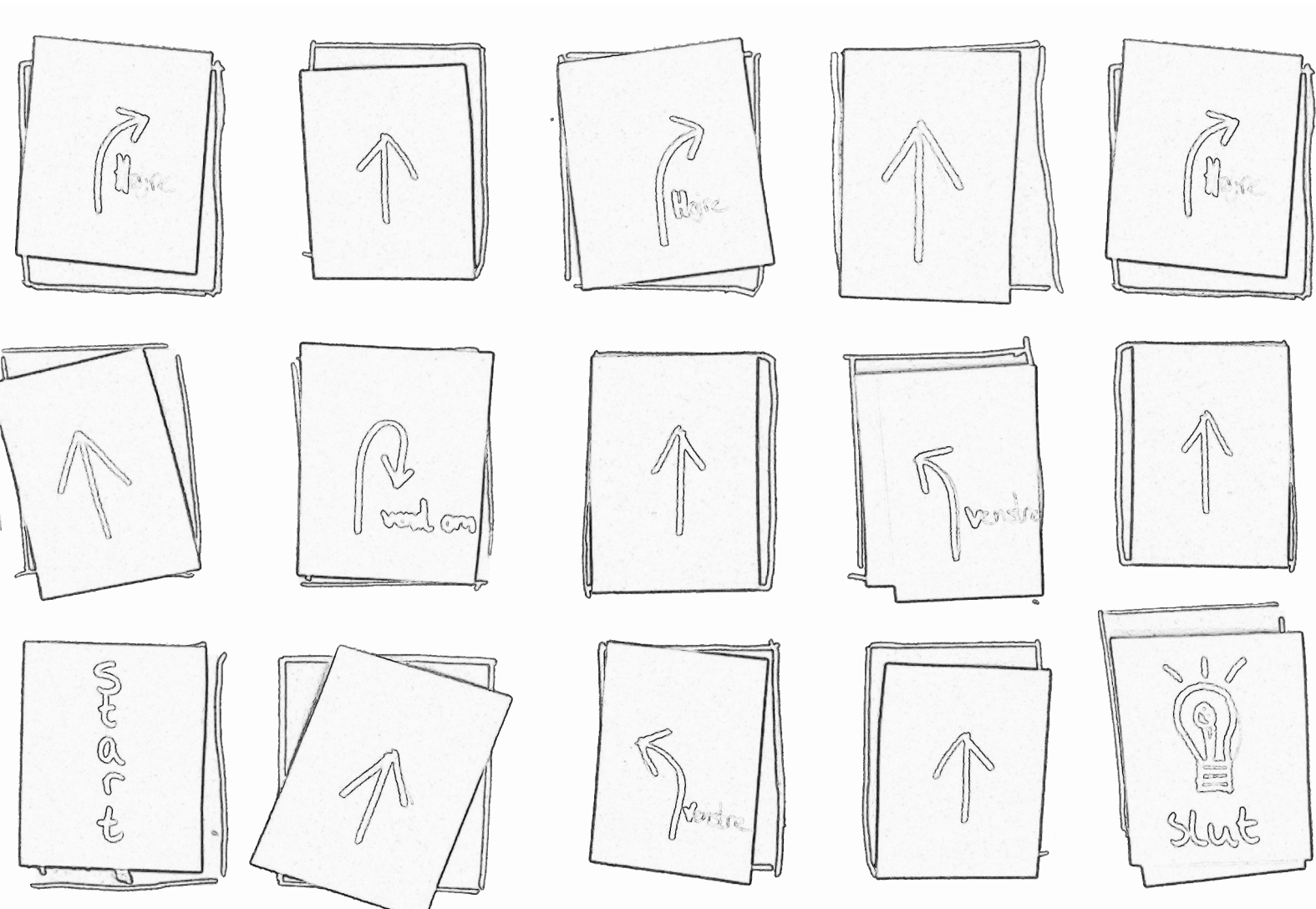
Vejleder: Thomas Ryberg

Anslag: 188.952

Normalsider: 78,7

PROGRAMMERING I GRUNDSKOLEN

Et casestudie om erfaringer fra pilotprojektet Coding Class og læringspotentialer
for programmering i undervisningen.



...

SPECIALEAFHANDLING AF PERNILLE RISØR ELVING

KANDIDATUDDANNELSE I IT, LÆRING & ORGANISATORISK OMSTILLING

VEJLEDER THOMAS RYBERG

AALBORG UNIVERSITET

JUNI 2017

ABSTRACT

This master thesis studies the subject-specific relevance and learning potential of implementing programming in the Danish primary school based on a case study of Coding Class. The objective of the case study is to explore the learning situation during Coding Class, how teachers assess the subject-specific relevance of programming in Coding Class and to discuss the potentials and challenges of implementing programming in the Danish primary school.

The research area in which this thesis operates is influenced by different findings on the subject-specific benefits when implementing programming in primary school teaching, both nationally and internationally. The aim of the study is therefore to gain further insights into which factors influence the learning situation and gain more information on how programming can gain further subject-specific relevance in the primary school. Based on the pragmatic paradigm, the study provides a basis for changing how programming is implemented in the Danish primary school through the case study of three schools using Coding Class as part of the teaching. The case study is conducted through ethnographic field studies to gain insight into the joint context and the individual circumstances of the three cases. This includes observations throughout an entire day of Coding Class in each school followed by interviews with the involved teachers. Furthermore, informal interviews have been conducted with both the project manager of Coding Class, the teaching consultants in Coding Class and involved teachers at each school. The study provides suggestions for teachers on how to make informed decisions when implementing programming as part of their teaching with the knowledge of how these decisions can influence the learning potential.

Findings

Through the empirical investigations in the case study it became apparent that several factors influence the subject-specific relevance of using programming in public school teaching. The study uses constructionism as a learning theory to understand and analyse how the use of programming has influenced the learning situation in Coding Class. The analysis shows that students are motivated, collaborative and problem solving, but that the subject-specific benefits are highly impacted by the groups' organisation and distribution of assignments. The analysis furthermore shows that teachers experience a high degree of student engagement, but lack control of the students' learning outcome in relation to the subject-specific goals. Furthermore, teachers doubt their own competencies and express a need for continuing education in order to ensure fulfilment of the subject-specific requirements when incorporating programming in their teaching. It is discussed which challenges are present when teachers incorporate programming into their teaching. These challenges are composed as concrete tensions for teachers to consider, which include whether programming should meet subject-specific goals or enable development of general skills and let the students' learning process be influenced by

their own interests, and whether the teacher should strive to be an expert or a pedagogical guide to students' workflow and learning process. Moreover, it is discussed how programming should be implemented in a meaningful way in the Danish primary school, including whether programming should be part of existing courses or become a separate course. It became apparent that teachers need practical experience and continuing education to implement programming as part of existing courses where the fulfilment of subject-specific goals are essential. Therefore, the thesis does not provide generalisations on how programming should be included in the Danish primary school, but provides insights on a pragmatic basis on which considerations should be made to enhance the subject-specific relevance when doing so.

Keywords: Programming, technology in use, computer science, problem solving, computational thinking, constructionism, primary school, K-12, Coding Class.

INDHOLDSFORTEGNELSE

KAPITEL 1: INDLEDNING	1
1.1 GRUNDSKOLEN I FREMTIDENS DANMARK	2
1.1.1 PROGRAMMERING I GRUNDSKOLEN	3
1.1.2 UNDERSØGELSESPØRGSMÅL	4
1.2 AFGRÆNSNING	4
1.3 BEGREBSAFKLARING	5
1.4 UNDERSØGELSENS OPBYGNING	6
KAPITEL 2: FORSKNINGSOMRÅDE	8
2.1 FORSKNINGSOMRÅDET KORT FORTALT	9
2.1.1 FORSKNING I PROGRAMMERING I GRUNDSKOLEN	9
2.2 UNDERSØGELSENS PLACERING I FORSKNINGSOMRÅDET	12
KAPITEL 3: FORSKNINGSDESIGN	13
3.1 UNDERSØGELSENS FORSKNINGSDESIGN	14
3.2 UNDERSØGELSESPERSPEKTIV: PRAGMATISME	14
3.3 UNDERSØGELSESTRATEGI: CASESTUDIE	15
3.3.1 CASESTUDIET I DENNE OPGAVE	16
3.4 UNDERSØGELSESMETODER	20
3.4.1 OBSERVATIONER	21
3.4.2 INTERVIEWS	24
3.5 METODEREFLEKSION	26
KAPITEL 4: LÆRINGSTEORETISK PERSPEKTIV	29
4.1 MOTIVATION FOR LÆRING	30
4.2 LÆRING GENNEM KONSTRUKTION	30
4.2.1 LÆRING GENNEM DESIGN	31
4.2.2 KOLLABORATIV LÆRING MED IT	31
4.3 KONSTRUKTIONISTISK UNDERVISNINGSSTRATEGI	32
4.4 KONSTRUKTIONISTISK LÆRINGSMILJØ	33

KAPITEL 5: CODING CLASS I UNDERVISNINGEN	35
5.1 LÆRINGSSITUATIONEN I CODING CLASS	36
5.1.1 KONSTRUKTION OG FORDYBELSE	36
5.1.2 FEJL OG PROBLEMLØSNING	39
5.1.3 FEEDBACK OG SAMARBEJDE	41
5.1.4 FRIHED UNDER ANSVAR	44
5.1.5 DEN GUIDENDE UNDERVISER	46
5.1.6 FAKTORER DER PÅVIRKER LÆRINGSSITUATIONEN	50
5.2 CODING CLASS I LÆRERNES PERSPEKTIV	51
5.2.1 LÆRERERFARINGER FRA CASE 1	51
5.2.2 LÆRERERFARINGER FRA CASE 2	53
5.2.3 LÆRERERFARINGER FRA CASE 3	56
5.2.4 FAKTORER DER PÅVIRKER PROGRAMMERINGS FAGLIGE RELEVANS	58
5.3 DELKONKLUSION: CODING CLASS I UNDERVISNINGEN	61
KAPITEL 6: PROGRAMMERINGS POTENTIALE	62
6.1 SPÆNDINGSFELTER VED PROGRAMMERING I UNDERVISNINGEN	63
6.1.1 FAGSPECIFIKT KONTRA TVÆRFAGLIGT ELEMENT	63
6.1.2 ELEVERNES FAGLIGE UDBYTTÉ KONTRA ELEVERNES INTERESSE	64
6.1.3 SELVVALGTE KONTRA LÆRERVALGTE GRUPPER	65
6.1.4 INDIVIDUELT ARBEJDE KONTRA GRUPPEARBEJDE	65
6.1.5 EKSPERTROLLE KONTRA GUIDENDE UNDERVISER	66
6.1.6 VEJLEDNING TIL INDDRAGELSE AF PROGRAMMERING I UNDERVISNINGEN	67
6.2 INDDRAGELSE AF PROGRAMMERING I GRUNDSKOLEN	69
6.2.1 PROGRAMMERING I EKSISTERENDE FAG ELLER SOM SELVSTÆNDIGT FAG	69
6.2.2 VIDEREUDDANNELSE AF LÆRERNE	72
6.3 DELKONKLUSION: PROGRAMMERINGS POTENTIALE	73
6.4 VIDERE ARBEJDE	74
KAPITEL 7: KONKLUSION	76
7.1 UNDERSØGELSENS RESULTATER	77
LITTERATUR	79

KAPITEL 1

INDLEDNING

...

Indledningsvis præsenteres hvilken kontekst undersøgelsen er opstået fra samt hvilket problemfelt den taler ind i. Herfra præsenteres det overordnede undersøgelsesspørgsmål med tilhørende arbejdsspørgsmål, hvorefter der følger en begrebsafklaring, en afgrænsning af undersøgelsen samt en gennemgang af undersøgelsens opbygning.

1.1 GRUNDSKOLEN I FREMTIDENS DANMARK

Ifølge IT-Branchen oplever hver tredje it-virksomhed i Danmark en mangel på kvalificeret arbejdskraft, fordi efterspørgslen på it-uddannede er større end udbuddet (IT-Branchen, u.å.(b)). I IT-Branchen ses derfor et behov for at dreje børn og unges uddannelse over mod en fremtid inden for informationsteknologi, så deres kompetencer i fremtiden følger den digitale udvikling i Danmark (IT-Branchen, u.å.(b)). Folkeskolereformen fra sommeren 2014 imødekommer den øgede digitalisering i samfundet ved at inkludere it og medier som tværgående tema for undervisningen i grundskolen, da *“it-kompetencer er blevet en naturlig del af såvel den almene dannelse, som de krav, uddannelser og arbejdsliv stiller. Det er således let at argumentere for, at it bør indtage en central plads i folkeskolens undervisning”* (EMU, u.å.(c)). Danmarks Evalueringsinstitut understreger, at der er et stort pædagogisk potentiale i øget anvendelse af it i undervisningen, som kan bidrage til at styrke elevernes faglighed og ruste dem bedre til fremtiden (Danmarks Evalueringsinstitut, 2009). Danmarks Vækstråd estimerer dog i deres rapport om kvalificeret arbejdskraft fra 2016, at arbejdsmarkedet i 2030 vil mangle op mod 19.000 specialister indenfor informations- og kommunikationsbranchen (Danmarks Vækstråd, 2016, s. 81). Efterspørgslen på it-specialister er stigende og samtidig kræver fremtidens arbejdsmarked en bredere kompetencesammensætning, hvor både kreative, tekniske og it-baserede færdigheder og evner er i fokus (Danmarks Vækstråd, 2016, s. 25). Danmarks Vækstråd anbefaler derfor, at computational thinking¹ introduceres tidligt for børn og unge for at kunne møde denne udvikling:

Der bør sættes ind på et tidligt stadie for at vække især børn og unges interesse for „Computational thinking“, hvilket på sigt kan være med til at imødekomme virksomhedernes voksende rekrutteringsbehov og dermed bidrage til at forbedre landets produktivitet og konkurrenceevne (Danmarks Vækstråd, 2016, s. 25).

For at imødekomme efterspørgslen på medarbejdere med teknologisk viden og erfaring kræver det ifølge Danmarks Vækstråd, at der stilles nye krav til vores uddannelsessystem, som skal tilpasses den fremtid, arbejdsmarkedet står over for (Danmarks Vækstråd, 2016, s. 82). En ændring i uddannelsessystemet og elevernes aktiviteter i grundskolen kan potentielt løse denne problematik:

Ved at styre de unges aktiviteter afgør samfundet ved at afgøre de unges fremtid også sin egen fremtid. Eftersom de unge på et senere tidspunkt vil udgøre det fremtidige samfund, vil naturen af dette samfund i høj grad være afhængig af den styring, børnenes aktiviteter får i deres barndom og ungdom (Dewey, 1980 [1912], s. 61).

¹ Defineres under 1.3 Begrebsafklaring

Som Dewey påpeger, påvirkes samfundet i fremtiden af de aktiviteter, som børn og unge indgår i, hvorfor en ændring af deres aktiviteter med it i grundskolen potentielt kan imødegå den digitale udvikling i samfundet. En ændring af elevernes aktiviteter opfattes ligeledes nødvendig, da *“samfundet forandrer sig hastigt, og det er i den virkelighed, at skolen skal finde sin vej og navigere i de nye formelle og uformelle krav fra samfundet”* (EMU, u.å(a)). Grundet de ovenstående anbefalingerne fra Danmarks Vækstråd vil en større opmærksomhed på programmering og computational thinking være det rette sted at starte. Folkeskolereformens ændringer for fagene matematik og fysik/kemi har især indrettet sig efter dette behov, da netop programmering fremhæves som et konkret kompetencemål i fagene. I fysik/kemi er fagenes læseplaner, vejledninger og mål blevet revurderet, så de nu kan inddrage digital styring og programmering, hvilket afspejler den *“stigende samfundsopmærksomhed på programmeringskompetencer”* (EMU, u.å.(b)). I matematik fremhæves programmering som et værktøj i faget, der *“kan understøtte matematiklæring i folkeskolen”* (EMU, u.å.(c)). Computational thinking og programmering er således ikke blevet en integreret del af fagene, men benyttes som redskaber i undervisningen. Dette har givet nogle nye muligheder for undervisning og læring med brug af programmering i undervisningen, men det er ikke tydeligt *hvordan* programmering kan bidrage til undervisningen, og *hvorvidt* inddragelsen af programmering har faglig relevans for den konkrete undervisningspraksis.

1.1.1 PROGRAMMERING I GRUNDSKOLEN

Den øgede opmærksomhed omkring it i uddannelsessystemet sås også under Folkemødet d. 16. juni 2016 omhandlende Danmarks it-parathed. Her blev blandt andet debatteret, hvorvidt børn skal lære at programmere, og mere specifikt om kodning skal på skoleskemaet i grundskolen (IT-Branchen, 2016). I debatten var der stor enighed blandt politikere, forskere, lærere, frivillige samt branchefolk om, at it og programmering skal integreres i den danske grundskole og være en del af den moderne dannelse (IT-Branchen, 2016). Panelet var ligeledes enige om at undervisningen i programmering skal omhandle, hvordan og hvorfor teknologi virker, samt at eleverne lærer at skabe med teknologi. Der var dog uenighed, om hvorvidt programmeringen skulle indgå i grundskolens undervisning. Flere argumenterede for, at programmering er en del af de digitale grundkompetencer i det 21. århundrede og derfor bør indgå som selvstændigt fag på linje med dansk og matematik, hvor andre talte for, at it bør indgå som integreret del af de eksisterende fag (IT-Branchen, 2016). Den aktuelle debat omhandler således, programmerings relevans i grundskolen, hvilket IT-Branchen har givet et konkret bud på i pilotprojektet Coding Class, som er igangsat fra skoleåret 2016 (IT-Branchen, 2017(b)). Visionen bag Coding Class er *“at få børn og unge til at tænke på teknologi og it – og at gøre it til en del af den almene dannelse i folkeskolen”* (IT-Branchen, u.å.(a)). Med Coding Class ønsker IT-Branchen dermed at skabe interesse hos eleverne for at arbejde innovativt med it, og at elevernes it-

kompetencer både kan gavne Danmark som helhed samt elevernes egen fremtid (IT-Branchen, u.å.(a)). På baggrund heraf er det relevant at udføre en kritisk undersøgelse af, hvorvidt og hvorfor grundskolen skal tilpasse sig efterspørgslen på arbejdsmarkedet, men på nuværende tidspunkt er det presserende at følge de allerede igangsatte initiativer for at sikre den faglige relevans ved inddragelse af programmering i grundskolen. En undersøgelse af programmering som fagligt element i undervisningen ved pilotprojektet Coding Class kan på denne måde bidrage til debatten om, hvordan programmering kan indgå meningsfuldt i grundskolen fremadrettet.

1.1.2 UNDERSØGELSESPØRGSMÅL

Med udgangspunkt i ovenstående problemfelt bidrager undersøgelsen til debatten om inddragelse af programmering i undervisningen i den danske grundskole ud fra følgende undersøgelsesspørgsmål:

HVORDAN KAN PROGRAMMERING INDGÅ SOM ET FAGLIGT ELEMENT I GRUNDSKOLENS
UNDERVISNING MED AFSÆT I ERFARINGER FRA CODING CLASS?

For at besvare undersøgelsesspørgsmålet tages udgangspunkt i et casestudie af tre folkeskolars involvering i Coding Class og deres erfaringer med at inddrage programmering i undervisningen. Undersøgelsen udføres med udgangspunkt i følgende arbejdsopgaver:

- ❖ Hvilken læringssituation er til stede under Coding Class-forløbene i praksis?
- ❖ Hvordan forholder lærerne sig til Coding Class samt vurderer programmeringens faglige relevans for deres undervisning?
- ❖ Hvilke udfordringer fremkommer ved at inddrage programmering i undervisningen?
- ❖ Hvilket potentiale er der i at inddrage programmering i henholdsvis eksisterende fag eller som selvstændigt fag i grundskolen?

1.2 AFGRÆNSNING

Undersøgelsen fokuserer på den faglige relevans ved at belyse den aktuelle læringssituation under Coding Class samt lærernes erfaringer hermed og afholder sig derfor fra at vurdere de enkelte elevers faglige og personlige udbytte. Derved fravælges at udføre formelle interviews med eleverne, da undersøgelsen ikke har til hensigt at kortlægge elevernes digitale kompetencer eller oplevelser med at programmere. Samtidig fravælger undersøgelsen ligeledes at belyse elevernes digitale dannelse, som

omhandler individets evner i brugen af digitale værktøjer og digitale kompetencer (Levinsen, 2011, s. 21; Martin & Grudziecki, 2006, s. 253-255). Undersøgelsen belyser udelukkende lærernes vurdering af programmeringens faglige relevans og forholder sig ikke til fagfaglige aspekter af undervisningen herunder programmeringens indfrielse af Undervisningsministeriets formelle krav til fagene. Med afsæt i pragmatismen har undersøgelsen til formål at lære fra erfaringerne fra Coding Class for at guide fremtidig arbejde med programmering, og har således ikke et transformativt mål (Mackenzie & Knipe, 2006) om at ændre eller reformere politiske dagsordener. Ligeledes fravælges derfor at udføre en kritisk undersøgelse af præmisserne for Coding Class som indfrielse af erhvervslivets behov.

1.3 BEGREBSAFKLARING

I løbet af undersøgelsen benyttes betegnelsen *programmering*, hvorfor der kort klarlægges, hvad brugen af ordet indebærer. Programmering dækker over de aktiviteter, som indgår i Coding Class-forløbene såsom spiludvikling i Scratch. Under Folkemødet omkring Danmarks it-parathed (IT-Branchen, 2016), samt IT-Branchens høring d. 1. februar 2017 om et it-fag i grundskolen (Bilag 2), blev der blandt andet brugt følgende udtryk for at dække over aktiviteterne og målene med et eventuelt it-fag samt Coding Class-forløbene: *skabe med it, it-kreativitet, it-fag, kreativ leg med koder, computational thinking, informatik, 21. århundredes kompetencer*. Det er derfor ikke oplagt, hvilken betegnelse, der skal benyttes om it, programmering, kreativitet og problemløsning i grundskolen. Selvom Coding Class indebærer ordet kodning, er formålet med forløbet, at børn skal lære “*at være kreative og skabende med it*” (IT-Branchen, u.å.(a)). Aktiviteterne befinder sig derfor i et spændingsfelt mellem den tekniske kodning samt design og udførelse af kreative projekter med brug af computerteknologi. Her kan programmering udgøre en digital grundkompetence uden teknisk betydning, hvorfor brug af ordet *kodning* til at dække over aktiviteterne og egenskaberne er mangelfuld. Programmering som digital grundkompetence kan rammesættes yderligere gennem begrebet computational thinking. Ifølge Wing dækker betegnelsen computational thinking over vores evner til problemløsning, kommunikation samt interaktion med andre mennesker. Computational thinking er ifølge hende en evne, der ikke afhænger af, om man kan programmere eller få computeren til at udføre en opgave, men derimod omhandler en persons måde at håndtere abstrakte og konceptuel tænkning: “*Thinking like a computer scientist means more than being able to program a computer. It requires thinking at multiple levels of abstraction*” (Wing, 2006, s. 34). Computational thinking dækker således over kompetencer, der ikke nødvendigvis kræver tekniske evner, men som er en måde at tænke på, der sætter os i stand til at agere hensigtsmæssigt med computere. Hensigten med at bruge programmering i grundskolen har således den effekt, at “*når vi lader eleverne være kreativt skabende med programmering på mange forskellige måder, giver vi dem mulighed for at opnå en forståelse for den tænkning, som ligger bag al programmering*” (Erkman & Petropouleas, 2017, s. 12). På denne

måde er der fokus på digitale grundkompetencer, som kan benyttes på tværs af fagligheder, hvilket i denne opgave udgør forståelsen for og brugen af ordet *programmering*.

1.4 UNDERSØGELSENS OPBYGNING

Herunder redegøres kort for, hvorledes undersøgelsen er opbygget, struktureret samt hvilken funktion undersøgelsens individuelle kapitler har.

Kapitel 1

Indledningen har til formål at præsentere hvilken kontekst undersøgelsen er opstået fra og hvilket problemfelt den taler ind i. Herefter præsenteres undersøgelsesspørgsmålet samt gives konkrete spørgsmål til udarbejdelse af undersøgelsens besvarelse heraf, hvorefter undersøgelsens fokus afgrænses kort.

Kapitel 2

Undersøgelsen rammesættes i en præsentation af det forskningsområde, hvori undersøgelsen befinder sig. Der gennemgås eksempler på forskellige forskningsperspektiver og resultater fra konkrete undersøgelser samt redegøres for undersøgelsens berettigelse i forskningsområdet.

Kapitel 3

Herfra følger en gennemgang af undersøgelsens forskningsdesign, hvor der redegøres for undersøgelsen brug af pragmatismen som undersøgelsesperspektiv samt casestudiet som overordnet undersøgelsesstrategi. Det konkrete casestudie præsenteres, hvorefter der redegøres for de konkrete undersøgelsesmetoder samt reflekteres over undersøgelsens opbygning og metodebrug.

Kapitel 4

Undersøgelsens teoretiske afsnit består af en redegørelse for konstruktionismen som læringsteori med afsæt i pragmatismens menneske- og videnssyn. Herefter følger en udledning af overordnede faktorer for et godt konstruktionistisk læringsmiljø, som vil benyttes som analyseapparat i undersøgelsens Kapitel 5.

Kapitel 5

Undersøgelsens analyse er opdelt i to overordnede dele. Første del har til hensigt at få indsigt i læringssituationen under Coding Class-forløbene i praksis, hvor konstruktionismen benyttes som analyseapparat. Anden del af analysen har til hensigt at afdække, hvilken faglig relevans lærere oplever ved at bruge programmering i deres undervisning.

Kapitel 6

Diskussionen har som pragmatisk udgangspunkt at bidrage med pointer, der kan forbedre arbejde med programmering i grundskolen fremadrettet på baggrund af erfaringerne fra Coding Class. Indledningsvis diskuteres spændingsfelter, der påvirker inddragelsen af programmering i undervisningen, hvilket leder til konkrete overvejelser til læreres undervisningspraksis. Herefter diskuteres potentialer og udfordringer i at inddrage programmering i eksisterende fag og som selvstændigt fag i grundskolen. Diskussionen afsluttes med konkrete forslag for videre undersøgelser, der kan gavne debatten og det videre arbejde med at inddrage programmering i grundskolen.

Kapitel 7

Afslutningsvis fremføres undersøgelsens konklusion, hvor erfaringerne fra casestudiet af Coding Class-forløbet bidrager til at besvare hvordan inddragelsen af programmering kan indgå som fagligt element i grundskolens undervisning.

KAPITEL 2

FORSKNINGSOMRÅDE

...

Dette kapitel præsenterer det forskningsområde, som undersøgelsen befinder sig inden for. Der redegøres for forskellige undersøgelsesperspektiver inden for forskningsområdet samt gives eksempler på konkrete undersøgelser og forskningsresultater, hvilket fører til en begrundelse for undersøgelsens relevans inden for feltet.

2.1 FORSKNINGSMRÅDET KORT FORTALT

Denne undersøgelse tager udgangspunkt i grundskolen som domæne, hvor brug af programmering i undervisningen konkret adresseres. Dette forskningsområde har eksisteret siden 1970'erne, hvor blandt andet Papert grundlagde konstruktionismen, som inddrog programmering og brug af teknologi i undervisningen som redskab til elevernes læring (Resnick, 2012). Siden er der opstået større opmærksomhed på området, og udviklingen har affødt nye undersøgelsesperspektiver inden for brug og implementering af it og programmering i grundskolen. Indenfor forskningsområdet omhandler flere undersøgelser eksempelvis elevernes fagspecifikke udbytte især inden for naturvidenskabelige fag (Misfeldt, 2013; Ejsing-Duun & Misfeldt, 2015; Foerster, 2016; Lopez, González & Cano, 2016). I disse undersøgelser opfattes programmering som et middel til at opnå faglige mål, der ses dog ligeledes undersøgelser omhandlende programmeringsundervisningens understøttelse af generelle kompetencer (Lu & Fletcher, 2009; Denner, Werner & Ortiz, 2012; Zaharija, Mladenovic & Boljat, 2013; Kalelioğlu & Gülbahar, 2014; Kalelioğlu, 2015; Nielsen, Pedersen & Majgaard, 2015). I forlængelse af dette undersøgelsesperspektiv findes ligeledes undersøgelser foretaget uden for grundskolens domæne, hvor udvikling af børn og unges humanistiske eller overordnede kompetencer ved brug af programmering undersøges (Smith & Grant, 2000; Rode, Weibert, Marchall, Aal, Rekowski, Mimoni, & Booker, 2015). På tværs af de forskellige retninger afdækkes ofte elevernes fagspecifikke udbytte og læringspotentialer i undervisningen. Forskningsområdet beskæftiger sig således i høj grad med et fokus på, hvorvidt og i hvilket omfang eleverne lærer, og ligeledes it-undervisnings indflydelse på elevernes læring og valg af uddannelse (Mathiasen & de Wit, 2010). Flere undersøgelser belyser ligeledes, hvordan lærerne skal uddannes til at undervise eleverne i it og programmering (Woo, 2016; Wastiau, Blamire, Kearney, Quittre, Van de Gaer & Monseur, 2013). Sidstnævnte retning inden for forskningsfeltet undersøger derfor ikke programmeringen og it i forhold til elevernes udbytte men i højere grad på lærernes udbytte og kompetenceudvikling.

2.1.1 FORSKNING I PROGRAMMERING I GRUNDSKOLEN

Herunder gennemgås et udsnit af undersøgelser, der belyser programmering som fagligt element i grundskolens undervisningen. Der redegøres både for undersøgelser, der hovedsageligt har fokus på elevernes faglige udbytte samt udvikling af mere generelle kompetencer såsom problemløsning, logisk tænkning og kreativ udfoldelse.

Ejsing-Duun & Misfeldt (2015) undersøger, hvordan programmering af robotenheder kan benyttes i matematikundervisningen for at understøtte elevernes læring. Gennem et casestudie undersøges samspillet mellem matematik og programmering i en sjette og tiende klasses programmering af LEGO Mindstorms i matematik. Herigennem belyser de elevernes evne til at tænke

i algoritmer, bruge matematik til at producere viden og konkrete artefakter samt til at tænke abstrakt (Ejsing-Duun & Misfeldt, 2015). De konkluderer, at robotter fungerer meningsfuldt i undervisningssituationer, især når robotten udgør et artefakt til at opnå viden eller undersøge et fagligt emne med (Ejsing-Duun & Misfeldt, 2015, s. 17). Ligeledes påpeger de dog, at det er vanskeligt for læreren at knytte robotrelaterede aktiviteter med fagets læringsmål (Ejsing-Duun & Misfeldt, 2015, s. 18). Denne undersøgelse belyser således programmeringens bidrag som middel til at understøtte matematikundervisningen samt de faglige udfordringer for lærerne.

Foerster (2016) undersøger ligeledes programmerings bidrag til matematikundervisningen med udgangspunkt i indarbejdelse af algoritmer og programmering i matematikfagets pensum om geometri i sjette og syvende klasser i Tyskland. Her undersøges, hvorvidt programmering kan benyttes som værktøj til at udvikle eleverne stringente tænkning, konceptforståelse af eksempelvis variabler og funktioner, problemløsning og italesættelse af matematiske problemstillinger (Foerster, 2016, s. 91). Efter tre år evaluerede Foerster på en kontrolgruppe og en gruppe af elever, der havde lært om programmering for at vurdere langsigtede fordele og effekter af programmering som del af matematikfaget. Undersøgelsen viste, at eleverne der havde lært om geometri ved hjælp af programmering efter tre år havde en bedre forståelse for emnet. Undersøgelsen giver således indsigt i de faglige fordele ved at benytte programmering i matematikundervisningen.

Lopez, González og Cano (2016) undersøger ligeledes elevernes matematiske læring i forbindelse med programmering i undervisningen med udgangspunkt i brugen af Scratch i grundskolen, hvor 107 elever fordelt i femte og sjette klasser på fem skoler blev iagttaget over to år (López, González & Cano, 2016). De evaluerede på, hvilke faglige læringselementer eleverne blev tilført i arbejdet med Scratch, hvor der var en positiv indflydelse på elevernes beregningsmetoder, deres logiske tænkning samt praktiske anvendelse af tekniske kompetencer (López, González & Cano, 2016). De oplevede en øget motivation og engagement blandt eleverne, hvilket de understregede vigtigheden af, da engagementet havde stor indflydelse på den aktive læring, som Scratch understøtter (López, González & Cano, 2016, s. 141). Undersøgelsen giver således indsigt i programmerings bidrag til matematikfaget samt sætter fokus på elevernes engagement som afgørende faktor for deres fagspecifikke udbytte.

Nielsen, Pedersen og Majgaard (2015) undersøger gennem observationer og spørgeskema, hvordan kreative programmeringsopgaver med interaktive robotartefakter med Arduino kan berige undervisningen i en dansk 8. klasse. I forløbets projektfaser blev eleverne introduceret til grundlæggende elementer for programmering og teknologi samt indgik i problemløsning og kreative projekter. Undersøgelsen viser, at programmeringen i høj grad berigede undervisningen både fagligt og personligt, og at eleverne var engagerede i opgaverne (Nielsen, Pedersen & Majgaard, 2015, s. 22). De erfarede ligeledes, at intuitive lærermidler og praktisk erfaring er essentielt for elevernes læring, når de skal eksperimentere og udtrykke sig kreativt med teknologi (Nielsen, Pedersen & Majgaard,

2015, s. 22). Derudover indsamlede de skriftlige refleksioner fra lærerne, som bidrog til et lærerfagligt perspektiv på elevernes arbejde, som viste, at projektforsløbet især opfyldte læringsmålene i natur/teknik, matematik og engelsk, samt at eleverne udviste stor motivation og selvstændighed (Nielsen, Pedersen & Majgaard, 2015, s. 16). Lærerne fremhævede dog ligeledes, at det krævede en særlig indsats at inddrage programmeringen i de specifikke fag. Nielsen, Pedersen og Majgaard fremhæver ligeledes, at elevernes motivation var høj i arbejdet med forskellige fagområder, når det hjalp dem til at løse konkrete problemstillinger. Denne undersøgelse bidrager således med erfaringer omkring udvikling af elevernes problemløsning, teknisk kunnen, kreative udfoldelse og faglige engagement i forbindelse med programmering i især naturvidenskabelige fag. Den bidrager ligeledes med indsigt i lærerens udfordringer ved at opnå relevans for programmeringen i forhold til fagenes læringsmål.

Zaharija, Mladenovic og Boljat (2013) belyser hvorledes introduktionen af programmering til elever i grundskolen kan bidrage til at udvikle deres evner indenfor problemløsning samt logisk tænkning. Undersøgelsen involverede 30 elever fra første klasse i grundskolen, hvor programmering og koncepterne bag blev introduceret gennem fire faser med leg med robotter (Zaharija, Mladenovic & Boljat, 2013, s. 1580). Faserne præsenterer gradvist eleverne for programmering, hvor de indledningsvis introduceres til basiselementer i programmering og på sigt får konkret erfaring med at programmere (Zaharija et al., 2013, s. 1582). Undersøgelsens resultater viser, at eleverne reagerede positivt på læreprocessen, og at deres evner indenfor problemløsning og logisk tænkning blev udviklet og udfordret (Zaharija et al., 2013, s. 1583). De erfarede ligeledes, at eleverne løste opgaverne korrekt og endda på den mest effektive måde, selvom de ikke var blevet bedt om dette. Undersøgelsen bruger leg til at inddrage programmering i undervisningen, hvilket har en positiv effekt for elevernes læreproces (Zaharija et al., 2013, s. 1583).

Kalelioğlu (2015) undersøger udviklingen af grundskoleelevers refleksive evner og problemløsning gennem et forløb med undervisning i Code.org. Studiet tog udgangspunkt i 32 grundskoleelever, hvis problemløsningsevner blev testet før og efter forløbet, hvilket viste, at forløbet med undervisningen ikke påvirkede elevernes problemløsningsevner. Studiet viste dog, at pigernes problemløsningsevner udviklede sig en smule mere end drengenes gennem forløbet (Kalelioğlu, 2015, s. 200). Fokusgruppeinterviews med 10 elever gav indsigt i deres erfaringer med forløbet og viste en høj interesse for programmeringsundervisningen, der dog blev beskrevet som meget udfordrende (Kalelioğlu, 2015). I dette studie blev lærerens refleksioner ligeledes bragt i spil, hvilket i dette tilfælde var en it-lærer. Læreren havde gode erfaringer med brugen af Code.org, både i forhold til de praktiske fordele som lærer og ligeledes i muligheden for, at eleverne kunne udvikle sig i deres eget tempo (Kalelioğlu, 2015, s. 208). Læreren oplevede et højt engagement fra eleverne, som blev forstærket ved belønninger gennem forløbet.

Kalelioğlu & Gülbahar (2014) præsenterer lignende resultater i undersøgelsen af 49 grundskoleelevers brug af Scratch i relation til udvikling af deres problemløsningsevner. Her fandt de, at der ikke var markante ændringer i deres problemløsningsevner, men at eleverne til gengæld blev mere selvstændige omkring deres evner for problemløsning (Kalelioğlu & Gülbahar, 2014). Studiet peger dog ligeledes på, at eleverne i høj grad var engagerede i programmeringen i Scratch og var motiverede for at forbedre deres problemløsningsevner.

2.2 UNDERSØGELSENS PLACERING I FORSKNINGSSOMRÅDET

Undersøgelserne nævnt ovenfor eksemplificerer, hvorledes programmering kan bidrage i grundskolens undervisning, især i matematikundervisningen, men viser blandede resultater omkring elevernes udbytte. Resultaterne fra Kalelioğlu samt Kalelioğlu og Gülbahar viser ikke en markant forbedring i elevernes refleksive og problemløsningsevner, men dette gør både undersøgelsen fra Nielsen et al. og Zaharija et al. Elevernes engagement fremføres i undersøgelsen af Lopez et al. som en afgørende faktor for læringsudbyttet, hvorfor dette er relevant at tage højde for i videre undersøgelser inden for feltet. Ejning-Duun og Misfeldt belyser ligeledes lærernes oplevelse af at inkorporere programmeringen i undervisningen, hvilket synes interessant at undersøge på tværs af fagligheder i stedet for udelukkende i matematikfaget. Både Nielsen et al. og Kalelioğlu inddrager lærerens erfaringer med at benytte programmering som del af undervisningen, hvor lærerne i begge undersøgelser er positive over elevernes udbytte med programmering. I undersøgelsen af Nielsen et al. udtrykker lærerne en udfordring i at inddrage programmering til de fagspecifikke mål, hvilket ligeledes synes interessant at undersøge nærmere.

Ovenstående undersøgelser giver således indblik i forskningsområdet og de tendenser, som i udpræget grad ses i forhold til de videnskabelige perspektiver samt konkrete resultater. Der er især fokus på elevernes udbytte og oplevelse med programmering, hvor elevengagementet er højt på tværs af undersøgelserne. Undersøgelsesresultater viser dog forskel på elevernes udbytte af programmering i undervisningen, hvorfor det fremadrettet er relevant at undersøge hvilke faktorer, der påvirker læringspotentialer i undervisningen samt læringsituationen i praksis. Der ses ligeledes forskel på lærernes oplevelser med inddragelse af programmering i undervisningen, hvorfor det fortsat er relevant at inddrage lærernes vurdering af programmeringens relevans for deres undervisningspraksis. Undersøgelserne ovenfor benytter hovedsageligt casestudier til at få indsigt i den konkrete praksis med programmering i grundskolen, og der er en overvægt af kvalitative undersøgelsesmetoder i form af interviews og observationer. Det vurderes derfor relevant ligeledes at benytte casestudiet i pågældende undersøgelse for at få indsigt i den konkrete undervisningspraksis i Coding Class.

KAPITEL 3

FORSKNINGSDESIGN

...

I dette kapitel beskrives indledningsvis undersøgelsens forskningsdesign, som tager udgangspunkt i pragmatismen som undersøgelsesperspektiv. Herefter følger en beskrivelse af casestudiet som undersøgelsesstrategi og en præsentation af de tre konkrete cases og deres fælles kontekst. Afslutningsvis gennemgås de konkrete undersøgelsesmetoder og foretages en metoderefleksion.

3.1 UNDERSØGELSENS FORSKNINGSDESIGN

Formålet med undersøgelsen er at få indblik i, hvordan programmering kan indgå som fagligt element i undervisningen og herved bidrage til debatten om, hvordan programmering kan indgå meningsfuldt i den danske grundskole fremadrettet. Undersøgelsen tager derfor udgangspunkt i pragmatismen som undersøgelsesperspektiv med den hensigt at undersøgelsens resultater kan bidrage til vejledninger til at skabe forandringer i praksis. For at besvare undersøgelsesspørgsmålet tages udgangspunkt i et casestudie som undersøgelsesstrategi, hvor tre konkrete cases fra Coding Class-forløbet benyttes til at belyse praksiserfaringer med at inddrage programmering i undervisningen i grundskolen. Gennem brug af observationer og interviews i de tre cases opsøger undersøgelsen indsigt i læringssituationen og lærerfaringerne ved inddragelsen af programmering i deres undervisning i Coding Class. Herved opnås kontekstbundet indsigt i, hvordan programmering kan indgå som fagligt element i undervisningen, hvorfor ambitionen med undersøgelsen er at bidrage med nye perspektiver ud fra aktuelle praksiserfaringer, som kan anspore det fremadrettede arbejde på området, fremfor at bidrage med generaliseringer ud fra casestudiet.

3.2 UNDERSØGELSESPERSPEKTIV: PRAGMATISME

Pragmatismen stammer fra Charles Pierce og er blevet videreført af blandt andet William James og John Dewey (Creswell, 2003, s. 11). John Dewey beskrev pragmatismen som håndens epistemologi for at illustrere, hvordan tilegnelsen af viden afhænger af menneskets interaktion med og handling i verden: *“What we experience and know about the world are primarily aspects of things that we interact with and manipulate”* (Dewey i Brinkmann & Tanggaard, 2010, s. 246). Det er altså ifølge Dewey i vores interaktion med verden, at vi kan blive klogere på den og forstå den, og i denne interaktion dannes refleksioner, teorier og erfaring, da vores opfattelse af verden ændres, når vi oplever og handler i den (Brinkmann og Tanggaard, 2010, s. 246). William James beskriver, hvordan sandhed inden for pragmatismen skabes ud fra den konkrete kontekst: *“The pragmatist clings to facts and concreteness, observes truth at its work in particular cases, and generalizes”* (James, 1995, s. 27). Pragmatikeren uddrager derfor elementer, der er til nytte i den konkrete situation, som bliver vedkommendes virkelighed og sandhed. Ifølge Creswell udspringer vores viden ligeledes fra handlinger, situationer og konsekvenser i verden, hvorfor en pragmatisk forskningstilgang griber ind i problemets *hvad* og *hvordan* ud fra konteksten, som undersøges (Creswell, 2003, s. 12). Forskeren bør derfor benytte meningsfulde tilgange til at indsamle data frem for at tilskrive sig og lade sig begrænse af specifikke metoder. Indenfor pragmatismen er metodevalget altså ikke afgørende, da metoder vælges ud fra hvilke, der kan give bedst indsigt i og mening for den konkrete problemstilling (Creswell, 2003, s. 11). Eftersom pragmatismens metodevalg afgøres af den aktuelle problemstilling, tages i den aktuelle undersøgelse udgangspunkt i casestudiet som undersøgelsesstrategi med henblik

på at få indsigt i programmeringens faglige relevans for undervisningen i grundskolen gennem konkrete erfaringer fra pilotprojektet Coding Class.

3.3 UNDERSØGELSESTRATEGI: CASESTUDIE

I denne undersøgelse benyttes casestudiet som strategi til at belyse brugen af programmering i undervisningen under Coding Class. Casestudiet giver mulighed for, at *“explore and investigate contemporary real-life phenomenon through detailed contextual analysis of a limited number of events or conditions, and their relationships”* (Zainal, 2007, s. 2). Casestudiet kan derved benyttes til at undersøge eksempelvis individer, et projekt eller et fænomen på tæt hold, hvorfor casestudiet i denne undersøgelse anvendes til at tilgå Coding Class-forløbet og tre skolars erfaring hermed.

Stake beskriver en case som et integreret system med grænser og delelementer (Stake, 1995, s. 2). Delene i systemet behøver ikke være velfungerende og kan være irrationelle, men forskeren skal forsøge at redegøre for og forstå casen, hvor aktørernes fortællinger skal forstås uden fordomme og forudindtagede holdninger (Stake, 1995, s. 1)². Casestudiet giver således forskeren mulighed for at forstå casens og dens omstændigheders kompleksitet: *“they present data of real-life situations and they provide better insights into the detailed behaviours of the subjects of interest”* (Zainal, 2007, s. 5). På denne måde kan casestudiet benyttes som instrument til at forstå noget større eller mere generelt end selve casen (Stake, 1995, s. 3). Casestudiet giver her mulighed for at komme i tæt forbindelse med de virkelige situationer, hvilket giver et mere nuanceret syn på virkeligheden omkring casen (Flyvbjerg, 2010, s. 467). Forskeren kan vælge at bruge data fra flere kilder og cases til at forstå undersøgelsesområdet bedre, hvor valget af en typisk case kan give en realistisk indsigt i forskningsspørgsmålet, mens en atypisk case derimod kan illustrere noget, som ofte overses i de typiske cases (Stake, 1995, s. 4). Det handler derfor om at udvælge sine cases med omhu, da formålet med casestudiet ikke er at bidrage med ny viden og alternative perspektiver til forskningsspørgsmålet gennem de specifikke cases frem for at producere flere generaliseringer (Stake, 1995, s. 8). Flyvbjerg understreger ligeledes grundet kritikken om, at casestudier ofte giver større spillerum til forskerens forudfattede meninger, at det ofte er atypiske cases, som giver den mest brugbare information, da det inddrager uventede dele, som forskeren skal redegøre for og derfor ofte må revidere eventuelle teser til (Flyvbjerg, 2010, s. 473).

² Der findes flere teoretiske tilgange til casestudiet som undersøgelsesstrategi, hvor den aktuelle undersøgelse fravælger blandt andet tilgangen fra Robert K. Yin, som anskuer casestudiet som en mere eksperimenterende tilgang (Yin, 2009; Yin, 2011).

3.3.1 CASESTUDIET I DENNE OPGAVE

I denne undersøgelse bruges casestudiet til at forstå aktiviteterne og erfaringerne med at inddrage programmering i undervisningen i forbindelse med pilotprojektet Coding Class. Der indgår tre cases i det aktuelle casestudie, hvor forløbet med Coding Class foregår forskelligt, hvorfor casestudiet er præget af en informationsorienteret udvælgelse med mulighed for at indhente information om proces og resultat på tværs af forskelligartede cases (Flyvbjerg, 2010, s. 475). Casestudiet benyttes således til at søge dybdegående informationer om erfaringerne fra Coding Class. Der indgår derved ikke en bred diversitet i dataene, hvorfor forskeren skal forholde sig objektivt og kritisk til dataindsamlingen, før der kan udledes generelle pointer på baggrund heraf (Zainal, 2007, s. 5).

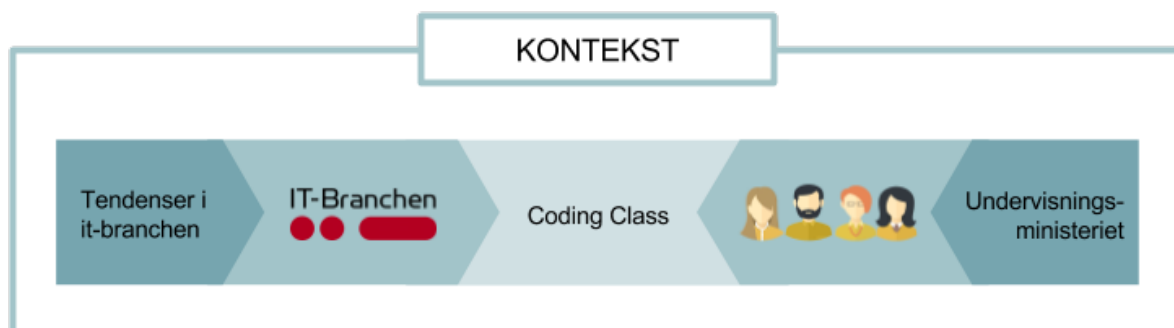
Ifølge Creswell foregår pragmatisk forskning altid indenfor en bestemt kontekst, eksempelvis sociale, historiske eller politiske kontekster, hvorfra der søges at forstå og løse problemet (Creswell, 2012, s. 466). For at forstå erfaringerne fra Coding Class er det således nødvendigt at undersøge konteksten for casestudiet, hvorfor jeg har indgået i dialog med projektlederen for Coding Class i IT-Branchen (Bilag 1) samt deltaget i en høring, hvor IT-Branchen inviterede forskellige politikere, eksperter mm. ind, for at debattere inddragelse af programmering i grundskolen (Bilag 2). Derudover har jeg været i uformel dialog med både lærerne på de enkelte skoler samt konsulenterne fra IT-Branchen undervejs under feltstudiet, hvilket har bidraget til en forståelse for rammerne for Coding Class. Casestudiet i denne undersøgelse er således influeret af en fælles kontekst grundet rammerne for Coding Class, hvor praksiserfaringer og brugen af Coding Class-forløbet har været forskellige i de tre involverede cases. Nedenfor gennemgås derfor først den fælles kontekst, hvorefter hver enkelt case præsenteres.

Konteksten

Coding Class er et resultat af IT-Branchens samarbejde med blandt andet Styrelsen for IT og Læring samt Coding Pirates, hvor en række femte og sjette klasser i henholdsvis Vejle og Københavns Kommune får it på skoleskemaet i et skoleår. Coding Class er igangsat, fordi *“der er brug for, at flere unge får interesse for at være kreative og skabende med it, og et bedre match mellem uddannelser og branchens behov”* (IT-Branchen, u.å.(a)). IT-Branchens mål er således at sikre elevers interesse for feltet og dermed flere it-uddannede i fremtiden, som imødekommer kravene på arbejdsmarkedet inden for it-branchen. IT-Branchen har i foråret 2016 indsamlet 1,5 millioner kroner, som blandt andet er gået til at frikøbe undervisere fra Coding Pirates³, der skal understøtte lærerne i inddragelsen af programmering i undervisningen (IT-Branchen, 2017(b)). De tre cases, der udgør det samlede casestudie, tager udgangspunkt i den samme kontekst, nemlig Coding Class og de rammer, som IT-Branchen har sat op for det. Det er dog lærerne, der i praksis har valgt at deltage i pilotprojektet og skal stå for planlægning og udførelse af forløbet i forbindelse med deres undervisningsforløb i fagene.

³ Coding Pirates er en forening, der arbejder på at forbedre kreative it-kompetencer ved børn mellem 7-17 år gennem kodning som fritidsaktivitet. Se mere på: <https://codingpirates.dk/>.

Lærernes undervisning skal tilrettelægges ud fra Undervisningsministeriets bestemmelser for fagformål, kompetencemål og underliggende færdigheds- og vidensmål (Undervisningsministeriet, 2017(a)). Casestudiets kontekst involverer således flere aktører og elementer, som influerer det konkrete forløb med Coding Class, hvilket illustreres nedenfor:

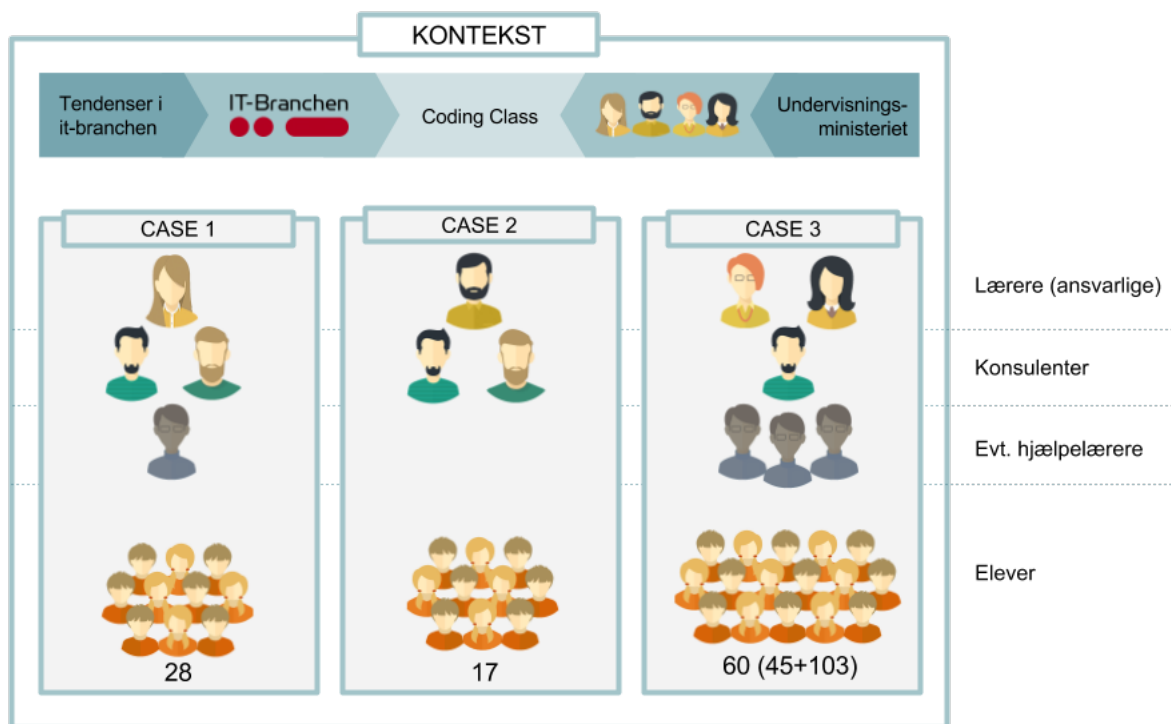


Figur 1: Oversigt over konteksten for casestudiet

Konteksten for casestudiet er derfor især repræsenteret af to forskellige hensigter med Coding Class, nemlig henholdsvis IT-Branchens mål om at sikre elevernes fremtidige interesse for feltet samt folkeskolelærernes mål om at sikre elevernes læring og opfyldelse af de formelle krav fra undervisningsministeriet. Konteksten inkluderer således de politiske bestemmelser fra undervisningsministeriet og tendenser, der ses i it-branchen og arbejdsrapporter omkring udviklingen på arbejdsmarkedet, væksten samt digitaliseringen i Danmark. Der er derfor flere aktører i spil i konteksten for forløbet, men kun IT-Branchen samt lærerne har direkte berøring med forløbet.

Præsentation af de tre cases

Hver skole har fået stillet to konsulenter fra Coding Pirates til rådighed, som har undervist i programmering fire gange i løbet af skoleåret 2016. Det har herfra været de enkelte skoler og læreres egne idéer og tanker, som har skullet forme indholdet i forløbet i dialog med konsulenterne, der kunne støtte implementeringen heraf. Alle skoler har deltaget i intromøder og selv valgt at deltage i forløbet. Forløbet er tiltænkt mellemtrinnet, hvor både femte og sjette klasser har været involveret. Forløbene på skolerne har været udformet og struktureret på forskellig vis. Nedenfor illustreres hver enkelt case med involverede lærere, konsulenter og elever samt den fælles kontekst:



Figur 2: Oversigt over de tre skoler i casestudiet⁴

Case 1: Smidstrup-Skærup Skole

Forløbet med Coding Class på Smidstrup-Skærup Skole foregik over seks uger i danskundervisningen, hvor en sjette klasse blev fordelt i grupper, der skulle fungere som virksomheder. Der var afsat fire dage til, at konsulenterne fra Coding Class kunne assistere i undervisningen, men de arbejdede også videre i de seks uger i danskfaget ved siden af, hvor de fik hjælp fra ældre elever på skolen. Eleverne blev introducerede til programmering før forløbets start, hvor eleverne fik til opgave at udføre basale programmeringsopgaver i Scratch samt opgaver, der introducerede dem til computational thinking. I forløbet med virksomhederne skulle hver virksomhed skabe et spil i Scratch, holde budget og reklamere for både virksomhed og spil, og det kostede penge for virksomhederne at få hjælp fra konsulenterne. **Læreren** der stod for undervisningen var dansklærer og havde ingen erfaring med programmering. Hun hjalp primært eleverne med deres virksomheder. Lærerne har forud for projektets opstart arrangeret en fremvisning af eleverne fra Case 2's programmeringsprojekter, så læreren og eleverne kunne få indsigt i mulighederne med programmeringsarbejdet. **Eleverne**⁵ bestod af tre sammenlagte sjette klasser. Læreren inddelte de 28 elever i grupper á fire, hvor de selv måtte fordele rollerne i virksomheden mellem sig i grupperne. **Konsulenterne** indgik som eksperthjælp og hjalp med den konkrete kodning af elevernes spil i Scratch. **Observationerne** foregik under fjerde gang i forløbet, hvor eleverne skulle færdiggøre deres spil, præsentere dem, få feedback fra konsulenterne og forbedre deres produkter i virksomhederne.

⁴ Vær opmærksom på, at i Case 3 er 60 elever fra sjette årgang involveret i forløbet, men på observationsdagen, markeret i parentes, deltager 45 elever fra sjette årgang og yderligere 103 elever fra fjerde og femte årgang.

⁵ Se en oversigt over eleverne i Case 1 og deres fordeling i grupper i Bilag 3, s. 15.

Case 2: Skibet Skole

Forløbet med Coding Class på Skibet Skole foregik i en femte klasse, hvor eleverne blev opdelt i grupper, der skulle fungere som virksomheder. Forløbet startede efter sommerferien og blev brugt løbende over hele skoleåret, når det passede ind i matematik- og natur/teknik-undervisningen. Her skulle eleverne designe spil, holde budget og reklamere for arbejdet i virksomhederne. Der var sat fire dage af til, at konsulenterne kunne assistere læreren og hvor eleverne i virksomhederne kunne købe sig til deres hjælp. **Læreren** havde ikke tidligere arbejdet med programmering i undervisningen, men havde eksperimenteret med det løbende i undervisningen og sammen med eleverne. **Eleverne**⁶ bestod af en enkelt femte klasse, hvor de 17 elever var inddelt af læreren i grupper á tre til fire personer, hvor de selv var ansvarlige for strukturering af roller i virksomhederne. **Konsulenterne** indgik som ekstra hjælp, hvor de assisterede læreren i den tekniske undervisning med programmering. De introducerede eleverne til forskellige programmer efter lærerens ønsker, hvor de på observationsdagen ikke havde stor erfaring med det brugte program. **Observationerne** foregik den tredje gang i forløbet, hvor eleverne skulle stifte bekendtskab for første gang med AppInventor og lave en app til deres virksomheder.

Case 3: Mølholm Skole

Forløbet med Coding Class var afviklet i efteråret 2016 og tilknyttet undervisningen i matematik. De fire gange blev fordelt over nogle uger, hvor de 60 elever i sjette klasse skulle udvikle spil i Scratch. De havde udelukkende arbejdet med programmering på de fire dage, hvor konsulenterne var til stede. Matematiklærerne havde efter det egentlige forløb med Coding Class planlagt en ekstra dag, hvor sjette årgang skulle undervise fjerde og femte årgang i programmering, på hvilken dag observationerne foregik. **Lærerne** bestod under forløbet af tre matematiklærere, hvor kun den ene havde tilmeldt sig forløbet, og de andre havde indvilget i at deltage. Efter læreren, der havde tilmeldt dem, gik på barselsorlov, havde de to resterende matematiklærere planlagt den ekstra dag, hvor de fik hjælp fra to ekstra lærere til at holde ro og orden. **Eleverne**⁷ bestod af 60 elever fra sjette årgang, som havde været gennem forløbet i efteråret 2016 og selv havde fordelt sig i grupper á cirka fire elever. På observationsdagen var eleverne fra sjette klasse fordelt i deres grupper fra forløbet, hvor hver af disse grupper blev tildelt en gruppe yngre elever fra fjerde og femte årgang. I alt var 148 elever samlet på observationsdagen, hvor 45 elever fra sjette årgang agerede elev-undervisere for de 103 elever fra henholdsvis fjerde og femte klasse. Grupperne blev fordelt i forskellige klasseværelser og møderum for at undgå, at de forstyrrede hinanden. Dagen før observationerne foregik havde eleverne fra sjette klasse haft tid til at planlægge deres undervisning ud fra lærernes anvisning. **Konsulenterne** havde været ansvarlige for planlægningen af indholdet i de fire dage i efteråret. Den

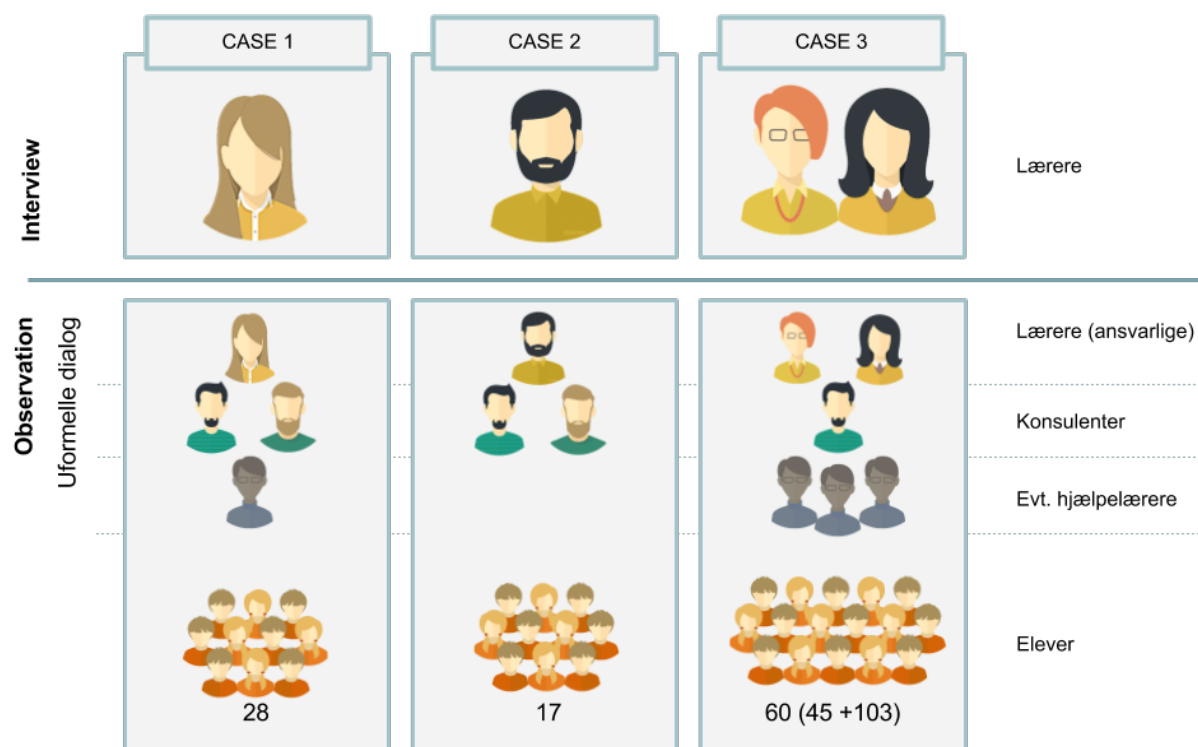
⁶ Se en oversigt over eleverne i Case 2 og deres fordeling i grupper i Bilag 3, s. 16.

⁷ Se en oversigt over eleverne i Case 3 og deres fordeling i grupper i Bilag 3, s. 17-18.

ene konsulent var til stede på observationsdagen for at understøtte eleverne fra sjette klasse i deres undervisning af de yngre elever. **Observationerne** foregik under femte undervisningsgang, som var en ekstra dag i forlængelse af forløbet.

3.4 UNDERSØGELSESMETODER

Ifølge Stake kan casestudier trække på en lang række metoder afhængig af, hvad der er relevant for den enkelte case (Stake, 1995, s. xii). I denne undersøgelse indsamles udelukkende kvalitative data, som har til formål at indsamle “*open-ended, emerging data with the primary intent of developing themes from the data*” (Creswell, 2003, s. 18). I denne undersøgelse anvendes både observationer og semistrukturerede interviews, der har til hensigt at opnå større forståelse og indsigt i det konkrete casestudie og herunder de tre cases. Nedenfor ses en figur over undersøgelsens feltarbejde ved de tre separate cases.



Figur 3: Oversigt over casestudiets feltarbejde.

Under observationsdagene var jeg løbende i dialog med både eleverne, lærerne, konsulenterne samt hjælpelærerne for at fremme indsigten i de involverede aktører og aktiviteter i de enkelte cases samt konteksten for casestudiet. Nedenfor redegøres for udførelsen af observationerne samt interviewene, og hvilket formål de hver især har for besvarelse af undersøgelsesspørgsmålet.

3.4.1 OBSERVATIONER

For at opnå indsigt i praksiserfaringer med Coding Class-forløbene udførtes observationer med det formål at få indsigt i, hvordan aktiviteterne foregår i den specifikke sammenhæng (Raudaskoski, 2012, s. 82). Ved at være til stede når aktiviteterne foregår, sikres at genstandsfeltet bliver forstået og kan behandles korrekt, da forskeren får en dybere forståelse for situationen og de elementer, der gribes ind i (Raudaskoski, 2012, s. 86). Af samme årsag fandt observationerne sted under hele dage, da *“if one’s interest is in the activities at or around a particular location, it is important to observe those activities at various times of the day”* (Blomberg et al., 1993, s. 132). For at kunne forstå de aktiviteter, som foregår i undervisningen samt i interaktionen mellem aktørerne, var det naturligt at følge en hel dags undervisning på hver skole.

Observationsfokus

Observationernes fokus har været at forstå læringssituationen under Coding Class ved at observere dynamikken i klasseværelset, hvilket både indebærer samspil mellem eleverne samt konsulenterne og lærernes interaktion i klasseværelset og med eleverne. For casestudiernes metode fremhæver Stake, at forskeren skal tilgå observationer af casen objektivt, men at der samtidig bør uddrages meninger under observationerne og fortolkes på det erfarede (Stake, 1995, s. 9). Formulerede forskningsspørgsmål skal derfor kunne ændres og modificeres løbende for at følge den bedst mulige forståelse af casen ud fra indsigter, forskeren får undervejs. Undervejs i observationerne viste sig forskelle mellem aktiviteterne på tværs af cases; i Case 1 og 2 var fokus på elevernes produkter, hvorfor observationsfokus inkluderede produktudbytte i disse to cases, mens der under observationsdagen i Case 3 blev fokuseret på elevernes engagement samt arbejdsproces, da eleverne i høj grad blev overladt til dem selv. Da opbygningen af de forskellige forløb har varieret, har det derfor været naturligt at inkludere opbygningen som fokus i observationerne, således der kan opnås en forståelse for aktiviteternes indflydelse på læringssituationen i praksis. Observationernes fokus har dermed taget udgangspunkt i de aflæselige træk i situationen og interaktionen i klasseværelset for at forstå omfanget af det, som foregår i brugen af programmering i undervisningen.

Observatørrolle

I casestudiets observationer foreslår Stake, at man som observatør forholder sig diskret til de aktører og de aktiviteter, som udfolder sig (Stake, 1995, s. 12). Jeg har derfor forholdt mig åbent og afventende til observationerne for at forstå realiteterne og ikke teste eller udfordre situationen. Under observationerne indtog jeg således rollen som en blanding mellem *observerende deltager* og *deltagende observatør* (Blomberg et al., 1993, s. 131; Maaløe, 2002, s. 148), hvorfor jeg befandt mig i

spændet mellem at være til stede fra sidelinjen og deltage aktivt i aktiviteterne. Jeg fulgte aktiviteterne og indgik løbende i dialog med de involverede aktører og tilpassede min interaktion til de konkrete aktiviteter. I klasseværelset forholdt jeg mig hovedsageligt observerende men kommunikerede både med elever og lærer, når det var naturligt. Jeg blev hvert sted introduceret af læreren til eleverne, hvor nogle efterfølgende hilste på mig, mens andre ikke bed mærke i min tilstedeværelse. I lærerværelset indgik jeg som observerende deltager på grænsen til komplet deltager (Maaløe, 2002, s. 147), hvor jeg deltog i lærernes og konsulenternes dialog samt deltog i deres aktiviteter. Ved at være afventende omkring forventningerne til min rolle som observatør opnåede jeg en naturlig rolle i både klasseværelset og lærerværelset, hvorfor jeg undgik at *“jeopardizing one’s position in the community by insisting on observing or participating in an activity, when doing so is deemed by inappropriate by the community, could easily spell a disaster for the project”* (Blomberg et al., 1993, s. 132).

Observationsnoter

For at overskue interaktionen i klasseværelset blev elevernes og gruppernes placering heri tegnet ned og inddelt i henholdsvis tal for den enkelte elev og bogstaver for den pågældende gruppe i observationsnoterne (Bilag 3). Kønnene på eleverne angives med P for pige og D for dreng foran elevens tal⁸, da lærerne i Case 1 og 3 nævner kønsopdelingen i grupperne, hvorfor det vurderes hensigtsmæssigt at indikere rollerne i observationsnoterne. Interaktionen i klasseværelset blev løbende skrevet ned, som aktiviteterne i klassen udfoldede sig og blev uddybet umiddelbart efter observationerne (Bilag 4, 5 og 6). Som deltagerobservatør fokuserede jeg mine observationer på hvem, der gjorde hvad og i relation til hvem, hvor og hvordan det foregik, samt under hvilke omstændigheder det skete (Maaløe, 2002, s. 157). Observationsnoterne var derfor detaljerede og inkluderede både tidsstempler, overordnede observationer om situationen samt mere specifikke observationer af interaktionen aktørerne imellem. Undervejs indgik jeg i dialog med lærerne samt konsulenterne fra Coding Class, som ligeledes blev noteret umiddelbart efter dialogen ophørte. På denne måde refereredes kort dialogen og lærernes udsagn for at bevare en så høj nøjagtighed som muligt. Intentionen med observationsnoterne er således at berette, hvad der observeres i situationen for senere at kunne analysere og fortolke de komplekse situationer uden forudindtagede meninger. Der tages således højde for, at *“mennesker kan have mange grunde til deres handlinger. Vi bør derfor nødig alene ud fra observation tilskrive andre motiver, endsige udtale os om deres indre bevægelse”* (Maaløe, 2002), 164). Observationsnoterne ligger derfor i spændet mellem beskrivelser af, hvad der foregår i situationen samt forklaringer, der har til hensigt at redegøre for den adfærd som iagttages. Observationsnoterne ligner således vignetter, som er narrativer baserede på feltnoter, der både er rige i detaljer men tillader læseren at nå frem til samme konklusioner ud fra det beskrevne (Erickson, 1986).

⁸ I Case 3 markeres elev-underviserne fra sjette årgang yderligere med et sekstal for at gøre forskel på dem i forhold til eleverne, som undervises.

Operationalisering

For at behandle observationerne udførtes en meningskondensering umiddelbart efter deltagelsen i feltet, som indgår indledende i observationsnoterne (Bilag 4, 5 og 6), hvilket havde til formål at give en så akkurat forståelse af situationen som muligt og fremhæve de første refleksioner og fortolkninger af feltet. Efterfølgende blev noterne fra observationerne kodet (Bilag 7, 8 og 9) med prædefinerede teoretiske kategorier, hvilket havde til formål at markere sammenhænge og forskelle i dataene (Brinkmann & Tanggaard, 2010, s. 47). De teoretiske kategorier udgøres af en teoretisk sampling, hvor data bearbejdes gennem begreber og teorier som allerede er udviklet (Brinkmann & Tanggaard, 2010, s. 227), hvilket gøres på baggrund af konstruktionismens læringsteori, som redegøres for i undersøgelsens Kapitel 4. De teoretiske kategorier vil benyttes som analyseapparat til at belyse læringssituationen i hver case og gøre det muligt at *“lede efter steder, personer og hændelser, der vil maksimere muligheder for at opdage variationer”* (Brinkmann & Tanggaard, 2010, s. 227). Den teoridrevne kodning har ligeledes til formål at frigøre mine tanker om observationerne, som udvikledes i og efter deltagelsen i feltet, og således sikre variation i databehandlingen, så nye perspektiver kan udvikles. Den teoretiske sampling har ført til følgende kategorier, som observationsnoterne blev kodet ud fra med følgende farvekode:

- = Konstruktion og fordybelse
- = Frihed under ansvar
- = Fejl og problemløsning
- = Den guidende underviser
- = Feedback og samarbejde

Datakodningen ses i Bilag 7, 8 og 9, hvor følgende er et eksempel på udførelsen af kodningen ud fra de ovenstående farvekoder:

Gruppe E er gået koncentreret i gang med deres produkter og en af D9 er gået i gang med at redigere deres spil. Lærer JB beder dem om at gå i gang med deres virksomhedsmøde. D9 kigger ikke en gang op fra sin computer, men fortsætter med at arbejde på spillet. De andre gruppemedlemmer småsnakker lidt over bordet, men dykker også ned i deres produkt igen. Først da Lærer JB kommer forbi igen og spørger D9 om, hvad han laver og om de er gået i gang med virksomhedsmødet, retter han blikket mod læreren og fortæller at han bare lige mangler at rette noget. Lærer JB fortæller D9 med fast stemme, at de skal gå i gang med virksomhedsmødet nu, hvilket de så gør.

Figur 4: Eksempel på datakodning af observationerne (Bilag 7, s. 73).

3.4.2 INTERVIEWS

For at få indsigt i praksiserfaringer med Coding Class udføres interviews, der giver forskeren mulighed for at indsigte i specifikke individers forståelse af bestemte begivenheder, situationer og fænomener (Brinkmann & Tanggaard, 2015, s. 37), hvor interview benyttes som metode til at få indsigt i erfaringerne med Coding Class. Blomberg gør forskel på det formelle og uformelle interview, hvor de uformelle interviews udgør en forberedelse til de formelle interviews: *“through informal interviews the ethnographer begins to learn enough about the community to conduct more structures, systematic interviews”* (Blomberg et al., 1993, s. 134). De uformelle interviews kan både foretages tidligt i feltstudier, som en metode til at opnå viden omkring problemfeltet, samt under deltagelse i feltet, som kontekstuelle interviews der giver indblik i den konkrete situation (Blomberg et al., 1993, s. 134-135). Der er derfor valgt både at udføre uformelle interviews som del af undersøgelsens forarbejde samt under observationerne og udføre formelle interviews på baggrund af den viden, som opnås i forarbejdet og i deltagelsen under Coding Class. Herunder redegøres for de metodiske overvejelser og valg i henhold til undersøgelsens uformelle og formelle interviews samt empiribearbejdelsen heraf.

Uformelle interviews

De uformelle interviews udgjordes af dialoger med flere aktører og blev afholdt i de indledende faser til undersøgelsen og under undersøgelsens observationer. De uformelle interviews var ustrukturerede og åbne, således aktørerne kunne forme samtalen og jeg kunne *“ask appropriate questions from the point-of-view of the members of the community under study”* (Blomberg et al., 1993, s. 134). De indledende uformelle interviews udgjorde et forarbejde for at forstå konteksten for casestudiet, dette inkluderede uformelle interviews med projektlederen fra Coding Class (Bilag 1) samt konsulenterne fra Coding Class i forbindelse med en høring afholdt af IT-Branchen. De løbende uformelle interviews blev udført med involverede lærere, hjælpelærere, elever samt konsulenterne i den konkrete kontekst under observationerne, som indgår i observationsnoterne (Bilag 4, 5 & 6), hvilket bidrog med større indsigt i de aktuelle cases og aktiviteterne heri. De samlede uformelle interviews bidrog således til en bedre forståelse for casestudiets kontekst, rammerne for Coding Class samt de konkrete undervisningsforløb og aktiviteterne i de forskellige cases.

Formelle interviews

De formelle interviews havde til formål at få indblik i lærernes betydningsverden og brugen af programmering i undervisningen ved hver enkelt case. Interviewene blev foretaget med de hovedansvarlige lærere, hvor der i Case 3 var to hovedansvarlige lærere, som derfor blev interviewet sammen. Interviewene blev opbygget som semistrukturerede interviews for både at opnå forståelse for

casen og indsigt i de temaer, som havde vist sig relevante under de uformelle interviews. På baggrund af de uformelle interviews og første observation, konstrueredes således en interviewguide (Bilag 10), der bestod af åbne interviewspørgsmål struktureret i forhold til temaer og forskningsspørgsmål. Det semistrukturerede interview tilbyder balance mellem at opsøge svar på konkrete forskningsspørgsmål, samt give plads til *“den fortælling, som interviewpersonen er mest optaget af at fortælle”* (Brinkmann & Tanggaard, 2015, s. 37). Interviewspørgsmålene blev således udformet som simple praksisnære spørgsmål, der var letforståelige og nemme at ændre undervejs til den konkrete case og lærernes egen fortælling.

Transskriptionsmetode

De formelle interviews blev lydoptaget⁹ og derefter transskriberet så ordret som muligt (Bilag 11, 12 & 13). I transskriberingen udelades dog fyldord, tøvende udråbsord samt forsøgt opstartede sætninger, hvis interviewpersonen undervejs efterfølger en anden tankestrøm. Ligeledes udelades mine opmuntrende udtryk, der havde til formål at indikere, at jeg lyttede og fulgte med i hvad interviewpersonen ytrede. Da læseren ikke har adgang til kropssprog, mimik og tonefald, blev det prioriteret at få meningen klart frem uden forstyrrelser fra eksempelvis tøvende udråbsord. Der er derfor til tider anført, hvis en interviewperson griner, bruger fagter eller på anden vis udtrykker mening, som ikke udtales aktivt hvilket indikeres i klammer “[]” direkte i transskriberingen.

Operationalisering

De formelle interviews blev bearbejdet ud fra en datadreven kodning, hvor der formuleres kategorier på baggrund af det empiriske datasæt (Brinkmann & Tanggaard, 2010, s. 47). Dette har til formål at fremhæve de fortællinger, som lærerne selv finder presserende, at få indblik i lærernes erfaringer med forløbet og muliggøre sammenligning af de tre forskelligartede cases. Interviewene blev behandlet gennem en kodning (Bilag 14, 15 og 16) ud fra de temaer, som viste sig i gennemlæsningen af interviewene og havde relevans for besvarelse af undersøgelsens overordnede undersøgelsesspørgsmål. Den datadrevne kodning tager udgangspunkt i følgende kategorierne med tilhørende farvekoder:

	= Coding Class-forløbet		= Elevoplevelser
	= Undervisningens opbygning		= Lærereoplevelser
	= Faglig mål og udbytte		= Lærerkompetencer

⁹ Lydoptagelserne af interviewene er ikke vedlagt, idet de alle indgår som transskriberinger i bilag. Hvis det ønskes, kan lydoptagelserne naturligvis fremvises.

Datakodningen ses i Bilag 14, 15 og 16 hvor følgende er et eksempel på udførelsen af kodningen ud fra de ovenstående seks kategorier:

256 SB: Jeg tænker faktisk, at pigerne helt sikkert har været mest tændt på det med
257 markedsføring, og det med at skulle lave merchandise og skulle lave reklamer og logo
258 til virksomhederne. Altså alle de der kreative opgaver. Så der kan man sige, at der
259 kunne man måske godt tænke over og være mere styrende omkring, hvem der har
260 hvilke roller eller sige, at i dag skal I i hvert fald skifte rolle. Det har egentligt stået
261 dem frit for om de har ville skifte roller for hver gang, sådan at rollerne roterede eller
262 man blev i samme roller. Og der tænker jeg, at man måske skal stille krav til at det
263 skal rotere, så pigerne også bliver nødt til at have fingrene i det der Scratch, og at
264 drengene bliver nødt til at vise, at de også har lært noget om reklamer og
265 markedsføring.

Figur 5: Eksempel på datakodning af interviewene (Bilag 14, s. 182).

Kategorierne blev dannet på tværs af de tre interviews, hvorfor nogle af interviewene ikke berører kategorierne i lige stor grad. Der er bevidst valgt at se bort fra forekommende temaer, som ikke har relevans for undersøgelsesspørgsmålet.

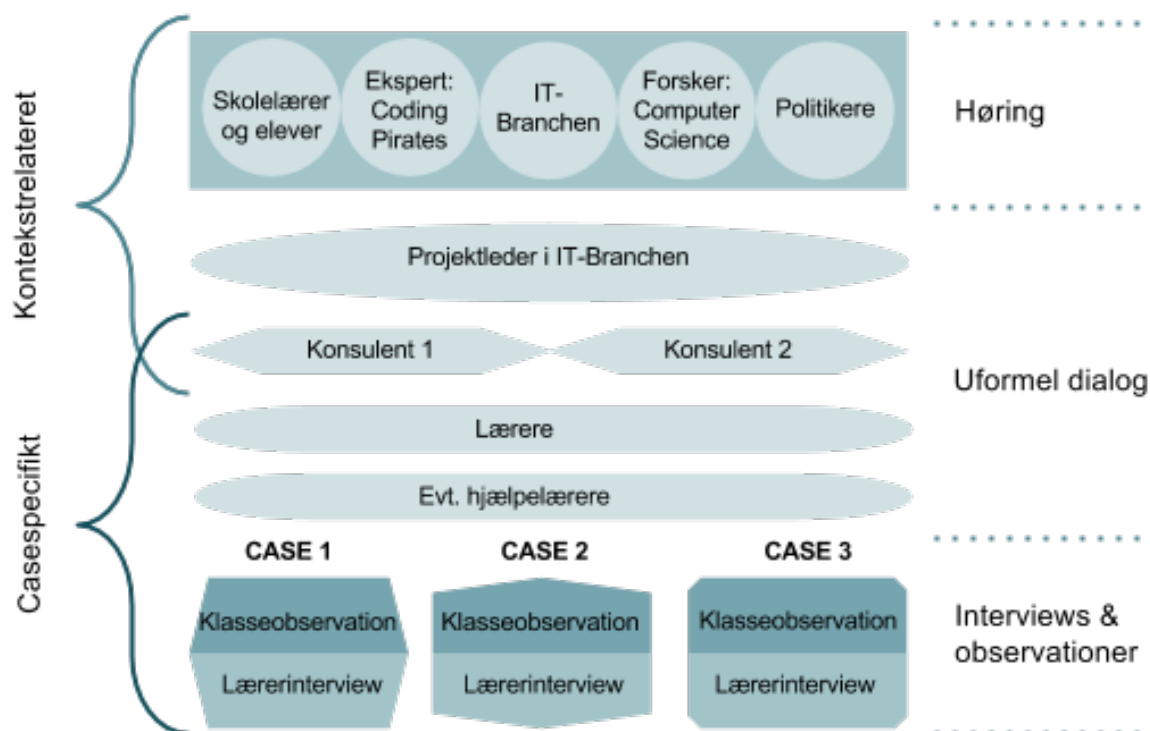
3.5 METODEREFLEKSION

Undersøgelsen besvarer undersøgelsesspørgsmålet gennem et casestudie, som giver indblik i tre konkrete tilfælde, hvor programmering bruges i undervisningen. Ambitionen med casestudiet er herigennem at opnå en større forståelse for hvorledes programmering kan indgå som fagligt element i undervisningen i praksis ud fra konkrete tilfælde, hvorfor undersøgelsen ikke har til formål at bidrage med generaliseringer, som kan give ny grundlæggende indsigt i hvordan programmering skal foregå i grundskolen. For at udlede generaliseringer vil det nødvendigt at inddrage flere skolars erfaringer med programmering i undervisningen eksempelvis ved at inddrage erfaringer fra Københavns Kommunes tilsvarende Coding Class-forløb. Disse erfaringer kan bidrage med større spredning i dataene og derved muligvis give andre eksempler på forløbenes opbygning, lærernes erfaringer samt inddragelse af forskellige fag. En bredere undersøgelse med flere cases kan medføre større generaliserbarhed for undersøgelsens resultater gennem sammenligning af flere praksiseksempler. Flyvbjerg argumenterer dog for, at der ikke findes forudsigende teorier og universelle begreber i studiet af mennesker og samfund, hvorfor konkret og kontekstbunden viden er mere værdifuld, og at casestudiet derfor ikke nødvendigvis skal sigte efter at generalisere på baggrund af undersøgelserne (Flyvbjerg, 2010). Med udgangspunkt i casestudiet som undersøgelsesstrategi har det derfor været nødvendigt at begrænse udstrækningen af dataene for at udføre en dybdegående undersøgelse af de pågældende cases. På trods af at *“pragmatisk validering støtter sig til observationer og fortolkninger og rummer en forpligtelse*

til at handle ud fra fortolkningerne” (Kvale 1997, s. 243), har undersøgelsen ikke til hensigt at handle ud fra resultaterne. Undersøgelsen kan derimod udelukkende drage konklusioner på baggrund af de tre cases i casestudiet, hvorfor konklusionerne blot kan benyttes som en vejledning til at skabe forandringer inden for området fremadrettet. Undersøgelsen bidrager på denne måde med et videnskabeligt indblik i problemfeltet for herved at tilskynde til handlingsforandringer i feltet.

Triangulering

Kombinationen af observationer og interviews benyttes i undersøgelsen til at opnå en forståelse for både den samlede kontekst for hele casestudiet samt hver enkelt case. På denne måde supplerer og validerer metoderne hinanden, da forståelsen ikke kommer fra én kilde, men der er en triangulering af forskellige datakilder: *“triangulation in qualitative research assumes that if two or more sources of data, theoretical frameworks, types of data collected, or researchers converge on the same conclusion, then the conclusion is more credible”* (Tracy, 2010, s. 843). Triangulering bruges til at understøtte undersøgelsens validitet, da de forskellige empiriske dataindsamlinger korrigerer den samlede forståelse der opnås gennem de kvalitative metoder (Stake, 2005). Observationsnoterne, billeddokumentation samt interviewtransskriberingerne bidrager til at bringe læseren med ind i databehandlingen, som er dokumenteret og vedlagt (Bilag 1-16), således kodningen, analysen og tolkningsarbejdet tydeliggøres for læseren. Trianguleringen belyser casestudiet gennem både casespecifik og kontekstrelateret data, hvilket visualiseres i modellen nedenfor. Det casespecifikke feltarbejde udgøres af de uformelle dialoger under skolebesøgene, klasseobservationerne samt lærerinterviewene. Det kontekstrelaterede feltarbejde har bidraget med indblik i den samlede kontekst for alle tre cases og inkluderer en dialog med projektlederen for Coding Class i IT-Branchen, deltagelse i høringen omkring et kreativt it-fag i folkeskolen d. 1. februar 2017 samt de uformelle interviews med konsulenterne fra Coding Class.



Figur 6: Oversigt over inddelingen af casestudiets feltarbejde.

Konsulenterne har kontakt til både den casespecifikke og kontekstrelaterede del af feltarbejdet, da de er ansat af IT-Branchen som eksperter til at undervise i Coding Class-forløbet på skolerne i Vejle Kommune, hvorfor de har berøring med begge arenaer. Triangulering giver således en metodisk kvalitet, der sikrer datavaliditet ved, at flere undersøgelsesmetoder supplerer hinanden (Tracy, 2010, s. 843). For at støtte trianguleringen kunne undersøgelsen med fordel have indarbejdet interviews med elever fra hver skole, hvilket kunne give et direkte indblik i elevernes oplevelse med programmeringen og deres faglige udbytte hermed. Undersøgelsen anvender observationerne i kombination med interviewene med lærerne til at belyse forløbets effekt på læringssituationen og det faglige udbytte. Her prioriteres jævnfør arbejdspørgsmålene lærernes forståelse af elevernes udbytte i relation til forløbets mål.

KAPITEL 4

LÆRINGSTEORETISK PERSPEKTIV

...

I følgende kapitel redegøres for konstruktionismen som undersøgelsens læringsteoretiske perspektiv med afsæt i pragmatismens menneske- og videnssyn. Der redegøres for konstruktionismens læringsteori samt strategi for undervisning, hvilket sammenfattes afslutningsvis i overordnede aspekter for et positivt konstruktionistisk læringsmiljø.

4.1 MOTIVATION FOR LÆRING

Tilegnelse af viden foregår i pragmatismen gennem interaktion og handling i verden (Brinkmann, 2007), hvilket skaber grundlag for et læringsteoretisk perspektiv, der i denne undersøgelse kan anvendes til at opnå forståelse for læringssituationen i Coding Class. Det læringsteoretiske perspektiv bundes således i det pragmatisk udgangspunkt for læring og motivation, der bygger på menneskets naturlige udforskning og handling i verden (Brinkmann, 2007). Motivation for læring beskrives inden for psykologien almindeligvis med en skelnen mellem et lystperspektiv, udgjort af en indre drivkraft, og et meningsperspektiv, som er rationelt betonet, hvor Dewey belyser motivation for læring gennem gøremål, hvor både mål, interesser og tænkning involveres i læreprocessen (Brinkmann, 2007, s. 97). Denne gøremåls læring knytter sig til det pragmatisk udgangspunkt for læring ved at den lærende har mulighed for at udtrykke sig aktivt og undersøgende i læreprocessen. På denne måde opnås motiverende læring i undervisningssammenhænge gennem en bestemt type læringslandskab, hvor den lærende indgår deltagende i en praksis (Brinkmann, 2007, 99), hvilket eleverne skal i Coding Class-forløbene. Elevernes praktisk erfaring med programmering i Coding Class, har samme forudsætning for læring som ved Deweys udlægning af, at erkendelse *“opstår altid i forbindelse med situationer, der involverer en ‘learning by doing’ - man lærer ved at gøre”* (Dewey, 1980, s. 201). Konstruktionismen er en læringsteori, der netop bygger på, at læring sker, når den lærende indgår i konstruktionsprocesser og praktiske erfaringer med verden, herunder teknologi (Ackermann, 2002). Undersøgelsen benytter på denne baggrund konstruktionismen som forståelsesramme for brugen af teknologi i aktiviteterne med programmering under Coding Class i overensstemmelse med det pragmatisk læringsperspektiv. Konstruktionismens læringsteori redegøres for herunder, hvilket både involverer et teoretisk perspektiv på den lærendes erfaringsdannelse og læreproces samt en strategi for undervisning (Kafai & Resnick, 1996, s. 1).

4.2 LÆRING Gennem KONSTRUKTION

Konstruktionismens læringsteori udspringer fra Piagets konstruktivistiske læringssyn, som bygger på at viden ikke overføres fra lærer til elev men konstrueres i en indre kognitiv proces hos den lærende (Ackermann, 2001). Konstruktionismen adskiller sig fra konstruktivismen i dens udbygning af, at læring ikke blot konstrueres af den lærende selv, men at læreprocessen optimeres, når konteksten aktivt engagerer individet i konstruktion af objekter (Papert, 1991, s. 1). Konstruktionismen involverer konstruktion på to niveauer, hvilket både involverer den lærendes konstruktion af viden og en fysisk aktivitet; *“the construction of knowledge in the context of building personally meaningful artifacts”* (Kafai & Resnick, 1996, s. 1). Den lærendes personlige nysgerrighed, engagement og passion for konstruktionen af artefakter er ledende for læreprocessen og konstruktionen af viden, hvorfor der inden for konstruktionismen derfor ikke én rigtig måde at lære på, da læreprocessen

afhænger af individets egne præferencer. Miljøet og de konkrete artefakter har dog indflydelse på den lærendes læreproces, da *”knowledge is formed and transformed within specific contexts, shaped and expressed through different media, and processed in different people's minds”* (Ackermann, 2001, s. 8). Miljøet og artefakter opstiller altså muligheder og begrænsninger for individets læring, fordi de former den individuelle erfaring med verden.

4.2.1 LÆRING GENNEM DESIGN

Individets læreproces styrkes ifølge konstruktionismen igennem designrelaterede aktiviteter: *“activities involving making, building, or programming - in short, designing - provide a rich context for learning”* (Kafai & Resnick, 1996, s. 4). De designrelaterede aktiviteter står i tæt relation til den konkrete og fysiske konstruktion af objekter, men lægger både vægt på individets læreprocessen, hvor det handler om at prøve sig frem og derigennem tilegne sig viden, samt den konkrete udarbejdelse et produkt. Ved at designe, skabe og udtrykke sig kreativt med teknologi, tilegner den lærende sig ny viden og strategier til problemløsning (Papert, 1980). Således bliver fejltagelser og mislykkede forsøg i konstruktionen en del af læreprocessen, hvor der løbende arbejdes på at forbedre konstruktionen i en iterativ designproces (Martinez & Stager, 2013, s. 70). På denne måde er fejltagelser og mislykkede forsøg en del af selve læringen, da den lærende får muligheden for at tilpasse sig til de erfaringer, der gøres undervejs. Konstruktionismen har således en legende og skabende tilgang til læring med en accept af, at fejl vil forekomme. Den konstruktionistiske tankegang bygger således på, at læring skal ske i en legende, interaktiv, eksperimenterende og altid konstuerende læreproces, da *“learners are most likely to become intellectually engaged when they are working on personally meaningful activities and projects”* (Kafai & Resnick, 1996, s. 2). Gennem designrelaterede aktiviteter kan den lærende således forfølge vinkler, elementer og tanker, som de finder interessante.

4.2.2 KOLLABORATIV LÆRING MED IT

Papert forestillede sig allerede i 1970'erne, at børn kunne blive flydende i brugen af computerteknologien og ikke blot interagere med den men ligeledes skabe med den. Han så et potentiale i computerens muligheder for at understøtte børns læring via digital leg og kreativ udfoldelse (Papert, 1980). Papert erfarede ligeledes, at børn bliver motiverede til at lære, når de har mulighed for at være udforskende og aktivt skabende i interaktion med nogen eller noget (Papert, 1980). Læring bør i konstruktionismen således både involvere fysiske objekter samt kollaboration med andre, hvor interaktionen kan understøtte individets læreproces, da: *“Community members act as collaborators, coaches, audience, and co-constructors of knowledge”* (Kafai & Resnick, 1996, s. 6). Dette understøttes ligeledes af den pædagogiske tilgang *Computer Supported Collaborative Learning*, som bygger på foreningen af computerbaseret og kollaborativ læring (Stahl, Koschmann & Suthers,

2006, s. 409). Computeren fungerer her som medium eller fælles ressource, som samler de lærende og muliggør konstruktionen af viden i en social interaktion (Stahl et al., 2006, s. 410). Derved samles de lærende i aktiviteter, som understøtter intellektuel fordybelse og kollaborativ konstruktion af viden. Læring opstår således i samspil mellem individerne i fællesskabet og i den sociale interaktion:

Collaborative learning involves individuals as group members, but also involves phenomena like the negotiation and sharing of meanings—including the construction and maintenance of shared conceptions of tasks—that are accomplished interactively in group processes. Collaborative learning involves individual learning, but is not reducible to it (Stahl et al., 2006, s. 411).

Individet bidrager således til den sociale interaktion og den fælles konstruktion af viden, som ligeledes bygger oven på eller rykker på individets egen konstruktion af viden gennem en forhandling af viden og deling af meninger i fællesskabet. På denne måde kan kollaboration med andre bidrage til, at individet kan forbedre sin viden, men ligeledes at den lærende gennem dialog, feedback og eksempler fra andre, kan forbedre sit eget arbejde, da et *”sharing atmosphere supports innovation, creativity, and the iterative design process”* (Martinez & Stager, 2013, s. 78). Det kan derfor være en fordel at de lærende bruger hinandens viden og konstruktioner, til at forbedre deres eget arbejde og læreproces. Interaktionen med andre kan gavne den iterative proces, så den lærende kan samarbejde med andre eller hjælpes videre af andre. Ved at fortælle andre om problemer eller løsninger og få feedback på samme kan nye muligheder åbnes for den lærendes egen refleksion, samarbejde og fælles problemløsning bidrager til den individuelle progression (Martinez & Stager, 2013, s. 78). Ved brug af læringsteknologier i undervisningen påvirkes den lærendes motivation ligeledes blandt andet af den lærendes egen nysgerrighed, udfordring, selvtillid samt kontrol (Issroff & Jones, 2007, s. 191), hvorfor målet således er, at den lærendes nysgerrighed vækkes gennem personlige præferencer og engagement, hvor den stillede opgave hverken er for let eller svær for den lærende at udføre. Hvis det rette læringsniveau er til stede, og den lærende opnår succes, opnår den lærende større selvtillid. Det er dog ligeledes vigtigt, at den lærende har en oplevelse af kontrol, og således selv har indflydelse på sin læreproces (Issroff & Jones, 2007, s. 191).

4.3 KONSTRUKTIONISTISK UNDERVISNINGSTRATEGI

Indenfor konstruktionismen er der ikke en påbudt metode til at undervise, da læringsmodellerne er centreret omkring den lærende med fokus på aktiv, social og projektbaseret undervisning (Martinez & Stager, 2013, s. 71). Målet med undervisningen er således i konstruktionismen, at elever får plads til at udvikle deres projekter i forskellige retninger, hvorfor udgangspunktet for læring således er, at

den lærende selv skal erfare og opleve for at lære (Martinez & Stager, 2013, s. 71). Instruktioner og informationer kan dog gives af læreren, når læreren vurderer, at giver mening for elevernes læring: *“wise teachers know when to dispense the smallest dose of information possible to ensure forward progress”* (Martinez & Stager, 2013, s. 71). Målet med en konstruktionistisk undervisning er således, at de lærende er i centrum i modsætning til traditionel klasseundervisning og forelæsning med læreren i fokus. Lærers vigtigste ansvar er at have en fornemmelse for elevernes viden, synliggøre den usynlige læring, muliggøre at eleverne kan udføre deres ideer ved hjælp af teknologiske redskaber og materialer, samt guide elevernes læreproces uden nogen form for tvang (Martinez & Stager, 2013, s. 76). I stedet for at instruere eleverne i, hvad de skal gøre, er det mere effektivt, hvis læreren vurderer elevernes forhåndsviden og kort introducerer emnet herudfra med fokus på at engagere eleverne med praktiske opgaver, så de selv kan erfare og konstruere deres viden selv: *“The longer you delay students from getting to the “making” part of the design cycle, the more students will disengage and the longer it will take to learn the lesson”* (Martinez & Stager, 2013, s. 77). For at give eleverne muligheden for at lære på egen hånd, skal læreren lade den lærende gå på opdagelse og eksperimentere med opgaven og understøtte elevens problemløsning, når fejl og misforståelser opstår. Læreren skal ikke kunne løse elevernes problemer, men skal understøtte eleverne i selv at rette deres arbejde og prøve sig frem, indtil problemet løses (Martinez & Stager, 2013, s. 77). Læreren kan indlede feedbackprocesser, hvis eleverne føler sig færdige eller er gået i stå og skal understøtte at elever lader sig inspirere af hinandens arbejde eksempelvis gennem remixing af andres projekter (Martinez & Stager, 2013, s. 78). Hvis eleverne oplever problemer og frustrationer undervejs, er det altså ikke lærers opgave at løse problemerne men at understøtte eleverne i selv at løse deres problemer.

4.4 KONSTRUKTIONISTISK LÆRINGSMILJØ

På baggrund af ovenstående redegørelse foretages en teoretisk sampling (Brinkmann & Tanggaard, 2010, s. 227), som sammenfatter den konstruktionistiske læringsteori til overordnede teoretiske kategorier. Kategorierne repræsenterer aspekter, som påvirker et konstruktionistisk læringsmiljø, der benyttes som analyseapparat til at få indsigt i og vurdere den konkrete læringssituation i Coding Class-forløbene. Den teoretiske samling udgøres af følgende kategorier for et konstruktionistisk læringsmiljø:

Konstruktion og fordybelse: Læring foregår i konkrete afprøvning, udførelse og skabelse med teknologi. Den lærende lærer bedst, når vedkommende involveres i en konstruerende proces af et objekt, hvor den lærende har et følelsesmæssigt engagement eller personlig interesse. Dertil hører, at den lærende skal have det sjovt og finde arbejdet interessant, men dette betyder ikke, at arbejdet skal

være let; når den lærende udfordres og samtidig har det sjovt, opstår et bedre udgangspunkt for læringen.

Fejl og problemløsning: Den lærende skal have plads til at lave fejl undervejs og vigtigere; at lære at sine fejl og rette op på dem. Det handler om at fejle og forbedre sit arbejde i en iterativ proces, hvor en ikke-konkurrerende men delende atmosfære gavner kreativiteten og læreprocessen. Herunder hører også, at inspiration og remixing af andres produkter gavner den lærendes eget arbejde.

Feedback og samarbejde: Feedback kan give den lærende anledning til refleksion, hvilket kan bidrage til den lærendes indre motivation for at forbedre et givent projekt. Både feedback fra læreren og andre lærende kan være gavnligt, når den lærende føler sig færdig med et projekt eller er gået i stå med et problem.

Frihed under ansvar: Den lærende skal selv tage ejerskab over sin læring og ikke være afhængig af at en lærer fortæller, hvad personen skal gøre eller lære, hvilket også gælder i forhold til den afsatte tid til at løse en opgave. Den lærende skal selv kunne administrere sin tid og prioritere sine opgaver og skal ikke begrænses i, hvornår og hvor længe vedkommende skal arbejde med en bestemt opgave. Det handler om at afsætte den nødvendige mængde tid til at opnå de ønskede resultater.

Den guidende underviser: Læreren skal ikke nødvendigvis have alle svarene men i stedet agere guide for elevernes læreproces samt indgå i en fælles problemløsning med eleverne. En lærer kan ikke vide sig sikker på retningen af elevernes projekter og kan derfor ikke forberede sig på at afhjælpe alle problemer, men skal i stedet indgå som en pædagogisk støtte i elevernes problemløsning. Læreren skal lade eleverne gå på opdagelse og få praksiserfaring med opgaven hurtigst muligt ved at undgå unødvendig instruktion.

KAPITEL 5

CODING CLASS I UNDERVISNINGEN

...

Undersøgelsens analyse beskæftiger sig med erfaringerne fra Coding Class gennem to analytiske perspektiver. Første analyseperspektiv har til formål at undersøge læringssituationen under Coding Class og herunder hvilke faktorer, der påvirker denne. I det andet analyseperspektiv undersøges, hvordan lærerne forholder sig til og vurderer Coding Class samt programmerings faglige relevans for undervisningen.

5.1 LÆRINGSSITUATIONEN I CODING CLASS

Følgende analyse behandler læringssituationen i Coding Class ud fra undersøgelsens observationer i de tre cases (Bilag 4, 5 og 6). Forud for analysen er observationsnoterne blevet kodet (Bilag 7, 8 og 9) ud en teoretisk sampling af konstruktionismens læringsteori opstillet i undersøgelsens Kapitel 4 som følgende kategorier: konstruktion og fordybelse, fejl og problemløsning, feedback og samarbejde, frihed under ansvar samt den guidende underviser. Følgende analyse tager udgangspunkt i disse teoretiske kategorier, hvorfra der fokuseres på hvorledes inddragelse af programmering i undervisningen har fungeret i praksis under de tre cases, samt hvilke faktorer der har haft indflydelse på læringssituationen under Coding Class.

5.1.1 KONSTRUKTION OG FORDYBELSE

Case 1

Eleverne udviser stor interesse for deres projekter og arbejder koncentreret det meste af dagen (Bilag 4). Eleverne virker meget opsatte på at udføre deres opgaver og arbejder fokuseret med hver deres opgave, hvilket indikerer, at de er optagede af det, de er ved at konstruere. Elevernes interesse i projekterne er forskellig både på tværs af grupperne og eleverne imellem. Gruppernes interesser viser sig især under præsentationerne af gruppernes spil, hvor det er forskelligt, hvad grupperne lægger vægt på både i selve udformningen af spillene og deres præsentationer af deres spil og proces i virksomhederne (Bilag 4). Gruppe B udviser eksempelvis stor interesse for koden i deres præsentation af spillets funktioner, da de viser koden frem, selvom de ikke er blevet bedt om det (Bilag 4, s. 30). De fortæller i munden på hinanden omkring spillets kvaliteter og udfordringer i koden. De oplevede nogle problemer i forhold til koden, men dette synes ikke at have sænket deres motivation men muligvis forstærket den. Gruppe C har derimod været optaget af det grafiske arbejde og selve designet af spillet (Bilag 4, s. 30). For Gruppe C bliver kodningen et middel til at lave et flot spil og muliggør således realisering af gruppens mere grafiske interesse. Elevernes fortællinger omkring deres spil og proces giver udtryk for, at deres arbejde og uddelegeringen af opgaverne i høj grad har været interessebåret, hvilket også ses internt i grupperne, hvor de inddeler deres opgaver efter, hvad de forskellige medlemmer synes er interessant eller har lettest ved at udføre. I Gruppe H har de eksempelvis samarbejdet om koden, men nogle af gruppemedlemmerne har fokuseret mere på deres merchandise, da de har haft større interesse for dette element end de andre (Bilag 4, s. 36). Grupperne giver altså plads til, at medlemmerne kan arbejde med det, de synes er sjovest at arbejde med, hvilket bidrager til en fri læreproces. Elevernes interesse for deres projekter viser sig også i, at nogle elever har arbejdet videre derhjemme med udfordringer, de har haft undervejs:

Drengene er hurtige til at forklare, hvilke tekniske dele, de har fået forbedret siden sidst. Konsulent CP spørger dybere ind til nogle af de problemer de havde sidst og sætter sig ned ved dem, imens drengene fortæller ham forskellige overvejelser de har gjort sig og hvilke tekniske løsninger, de har forsøgt sig med (Bilag 4, 23).

Dette vidner om en stærk interesse for projektet blandt eleverne men ligeledes om elevernes motivation for at løse problematikker, de oplever i deres proces. Der ses på denne måde en sammenhæng mellem elevernes interesse og udfordringerne i deres praktiske arbejde, da de gerne vil bruge deres fritid på at arbejde videre med det.

Case 2

Arbejdsfordelingen i grupperne i Case 2 virker ligeledes lystbetonede, da de laver opgaver, de enten er motiverede til eller ud fra, hvad de hver især er gode til (Bilag 5, s. 45). Nogle elever bruger eksempelvis meget tid på at beskrive deres roller i virksomhederne til den app, de er ved at udvikle, og er opmærksomme på, at de har bidraget med noget forskelligt til virksomhederne (Bilag 5, s. 46). Nogle elever fokuserer på beskrivelsen, mens andre er optagede af tegningen af gruppemedlemmerne til appen, hvor andre især er interesserede i selve kodningen. Eleverne virker således personligt engagerede i deres virksomheder i grupperne og er bevidste om, at de har forskellige roller og besidder forskellige kompetencer. Nogle elever er så optagede af deres opgaver, at de bliver inde i frikvarteret for at arbejde videre (Bilag 5), hvilket kan vidne om, at de enten er yderst motiverede eller udfordrede med at udføre deres opgaver. Sidst på dagen spørger Lærer HG eleverne om tilbagemeldinger på dagen, hvor eleverne giver udtryk for, at det både havde været svært og sjovt:

P1 siger, at det har været sjovt, men det har været lidt trættende med så meget kodning. D7 bryder ind og siger, at det er det, der har været det sjoveste. (...) D4 siger, at han synes det er godt, fordi det ikke er en normal skoledag. Det har både været sjovt og teknisk. (...) P3 synes det har været sjovt, fordi hun kun havde programmeringen (Bilag 5, s. 50).

Der ses således en tendens til, at eleverne forbinder deres projekter med både at være udfordrende og spændende, selvom det er forskelligt, hvad de har udvist interesse for.

Case 3

Eleverne fra sjette årgang virker meget forberedte og spændte på dagens aktiviteter, som fjerde og femte klasse tager imod med stor begejstring (Bilag 6). De yngre elever er optagede af opgaverne og

fravælger flere gange at holde pause for at kunne arbejde videre med opgaverne. Dette ses i flere grupper, hvor P17 fra Gruppe D eksempelvis udtrykker skuffelse over, at de har pause, da det ringer til det første frikvarter, hvor hun skuffet udbryder: *“Ej, har vi pause? Allerede?”* (Bilag 6, s. 58). Eleverne udviser et engagement for programmeringsopgaverne i deres modvillighed for at holde pause og ønske om at arbejde videre med aktiviteterne. Da lærerne kommer tilbage fra lærerværelset efter første frikvarter, er eleverne eksempelvis allerede startet i grupperne og sidder koncentrerede med deres computere (Bilag 6, s. 58). Dette tyder på, at eleverne både er optagede af opgaverne og synes, at det er så interessant, at de prioriterer arbejdet med aktiviteterne højere end at holde. En af hjælpelærerne, som har femte klasse til dansk og ikke har været involveret i Coding Class-forløbet, udtrykker stor begejstring for elevernes koncentration og roen i grupperne på trods af, at så mange elever er samlet om en fælles aktivitet (Bilag 6, s. 58). Elevernes engagement ses især, når de udfordres i det, de laver. D4 fra Gruppe B virker eksempelvis motiveret i sin problemløsningsproces:

D4 siger frustreret ud i rummet “Jeg får ikke nogen score, hvorfor får jeg ikke nogen score?”. Han arbejder intenst videre, hvilket medfører at en høj bippen begynder at afspilles højt, hvilket får ham til at grine højt. “Nu virker scoren vist” udbryder han i glæde ud i rummet og arbejder så videre med at tilpasse scoren. (Bilag 5, s. 56)

D4 udviser en ivrighed for at løse sit problem og arbejder meget fokuseret på at løse det. Det virker derfor som om, at udfordringen er med til at motivere ham, men det er dog tydeligt, at han i sin problemløsning sætter pris på en progression i problemet, hvilket giver ham en iver for at fortsætte.

På tværs

Eleverne i alle tre cases er optagede af deres arbejde med programmeringsopgaverne, hvor de opsøger og udfører opgaver, de synes er spændende. Elevernes interesser i Case 1 og 2 viser sig både i deres individuelle og gruppearbejde, hvor deres opgaver inddeles ud fra, hvad de forskellige medlemmer synes er mest interessant eller har lettest ved at udføre. I Case 3 ses interessen især i elevernes iver for, at løse de problemer, de støder ind i. Det er på tværs af de tre cases forskelligt, hvad der inspirerer og motiverer eleverne, men der synes altså at være en sammenhæng mellem udfordrende elementer i deres arbejde og deres engagement. Det er derfor også forskellige aktiviteter, eleverne sidder med, hvilket kan have en konsekvens for elevernes udbytte, da de ikke får samme erfaring med programmeringen. Den interessedrevne inddeling af opgaver bevirker dog så høj grad af engagement, at eleverne vælger at bruge deres fritid og frikvartererne på at arbejde videre med det. På tværs af alle cases var der stor koncentration og engagement i starten af dagen, hvilket dog især faldede efter frokostpausen, hvilket vidner om, at eleverne bliver trætte i hovedet i løbet af dagens aktiviteter.

5.1.2 FEJL OG PROBLEMLØSNING

Case 1

Dialogen blandt eleverne i Case 1 bærer præg af en fælles opfattelse af, at det er tilladt at fejle og bede om hjælp. Hvis der opstår problemer, indgår eleverne ofte i en fælles problemløsning, hvilket eksempelvis ses i Gruppe B, hvor de inddrager konsulenten for at forbedre deres program (Bilag 4, s. 23). Eleverne og konsulenten indgår her i en dialog med plads til opklarende spørgsmål og problemløsning i fællesskab. Det kostede penge for elevernes virksomheder at få hjælp af konsulenterne, hvorfor eleverne virker meget opmærksomme på, hvornår det er nødvendigt at få hjælp, og hvornår de kan nøjes med at hjælpe hinanden (Bilag 4). Problemløsningen foregår derfor ofte i grupper på trods af, at eleverne arbejder med hver deres opgaver. Eleverne er villige til at stoppe deres eget arbejde for at hjælpe hinanden, hvis andre elever oplever problemer. Dette ses eksempelvis i Gruppe E:

Gruppe E arbejder rimelig tavst. P6 sidder på modsatte side end de tre drenge i gruppen. P6 og D10 taler på tværs af bordet, hvor P6 prøver at forklare D10, hvordan han kan løse sit problem. Efter nogle minutter med dette, rejser P6 sig op og går over på den anden side for at hjælpe D10, som er gået i stå i sit grafiske arbejde. Efter hun har hjulpet ham, sætter hun sig ned igen, D10 går videre med sit arbejde og gruppen bliver tavse i deres arbejde igen (Bilag 4, s. 25).

Eleverne er altså gode til at støtte hinanden i de udfordringer og problemer, de oplever undervejs, og de synes altså at have en fælles ansvarsfølelse over for projekterne.

Case 2

Eleverne sidder med et helt nyt program og laver tutorials i første undervisningsdel, hvor de i høj grad spørger læreren og konsulenterne om hjælp uden selv at forsøge at løse problemerne gennem tutorials, på egen hånd eller ved at spørge sidemanden først (Bilag 5, s. 43). Efterhånden som de begynder at arbejde med deres egne projekter i programmet, bliver eleverne dog mere selvstændige og begynder at løse problemerne internt i grupperne i stedet for at spørge læreren eller konsulenterne (Bilag 5). Dette sker dog samtidig med, at hjælpen fra konsulenterne og læreren begynder at koste penge for virksomheden, hvilket derfor kan være understøttende for at eleverne ønsker selv løser deres problemer. Eleverne virker dog til både at besidde evnerne til og stor interesse i at hjælpe hinanden, hvis der opstår problemer. D1 støtter eksempelvis P1 i hendes problemløsning:

P1 læner sig på tværs ind over bordet til D1, for at se hvordan han har sat sin kode op. Han læner sig derefter ind over hendes computer, for at se hvad hun mangler. Han peger på skærmen og forklarer hende stille og roligt hvad hun skal gøre. Derefter fortsætter de hver for sig (Bilag 5, s. 43).

D1 går her fra hans eget arbejde for at hjælpe P1, fordi han fornemmer, at hun har brug for hjælp. P1 er samtidig opmærksom på, hvad D1 er i gang med, hvilket gør, at hun hurtigt kan få hjælp af nogle, som er længere fremme end hende selv. Der er en generel konsensus i klassen om, at det er tilladt at afbryde hinandens arbejde for at få hjælp og det er således også tydeligt i klassens miljø, at det er okay at sidde fast eller have problemer. De indgår ofte i en proces af fælles problemløsning uden at dømme, men der ses også få eksempler på, at eleverne kan blive frustrerede, når de hjælper hinanden:

P1 sidder med koden i Gruppe A og spørger læreren efter hjælp, men han siger, at hun skal google sig frem til svaret. P2 siger hurtigt, at hun ved hvordan man gør, hvorefter de to begynder at sidde om samme computer for at løse problemet. P2 peger på computeren og forklarer P1, hvad hun skal gøre, men bliver en smule frustreret fordi P1 ikke forstår, hvad hun skal gøre. P2 forklarer derefter lidt grovere hvad der skal gøres. De bliver ved til de løser problemet (Bilag 5, s. 47).

Det er her usikkert, hvorvidt frustrationen opstår på grund af elevens manglende evner til at hjælpe den anden på det niveau, hun befinder sig, eller om det netop er den anden elevs manglende forståelse, som gør, at den hjælpende elev bliver frustreret. Tonen mellem de to elever bliver en smule grovere grundet problemerne, men det hindrer dem ikke i at fortsætte deres problemløsning. Eleverne tager altså ansvar for de fælles projekter og hinandens opgaver, omend frustration kan opstå, hvis eleverne ikke forstår hinanden.

Case 3

Nogle af fjerde og femte klasses eleverne følges ad i løbet af dagen, hvor andre blot hjælper hinanden, når der opstår problemer (Bilag 6). På trods af, at eleverne ikke har nogen erfaring med programmering, er det naturligt for dem at søge hjælp ved hinanden, hvilket eksempelvis ses i Gruppe H, hvor medlemmerne hjælper hinanden på tværs:

P27 forklarer P28 hvad hun skal gøre i sit spil. Hun læner sig ind over bordet, for at følge med i, om hun gør det rigtigt. De sidder og arbejder videre sammen. P25 udtrykker frustration over, at hun ikke forstår noget i hendes kode. P27 og P28 vender sig og går i

dialog med hende og vender P27s skærm over mod hende, så de kan pege i koden, for at forklare hende det (Bilag 6, s. 61).

Der ses her en accept af, at eleverne kigger hinanden over skulderen for at følge med, samt at de har brug for hjælp for at komme videre. Dette ses eksempelvis også i Gruppe G:

D27 læner sig ind for at hjælpe D28 med hans kode. D27 forklarer stille og roligt, hvad D28 skal gøre. Efter lidt tager D27 fat i hans egen computer og finder hans kode frem, fordi han er færdig med tutorialen og kan vise hvordan han selv har gjort det. D27 og D28 kigger sammen på tværs af de to koder, og med inspiration fra D27s kode finder de ud af hvordan D28 skal tilpasse sin. D27 forklarer "Du skal skrive 10 her" og peger på skærmen, "det er fordi jeg har flere med, at der står 20 i min kode" (Bilag 6, s. 60).

I dette eksempel går D27 ind og støtter D28 i hans problemløsning ved at tage udgangspunkt i sin egen kode og forklarer, hvordan koden er sat op, og hvordan han skal tilpasse den. Den fælles problemløsning, ses i flere grupper, hvor eleverne samarbejder for at løse deres problemer. I Gruppe M forklarer P42 eksempelvis D51, hvordan han skal tilpasse deres fælles kode, så den udfører den ønskede funktion (Bilag 6, s. 65). Hun tager ikke over, men forklarer med ord, hvordan D51 skal gøre, selvom han har udfordringer ved at forstå problemet i koden. Eleverne er således gode til at støtte hinanden i deres problemløsning, når der er en, der har en bedre forståelse end en anden.

På tværs

Eleverne virker på tværs af de tre cases til at være optaget af løsningen af de problemer, de støder ind i undervejs, hvor disse ikke har en negativ effekt på deres engagement. Eleverne fokuserer ikke på de problemer, de støder på, men på deres løsning heraf, hvor der ikke ses nogen skam i at tage fejl eller bede andre om hjælp. Eleverne går naturligt til problemløsningen i fællesskab, men der ses nogle få eksempler på, at eleverne kan blive utålmodige med hinanden. I Case 1 og 2 er eleverne opmærksomme på at hjælp fra konsulenter og lærere koster penge for virksomhederne, hvorfor de prøver at løse problemerne selv først.

5.1.3 FEEDBACK OG SAMARBEJDE

Case 1

I løbet af dagen ses en tendens til, at eleverne arbejder meget hver for sig, men at de begynder at samarbejde, så snart der opstår problemer (Bilag 4). Eleverne er gode til at hjælpe hinanden, men de

kan have svært ved ikke at overtage den anden elevs arbejde og løse problemerne for dem, hvilket eksempelvis sker i Gruppe C:

I Gruppe C sidder P3 og arbejder på en intro til deres spil. Hun spørger P2 om hjælp til noget kode i Scratch, hvor P2 først prøver at forklare det fra siden, mens hun læner sig ind fra siden, for at følge med i, hvad P3 foretager sig. Der går nogle minutter, hvor P2 vejleder P3, indtil P2 drejer computeren over mod sig og begynder at rette i det selv. De sidder lidt og arbejder sammen om problemet indtil det er løst, hvorefter P3 får computeren tilbage igen og kan fortsætte selv (Bilag 4, s. 25).

P2 har en forståelse for, at P3 har brug for hjælp, men der tages ikke hensyn til, at P3 selv skal have mulighed for at løse sit problem. På denne måde blokeres P3's egen problemløsning og potentiel læring, men samtidig kan de begge hurtigere fortsætte med deres egne opgaver. Ved at inddrage hinanden undervejs får eleverne således mulighed for hurtigt at kunne fortsætte deres arbejde, end hvis de skulle løse deres problemer selv. Dette kan dog påvirke elevernes egen læring, da de ikke selv får mulighed for at finde potentielle løsninger på deres problem.

I klasseværelset er der et miljø, hvor eleverne kan få feedback og vise deres arbejde frem for andre elever, lærere eller konsulenter. D9 fra Gruppe E viser eksempelvis noget grafisk arbejde frem for en af konsulenterne, som med det samme påpeger en forbedringsmulighed med produktet (Bilag 4, s. 26). Eleven tager ikke feedbacken personligt men kan forstå konsulentens pointe, hvorfor han går direkte i gang med forslaget. Midt på dagen skal eleverne præsentere deres spil og virksomhed for konsulenterne, som giver dem feedback og forslag til, hvad de skal fokusere på for at nå at færdiggøre deres projekt, inden dagen er omme. Eleverne får på denne måde respons på deres arbejde, hvor der lægges vægt på den løbende forbedring af deres projekter i en iterativ proces. Feedbacken sikrer ligeledes, at eleverne får udarbejdet det bedste produkt, på den tid, som de har til rådighed og udfordrer dem i deres læreproces.

Case 2

Opgaverne og rollerne i virksomhederne synes at være inddelt efter interesse, hvor nogle elever arbejder mere sammen om opgaverne internt i grupperne end andre. Gruppe C arbejder eksempelvis meget effektivt og målrettet, hvor gruppen sidder stille og arbejder med hver deres opgaver og spørger kun indimellem hinanden til råds (Bilag 5). Modsat arbejder Gruppe E eksempelvis med meget udtalte roller og er løbende i dialog med hinanden om deres fælles projekt, hvor de også strukturerer udførelsen af opgaver sammen (Bilag 5, s. 47). Der er således en meget demokratisk og ærlig dialog i gruppen, hvor de sammen får struktureret deres arbejde og deres produkt i en forhandling blandt medlemmerne i gruppen. Denne forhandling ses også i feedbackprocesserne, hvor eleverne løbende

henvender sig både til gruppemedlemmer, andre elever i klassen, konsulenterne eller læreren for at få tilbagemeldinger på deres projekter. D2 fornemmer eksempelvis, hvad D4 laver, og henvender sig for at få hjælp med et problem (Bilag 5, s. 43). D4 forklarer, hvad han skal gøre, hvorefter D2 begynder at arbejde på det i fællesskab med sin sidemand. Eleverne lader sig altså inspirere af hinanden og bruger hinanden, når det giver mening for dem. Eleverne henvender sig især til læreren, hvis de er færdige med deres opgaver, og læreren kommer med inspiration til, hvordan de kan forbedre deres projekter. Han guider eksempelvis D2, som er færdig før tid, videre med hans app ved at komme med konkrete forbedringsforslag:

D2 spørger højt "Hvad gør man, når man er færdig?", Læreren spørger "med det hele?", hvortil D2 nikker. Læreren går hen for at se, hvad han har fået lavet. Læreren foreslår nogle konkrete dele, som D2 kan arbejde videre med og derved forbedre hans app [...]. D2 sætter sig derefter koncentreret og arbejder videre med sin app (Bilag 5, s. 43).

Lærer HG har således en forståelse for, hvad der teknisk er muligt og bruger dette til at være en guidende underviser, der gennem sin feedback kan inspirere eleverne til deres videre arbejde. På denne måde kan eleverne løbende tilpasse deres arbejde og bliver udfordret gennem feedbacken fra andre.

Case 3

Eleverne fra fjerde og femte klasse samarbejder konstruktivt og giver hinanden feedback undervejs på trods af, at de ikke har erfaring med at programmere. Dette ses eksempelvis i samspillet mellem D1 og P10, som arbejder sammen på en computer og P9, der støtter sig op af de to:

D1 og P10 arbejder stadig på samme computer og hjælpes ad med at opbygge deres spil. P9 sidder ved siden af, men læner sig ofte ind og hjælper dem også eller beder om hjælp af dem. De prøver i fællesskab D1 og P10s spil af, men der opstår fejl, så de går ind i koden igen og begynder at løse problemet (Bilag 6, s. 56).

De arbejder i fællesskab om at forbedre deres spil og indgår automatisk i dialog med hinanden og giver hinanden feedback og hjælper dermed hinanden til at teste, give feedback og i dette tilfælde hjælpe med problemstillinger, der opstår undervejs. Elevernes arbejde er hovedsageligt individuelt i løbet af dagen, men de fleste støtter sig enten til hinanden eller følges ad i deres arbejde. Det er således også i orden at lade sig inspirere af hinandens arbejde. Eleverne indgår i en iterativ proces,

hvor de løbende tester spillet for at forbedre det, opdage fejl eller for at tilpasse det i forhold til de andres løsninger.

På tværs

Eleverne arbejdede sammen og indgik i en fælles læreproces på tværs af de tre cases. Selv ved individuelle opgaver havde eleverne en tendens til at samarbejde om deres opgaver, især hvis der opstod problemer undervejs. Selv i Case 3, hvor eleverne ikke havde nogen erfaring med programmering, var de opsøgende for at hjælpe hinanden. Elevernes samarbejde bevirkede både at de blev hjulpet videre, når de sad fast, men nogle elever havde tendens til at overtage, uden at lade eleverne løse problemerne selv. Eleverne gik på tværs af de tre cases meget op i at forbedre deres produkter, hvorfor de brugte hinanden, lærerne og konsulenterne til at give feedback på deres arbejde. Disse feedbackprocesser medvirker både til, at eleverne kan forbedre deres arbejde og holde sig beskæftigede i deres læreproces, men medvirker ligeledes til at udfordre eleverne til at opsøge andre muligheder med deres arbejde.

5.1.4 FRIHED UNDER ANSVAR

Case 1

Eleverne virker til at vide, præcis hvad de skal foretage sig i løbet af dagen, og hvilke opgaver de er ansvarlige for at løse i gruppen. De virker meget målbevidste og flere gange i løbet af dagen er der fuldstændig ro i klassen, fordi de sidder fordybet i deres opgaver (Bilag 4). Nogle elever begynder dog sidst på dagen at gå rundt mellem hinanden og virker ukoncentrerede og en smule rastløse (Bilag 4). Dette kan både handle om en manglende interesse i det, som de er i gang med, eller at de ikke tager ansvar over deres arbejdsopgaver. Nogle af grupperne fik ikke afholdt deres virksomhedsmøder først på dagen, selvom det var et krav fra læreren (Bilag 4). Lærerne må derfor bede nogle grupper aktivt om at stoppe det, de er i gang med, for at sikre sig at de når at afholde et virksomhedsmøde. Eleverne i Gruppe E arbejder eksempelvis koncentreret med deres projekt, og D9 er allerede startet med at redigere deres spil, hvor hans koncentration bliver afbrudt på grund af læreren:

Lærer JB beder dem om at gå i gang med deres virksomhedsmøde. D9 kigger ikke engang op fra sin computer, men fortsætter med at arbejde på spillet. De andre gruppemedlemmer småsnakker lidt over bordet, men dykker også ned i deres opgaver igen. Først da Lærer JB kommer forbi igen og spørger D9 om, hvad han laver og om de er gået i gang med virksomhedsmødet, retter han blikket mod læreren og fortæller at han bare lige mangler at rette noget. Lærer JB fortæller D9 med fast stemme, at de skal gå i

gang med virksomhedsmødet nu, hvilket de så gør (Bilag 4, s. 24).

Elevernes engagement og praktiske udførelse af deres opgaver blokeres på denne måde af læreren, som prøver at guide deres proces og hjælpe dem med at opnå nogle konkrete mål. Eleverne får således ikke mulighed for selv at prioritere deres arbejde samt at tage den nødvendige tid til at udføre opgaven, fordi læreren prøver at sikre, at gruppen bliver færdige, før dagen er omme. Kravet om afholdelse af virksomhedsmøder kan således i nogle tilfælde bremse elevernes koncentrerede arbejde. Det viser dog også, at eleverne ikke tager højde for den opgave, læreren har bedt dem om at udføre, hvorfor de mister en del af deres frihed og ansvar.

Case 2

Alle eleverne er selvkørende men er i løbende dialog med hinanden internt i grupperne og med læreren om, hvad de mangler (Bilag 5, s. 48). Grupperne inddeler selv opgaverne mellem dem og sørger for progression i deres arbejde, med lidt assistance for læreren, som løbende hjælper med de overordnede rammer og giver eleverne løbende tidsfrister op til pauserne. På trods af lærerens løbende tidsfrister er flere elever stressede op til deres fremlæggelser sidst på dagen:

Gruppe D bliver desperate over, at de skal præsentere om lidt. De vil rigtig gerne have fem minutter mere til at færdiggøre deres app. Konsulent CP sætter dig derfor ned til dem, for at få styr på det sidste, og læreren siger, at de kan arbejde stille indtil det bliver deres tur til at præsentere, og at de burde have gjort det færdigt til nu (Bilag 5, s. 50).

Gruppe D er således meget opsatte på, at de ønsker at gøre deres projekt færdigt, før de skal fremvise det. Eleverne kan være endt i en presset situation, fordi de på én dag skal nå at lave en færdig app, selvom de aldrig har prøvet programmet før, eller fordi gruppen ikke har tilpasset deres ambitioner ud fra den tid, de har haft til rådighed. Det viser dog også en ambitiøs gruppe, der går op i det produkt, som de er ved at udarbejde.

Case 3

De fleste sjette klasser har forberedt sig godt og tager et stort ansvar for aktiviteterne, som de yngre elever skal igennem. Eleverne fra sjette årgang i Gruppe A, deltog ikke dagen før og har derfor ikke forberedt noget til deres gruppe af yngre elever (Bilag 6, s. 55). De løser dog problemet i situationen ved at gennemgå de samme tutorials med eleverne, som de selv tidligere har arbejdet med, hvorfor de således løser problemet på trods af, at de ikke er forberedte. Modsat ses ved Gruppe L en høj grad af forberedelse, hvor de blandt andet har sat et lamineret skilt op på døren, der byder de yngre elever

velkomne (Bilag 6. 65). Dette viser både, at eleverne har glædet sig til dagen, og at de har forberedt sig til, at de yngre elever kom. I de fleste klasser virker de ældre elever til at tage deres opgave som lærere seriøst. I Gruppe C er klasseværelset eksempelvis helt stille, og de yngre elever sidder koncentreret og arbejder, imens eleverne fra sjette cirkulerer rundt og hjælper de yngre elever (Bilag 6, s. 57). Lærerne udtrykte bekymring omkring denne gruppe på lærerværelset, da eleverne fra sjette årgang ikke var fagligt stærke, men de viste sig at være både pædagogiske og engagerede i de yngre elevers arbejde. At de var svage elever fornemmes ikke i klasseværelset, da de både virker til at forstå programmet og evne at kommunikere det videre til de yngre elever (Bilag 6, s. 57). Der ses dog ligeledes få eksempler på, at elev-underviserne ikke formår at hjælpe eleverne og være støttende lærere. I Gruppe K sidder sjette årgang eksempelvis i et hjørne for sig selv og snakker eller sidder med deres mobiltelefoner i stedet for at hjælpe de yngre elever (Bilag 6, s. 64). De spørger løbende, om nogen mangler hjælp, men de er mere optagede af hinanden, hvorfor de ikke lever op til det ansvar, de er blevet pålagt. De yngre elever sidder dog fordybet i deres programmeringsarbejde på trods af elev-underviserens manglende engagement, hvilket viser, at eleverne fra sjette klasse ikke nødvendigvis er lige så engagerede i at undervise de yngre elever i programmering, som de yngre elever er i at blive undervist og stifte bekendtskab med programmering.

På tværs

Eleverne er på tværs af de tre cases meget selvstændige i udførelse af deres opgaver og planlægningen heraf. Lærerne indgår i alle cases som planlægger i forhold til de overordnede rammer og giver eleverne nogle konkrete krav for at sikre progressionen i deres arbejde. Eleverne i Case 1 og 2 er lidt tidspresset i løbet af dagen, hvor de ikke får helt frihed til at løse deres opgaver på den måde og med den nødvendige tid, de har brug for. Forløbet har et stramt program og der ses eksempler på, at lærerne må tage styringen for at sikre, at eleverne bliver færdige, hvilket blokerer elevernes naturlige progression i deres arbejde. Her fratages eleverne noget af deres frihed, da lærerne blander sig i gruppernes aktiviteter, for at sikre elevernes læring og progression i aktiviteterne.

5.1.5 DEN GUIDENDE UNDERVISER

Case 1

Lærerne indgår mest i en dialog med eleverne omkring deres proces frem for om deres kode. En lærer har dog vejledt Gruppe A om deres kode, som gruppen fortæller om under præsentationerne; de havde en idé til at forbedre spillet med nogle specifikke funktioner, men blev frarådet dette, da læreren mente det var for svært (Bilag 4, s. 28). Konsulenterne giver dog udtryk for, at gruppens idé godt kan lade sig gøre og foreslår dem at arbejde videre med det efter præsentationerne, hvor de vil hjælpe dem med det. Her kan læreren have bremset elevernes kreative udfoldelse og potentiel læring ved at

fraråde eleverne at lave funktionerne. Konsulenternes tilstedeværelse understøtter her elevernes kreative udfoldelse, da de ikke ser begrænsninger i forhold til sværhedsgrad, hvor læreren har en mere guidende rolle som sikrer, at de arbejder med realistiske mål i deres virksomheder.

Under præsentationerne havde eleverne nogle generelle problemer i kodningen, som konsulenterne derfor gennemgår på klassen i tredje undervisningsdel (Bilag 4). Instruktionen holdes kort og generel, så eleverne således selv har mulighed for at tilpasse den generelle information til deres egen kode og selv må opnå erfaring med at løse problemer, de støder på undervejs. For at sikre at eleverne bliver færdige til tiden, minder lærerne løbende eleverne om, hvad de skal huske at tage stilling til og hjælper dem med at konkretisere deres opgaver for at blive færdige. Lærerne sikrer dermed, at eleverne får udnyttet den sidste dag med Coding Class bedst muligt og får færdiggjort deres projekter. Det har en guidende effekt, hvor læreren kan støtte elevernes proces, før de bliver tidspressede og muligvis ikke kan nå at færdiggøre deres projekter. Nogle af eleverne synes at være pressede, imens de arbejder på at blive klar til at præsentere deres produkter. Dette ses eksempelvis i Gruppe C, hvor læreren guider D7 videre:

Lærer SB spørger ham, om han er ved at blive presset her op mod deadline. D7 bekræfter at han har lidt travlt og at det er irriterende at det ikke virker. Lærer SB foreslår at han i stedet begynder at lave præsentationen, som de alligevel ikke er færdige med, for at bruge tiden fornuftigt. Dette går D7 i gang med med det samme (Bilag 4, s. 26).

Læreren fungerer her som en guide for elevens proces uden at løse problemet for ham, hvilket bevirker, at eleven ikke går i stå, men kan arbejde videre med projektet og nå at blive færdig.

Case 2

Dagen er planlagt således, at eleverne først får en kort introduktion af det nye program på klassen, hvorefter de skal arbejde individuelt med tutorials i programmet (Bilag 5). Eleverne færdiggør opgaverne med forskellig hastighed, men både konsulenterne og læreren er meget opmærksomme på, at give konkrete forslag til forbedringer eller bede dem om at hjælpe hinanden, når en elev er færdig før tid. På denne måde udfyldes tiden og eleverne engageres med opgaver, selvom de er færdige end før de andre elever. Eleverne spørger meget læreren om hjælp, men læreren er dog opmærksom på at fortælle eleverne, at de skal finde hjælp i tutorials eller hjælpe hinanden. Både læreren og konsulenterne tydeliggør for eleverne, at de ikke kan hjælpe med alle problemer og er meget åbne omkring, at de ikke har alle svarene. Konsulent CP sidder eksempelvis med P3, hvor de prøver sig frem sammen:

P3 sidder med Konsulent CP og arbejder på knapperne i appen. Konsulent CP kender ikke svaret på hendes spørgsmål, men han foreslår nogle ting de kan prøve. De sidder i nogle minutter og prøver sig frem sammen, hvor P3 selv prøver nogle ting af i programmet og konsulenten kommer med nogle forslag indtil de har løst problemet (Bilag 5, s. 46).

Her indgår konsulenten i en fælles læreproces med eleven og engagerer vedkommende i en problemløsningsproces, hvor det er tilladt at lave fejl og ikke kende udfaldet på forhånd. Eleven virker ikke forstyrret heraf, tværtimod synes hun at være engageret i at løse problemet og virker ikke bange for eller flov over at fejle. Konsulenten viser på denne måde, at han ikke ved eller kan hjælpe eleverne med alt men sætter fokus på problemløsning. Til tider giver Lærer HG udtryk for, at han ikke kan løse elevernes problem, men han er opmærksom på elevernes fremgang og deres kompetencer, hvilket medvirker til, at han hurtigt kan udpege nogle andre, som kan hjælpe med de aktuelle problemer (Bilag 5, s. 43).

Case 3

Lærernes involvering i undervisningen af de yngre elever er begrænset; de går rundt blandt grupperne det meste af dagen og hjælper til, hvis der opstår sociale problemer eller for at støtte elev-underviserne med deres planlægning og udførelse af dagens aktiviteter (Bilag 6). Lærerne overlader på denne måde overlader en stor del af ansvaret for dagens forløb til elev-underviserne i de forskellige grupper, som har haft tid til at planlægge forløbet med støtte fra lærerne den forgangne dag. Konsulenten går rundt blandt klasserne og hjælper med udfordringerne i koden, men dette gør ingen af de andre lærere. Lærernes rolle er derfor især pædagogisk, hvor de er opmærksomme på manglende koncentration i grupperne eller sociale problemer. Efter frokostpausen sidder lærerne i fællesrummet og drikker kaffe og venter på, at eleverne henvender sig selv med problemer eller spørgsmål.

Elev-underviserne afholder fælles introduktioner rundt i klasserne og får hurtigt de yngre elever til at arbejde med tutorials selv. Flere grupper bruger storskærmen til den generelle introduktion og hjælper herefter løbende eleverne med deres individuelle problemstillinger. Den generelle introduktion passer ikke til alle de yngre elevers niveau, og elever i Gruppe E bliver eksempelvis utålmodige, fordi præsentationen er for langtrukket til dem (Bilag 6, s. 58). Der ses dog også eksempler på, at elev-underviserne tilpasser undervisningen til de enkelte elevers behov. I Gruppe G har D6.13 eksempelvis udformet et lille spil, for at sætte en yngre elev ind i mekanismerne i koden, som han har svært ved at forstå (Bilag 6, s. 60). Den yngre elev skal her lægge papstykker med instruktioner ned i en rækkefølge, som D6.13 herefter udfører. D6.13 målretter her undervisningen til den enkelte elevs behov gennem en alternativ og kreativ tilgang, som kan medvirke til, at eleven får en for forståelse og dermed et bedre udgangspunkt for at løse den reelle opgave.

Der virker til at være en forståelse på tværs af årgange for, at eleverne fra sjette årgang ikke har svarene på alt, selvom de har haft programmering før. De yngre elever kræver ikke svar fra de ældre elever, og de ældre elever virker ligeledes ikke til at skamme sig over, når de ikke kan hjælpe de yngre elever. I gruppe J må D6.21 eksempelvis erkende, at han ikke kan hjælpe P29 med et problem og siger at *“det er rigtig svært og det har vi ikke selv lært endnu”* (Bilag 6, s. 62). Det varierer hvor pædagogiske elev-underviserne er; hvor nogle er opmærksomme og opsøgende for at hjælpe, er andre enten ikke opmærksomme eller har en tendens til at overtage de yngre elevers arbejde. I Gruppe D hjælper P6.17 eksempelvis D9 kort med et problem, men lader ham arbejde videre med det selv:

P6.17 ser på D9s skærm og bukker sig ned ved siden af ham og forklarer ham hvorfor noget i programmet ikke virker. Hun dirigerer ham i, hvordan han kan få det til at virke, hvorefter hun går videre og lader ham arbejde videre med det selv. Lidt efter spørger hun “virker det?” og D9 svarer begejstret “Ja, den hopper nu” (Bilag 6, s. 58).

P6.17 giver på denne måde D9 mulighed for selv at arbejde med sit problem uden at overtage hans arbejde eller overlade ham til sig selv. Dette ses også i Gruppe C, hvor de ældre elever ligeledes er opmærksomme på at være tæt på de yngre elevers arbejde og løbende går rundt og kigger på deres skærme eller spørger, om de mangler hjælp. Modsat har nogle af elev-underviserne svært ved at være guidende i deres hjælp og har en tendens til at overtage arbejdet for de elever, som de hjælper. I Gruppe G hjælper D6.12 eksempelvis D30 med at løse et problem i hans program og overtager computeren for at løse problemet for den yngre elev i stedet for at instruere eller guide ham i selv at løse problemet (Bilag 6, s. 60). Dette sker også i Gruppe K, hvor D6.25 hjælper D42 med et problem i spillet, og overtager computeren for at løse det, men hvor det halvvejs går op for D42, hvordan han skal gøre hvor han så griber fat i computeren og overtager den tilbage (Bilag 6, s. 63). Dette viser et engagement fra begge parter, men et eksempel på, at eleven fra sjette har svært ved at lade den yngre elev nå sine egne erfaringer og nøjes med at instruere i stedet for at overtage.

På tværs

Der er på tværs af de tre cases kun kort fælles introduktion, hvor eleverne opfordres til at få praktisk erfaring med opgaverne. Der er stor forskel på lærernes tekniske kunnen i de tre cases og hvor meget de var i berøring de selve kodningen. Konsulenterne indgår generelt som teknisk støtte for lærerne i alle cases og er i tæt berøring med elevernes kodning. Læreren i Case 2 er meget nysgerrig på teknikken og går derfor ind og har berøring med selve koden og hjalp så vidt muligt i samme grad som konsulenterne. I Case 1 og 3 er lærernes rolle mere pædagogiske, hvor de er afhængige af elevernes samarbejde, brugen af ældre elever eller konsulenterne til at håndtere den tekniske kodning. Ved at bruge ældre elever som undervisere i Case 3, kommer der større teknisk hjælp til de lærende

elever, hvor der ikke forventes at elev-underviserne kunne hjælpe med alting. Nogle elev-undervisere har dog her svært ved at tilpasse deres undervisning til de yngre elevers behov, hvilket enten kan være grundet manglende interesse eller pædagogiske evner.

5.1.6 FAKTORER DER PÅVIRKER LÆRINGSSITUATIONEN

Eleverne i alle tre cases var i høj grad optagede af deres arbejde med programmeringsopgaverne uafhængigt af, hvordan forløbet var opbygget, men der ses dog nogle konsekvenser for læringssituationen ud fra forløbenes forskellige opbygning. Ved virksomhedsopbygningen i Case 1 og Case 2, var eleverne fordybede i gruppearbejdet og samtidig produktorienterede. Her var der dog stor forskel på, om eleverne fokuserer på kodningen eller på andre dele af projekterne såsom at skrive profilttekster for virksomhedens medlemmer eller udarbejde reklamemateriale. I begge cases var eleverne interesserede i fordelingen af opgaver både i deres individuelle arbejde og gruppearbejdet, hvor uddelingen af opgaver internt i grupperne foregik demokratisk. På denne måde giver virksomhedsopbygningen plads til stor interessebåret læring, hvor eleverne får mulighed for at opsoge de dele, som de selv finder interessant, men det har samtidig den konsekvens, at eleverne ikke opnår lige stor berøring med programmeringen. I Case 2 skulle eleverne indledningsvis arbejde selvstændigt med programmering af tutorials i det nye program, hvilket sikrede, at alle eleverne havde berøring med koden. I Case 1 og 2 var eleverne under tidspres for at færdiggøre deres projekter, hvilket indikerer en problematik i, eleverne fratages frihed til at benytte den nødvendige mængde tid, for at færdiggøre deres projekter, hvilket både kan handle om forløbets opbygning, lærernes forventninger eller elevernes egne ambitioner og opgaveprioritering i forhold til den tid, som er til rådighed. I Case 3 skulle eleverne, som har gennemgået Coding Class-forløbet, introducere yngre elever til programmering under observationsdagen. De ældre elever, som havde rollen som elev-undervisere, viste generelt et stort ansvar over for de yngre elevers læring og havde stor berøring med de yngre elevers kodning. Det varierede dog, om elev-underviserne udviste engagement i at undervise i programmering, men de yngre elever var alle optagede af programmeringsarbejdet. Ved denne opbygning havde de yngre elever mulighed for at udforske og stifte bekendtskab med programmering med tæt støtte fra ældre elever, hvor der ses et stort sammenhold og fælles problemløsning på tværs af årgangene. Brug af elev-undervisere giver således god støtte til de yngre elevers arbejde, men ikke alle elever er nødvendigvis interesserede i og engagerede i at undervise andre elever i programmering. De yngre elever i Case 3 udviste, modsat eleverne fra Case 1 og 2, ikke frustration over, at dagen sluttede og deres projekter skulle gøres færdige, hvorfor opbygningen i Case 3 kan have muliggjort en mere fri læringssituation for de lærende elever, men have konsekvens for undervisende elevers engagement. Der er således en konflikt mellem elevernes frihed og interesse i forhold til styringen af elevernes aktiviteter.

5.2 CODING CLASS I LÆRERNES PERSPEKTIV

Følgende analyse behandler lærernes erfaringer med Coding Class ud fra undersøgelsens interviews (Bilag 11, 12 og 13). Forud for analysen er interviewene kodet (Bilag 14, 15 og 16) med udgangspunkt i empirisk udledte kategorier omhandlende rammerne i Coding Class, undervisningens opbygning, faglige mål og udbytte, elevernes oplevelser samt lærernes oplevelser. Først analyseres lærernes generelle erfaringer med forløbene i hver enkelt case, hvorefter tendenser på tværs af de tre cases analyseres, for at synliggøre lærernes perspektiver på Coding Class-forløbet samt programmeringens faglige relevans for deres undervisning.

5.2.1 LÆRERERFARINGER FRA CASE 1

Lærer SB giver udtryk for, at hun *“synes det har været et godt forløb for alle”* (Bilag 11, l. 368). Hun fortæller, at *“jeg synes ikke der er nogle der er blevet tabt, ikke de fagligt svage i hvert fald”* (Bilag 11, l. 373), og uddyber, at de har set nogle af de fagligt svage elever blomstre i forløbet. De fagligt stærke elever har ikke nødvendigvis fået nær så meget ud af forløbet, som hun havde forventet, men hun fortæller, at *“det har været godt at se nogle af dem, som vi måske ser som stærke fagligt, også blive udfordret på en anden måde”* (Bilag 11, l. 368). Alle elever er således blevet udfordret på forskellig vis under programmeringsarbejdet i forløbet. Hun giver ligeledes udtryk for, at eleverne har været engagerede i forløbet: *“jeg synes de har været tændte på det. Og jeg har en oplevelse af at de alle sammen har syntes det har været sjovt”* (Bilag 11, l. 292). Elevernes engagement og passion kom til udtryk forskellige steder, hvor Lærer SB fortæller, at hun eksempelvis brugte de fiktive penge som belønningssystem, der motiverede eleverne, og de opsøgte derfor ekstraopgaver, fordi de syntes det var sjovt at tjene penge til deres virksomheder (Bilag 11, s. 120). Virksomhedsopbygningen med regnskaber medførte, at eleverne opsøgte flere opgaver, men i et fremtidigt Coding Class-forløb ønsker Lærer SB ikke nødvendigvis anvende samme format, da det var en udfordring for hende at have overblik over gruppernes økonomi (Bilag 11, s. 120). Denne problematik tyder på, at læreren enten skal holde styr på elevernes økonomi eller det skal tillades, at grupperne ikke fokuserer lige meget på regnskabet. Lærer SB oplevede ligeledes en motivationsfaktor i de forskellige roller i virksomhederne, hvor eleverne kunne udvælge deres egne opgaver, eksempelvis foretrak pigerne kreativt arbejde med analoge redskaber, hvor drengene blev tændte på at designe og være kreative med computeren (Bilag 11, s. 119). Hun var overrasket over, hvor grebet især drengene blev af kreativt arbejde med computeren: *“det gik de ret meget op i og nogle af dem sad faktisk og var meget nørklende og detaljeorienteret med det, meget mere end vi har set dem fordybe sig tidligere. Så det var en super fed oplevelse”* (Bilag 11, l. 276). Nogle af drengene var så engagerede, at de lavede et netværk i fritiden, hvor de udviklede spil for sjov sammen (Bilag 11, s. 121). Drengenes engagement

medvirkede ifølge Lærer SB til, at de ofte tog over med programmeringsarbejdet i de fleste grupper, og nogle piger fik derfor samme praksiserfaring med koden som drengene. Hun vil derfor fremadrettet opdele virksomhederne i mindre grupper med enten tvunget rolleskift eller med ligeligt fordelte opgaver, så hun kan sikre sig, at alle eleverne får berøring med koden (Bilag 11, s. 119). I et fremtidigt forløb vil hun forsøge at sammensætte grupper udelukkende bestående af piger for at give dem mulighed for at få mere praktisk erfaring med programmeringen (Bilag 11, s. 119). Hun udtrykker altså et behov for at have mere styring over gruppernes roller på trods af, at det kan gå ud over nogle af elevernes interesser, idet de dermed ikke kan vælge deres arbejdsopgaver ud fra deres egne interesser. Med større styring over rollefordelingen kan hun ligeledes i højere grad styre elevernes læring og forsøge at sikre, at alle elever lærer noget i Coding Class. Hun påpeger dog, at elevernes udbytte af forløbet varierer: *“sådan er det nok i de fleste undervisningsforløb, at nogle får mere ud af det end andre og nogle har også fået mere ud af det end vi havde forventet”* (Bilag 11, l. 338). Derfor mener hun, at introforløbet fremadrettet bør forlænges, så eleverne kan få en bedre basis for at arbejde mere selvstændigt ude i grupperne (Bilag 11, s. 117). I forbindelse med introforløbet viste elever fra Skibet Skole deres projekter frem for hendes klasse, hvilket var konstruktivt for eleverne, idet de fik mulighed for at se, hvad kodning var for noget, da nogle intet vidste om det på forhånd, og det ligeledes gav eleverne blod på tanden (Bilag 11, s. 113). Lærer SB giver således udtryk for, at et længerevarende introforløb både kan give eleverne bedre forståelse for programmering, så de i højere grad kan arbejde selvstændigt senere i forløbet, og det kan ligeledes motivere eleverne at se andre elevers arbejde og dermed programmerings muligheder.

Lærer SB's holdning til de faglige mål og opfyldelsen heraf varierer; det er nødvendigt for læreren, at forløbet opfylder de danskfaglige mål, og Lærer SB er gået til forløbet med det udgangspunkt at: *“hvis man vælger at bruge kodningen i et danskforløb, så skulle man gerne kunne tikke nogle fællesmål af”* (Bilag 11, l. 492). Hun havde først sat sig som mål om at koble danskfaglige mål som genre eller grammatik på elevernes arbejde med programmering men valgte en virksomhedsopbygning, hvor eleverne skulle designe spil. Ved denne opbygning blev danskfaglige mål om fremstilling og præsentation opfyldt, men selve programmeringsdelen i undervisningen bidrog ikke hertil:

Man kan sige, at det vi endte ud med, der er jo ikke noget danskfagligt i nogle af de spil de har lavet overhovedet. Det er nogle rigtig fine spil de har lavet, men det er jo alt det i virksomheden og alt det udenom, som så er blevet det danskfaglige (Bilag 11, l. 475).

Den danskfaglige relevans ligger således ikke i selve kodningen men i de aktiviteter, som forløbet er opbygget omkring. Programmeringen bliver dermed et middel til at arbejde med andre danskfaglige mål, men Lærer SB er fortsat opsat på, at programmeringsdelen kan blive relevant for danskfagets

mål: *“der har jeg nok ikke helt sluppet tanken om, at jeg godt vil arbejde med at få lavet nogle grammatikspil eller lave et genrespil”* (Bilag 11, l. 478). Lærer SB har således en interesse i at kombinere programmering og danskfaget, men hun udtrykker dog en begrænsning i forhold til sine egne evner:

Vi har fået koblet nogle danskfaglige mål ind i det, men ikke i den grad som jeg gerne ville. Men det tror jeg handler om, at jeg ikke har kundskaberne i kodningen endnu, til at kunne tænke i de baner (Bilag 11, l. 701).

I Lærer SB's perspektiv bliver hendes egne manglende evner en begrænsning for at bruge programmering på en relevant måde i sin undervisning. Hun havde forventet i højere grad at få hands on erfaring med programmering og mulighed for at udvikle sine egne kompetencer under forløbet (Bilag 11, s. 114). Hun fortæller, at lærerne ikke selv har tiden til at sidde med kodningen men understreger vigtigheden af, at lærerne får mulighed for at udvikle deres kompetencer i forhold til et forløb som Coding Class, da lærerne ikke på forhånd har kompetencerne til at håndtere et forløb som Coding Class (Bilag 11, s. 112). Hendes manglende viden og erfaring bliver en begrænsning for at kunne undervise i programmering: *“i forhold til kodning og programmering har jeg bare ingen basis overhovedet, så jeg har ikke så meget at trække på”* (Bilag 11, l. 419). Hun udtrykker et behov for at opbygge konkrete kompetencer til at varetage undervisningen i programmering og at hun har manglet en præsentation af programmerings muligheder, de forskellige programmeringssprog og måder, hvorpå de kan anvendes i undervisningen (Bilag 11, s. 125). I Lærer SB's perspektiv skal lærere undervises i det, de underviser i, *“så man ikke får stillet et krav op til noget, som man ikke selv kan hjælpe dem [eleverne] med i sidste ende”* (Bilag 11, l. 535). Det er altså problematisk, hvis lærere skal undervise i noget, de ikke har kendskab eller kundskaber indenfor, især hvis de samtidig ikke får mulighed for kompetenceudvikling eller videreuddannelse.

5.2.2 LÆRERERFARINGER FRA CASE 2

Lærer HG udtrykker, at både han og eleverne synes, det er sjovt at arbejde med programmering i undervisningen (Bilag 12, s. 136). Han fortæller, at elevernes motivation kommer til udtryk på flere måder, da *“de kan jo godt lide at arbejde på en computer. De kan også godt lide at det lykkedes for dem, og at deres opgave giver mening”* (Bilag 12, l. 92). Det er således både arbejdet med computeren, samt at det er meningsfulde opgaver, hvor eleverne kan se en sammenhæng med det de laver i virksomhederne, hvor produktet samtidig bliver godt. Han fortæller eksempelvis, at eleverne *“hader Scratch, fordi de ikke laver noget, som er godt”* (Bilag 12, l. 317), og de har ligeledes brugt JavaScript, men *“det bruger de ikke, for det er for svært at huske [...]”. Det siger dem ikke så meget,*

det tager for lang tid for dem at lave noget, der er fedt” (Bilag 12, l. 535). Det er således vigtigt, at eleverne synes, at opgaverne og projekterne er sjove, og at produktet ligeledes bliver godt, men læreren prioriterer her produktet over over den mest passende sværhedsgrad. Ligeledes oplever lærer HG, at muligheden for, at eleverne kan vise deres arbejde frem, er en motivationsfaktor for dem (s. 18). Lærer HG understreger, at han ønsker at udfordre eleverne i deres programmeringsarbejde, hvorfor han bevidst stiller høje forventninger til dem:

Og som jeg siger til mine elever, så er der aldrig nogen, der kan mere end jeg forventer, for så har jeg forventet forkert. I kan måske mere end ministeriet forventer (...), men jeg sætter dem så højt, at de ikke kan nå dem. Så er der noget at stræbe efter, og det kan de fleste faktisk godt lide (Bilag 12, l. 221).

Lærer HG forventer således mere, end det er muligt for eleverne at opnå, da de derfor har noget at stræbe efter, hvilket han oplever som en motivationsfaktor for eleverne. Han fortæller også, at både de fagligt svage og stærke elever i høj grad får et læringsudbytte ud af forløbet, men deres udvikling heri varierer: *“Når de kommer til det her, for der starter de tit samme sted. Og så er der bare nogle der får en interesse for det, og andre gør ikke”* (Bilag 12, l. 228). Elevernes udbytte af undervisningen handler ifølge Lærer HG altså om, hvorvidt eleverne får interesse for programmeringen og de aktuelle opgaver. Eksempelvis nævner han to elever, som normalt er blandt de fagligt svage elever men som har blomstret i forløbet og blandt andet har lavet ekstra spil til yngre årgange (Bilag 12, s. 149). Elevernes fagspecifikke udbytte kan således ikke forudsiges på forhånd, og elever, som normalt ses som fagligt svage, kan få mere ud af forløbet end forventet.

Coding Class-forløbet kører i Lærer HG’s natur/teknik- og matematikundervisning, men dette har mindre betydning for ham: *“jeg har en god fornemmelse af, hvad de sådan skal kunne, for at kunne klare sig godt. Og det når vi nok. Selvom vi måske ikke når lige det der mål med at tegne en trekant for eksempel”*. Lærer HG giver udtryk for, at forløbet især understøtter danskfaglige mål og i mindre grad er fyldestgørende for de fagspecifikke mål i natur/teknik og matematik (Bilag 12, s. 139). Han har ikke stort fokus på de fag, som forløbet er tilknyttet, men går i højere grad op i, om eleverne tilegner sig generelle tværfaglige kompetencer:

Jeg synes det er sjovt, og jeg synes det er givende for eleverne. Jeg synes de er gode til at tage imod det, og jeg synes de får meget ud af det. Deres måde at tænke på, fx hvis de har taget fejl, så retter de bare deres fejl. Det kan de også tage med i andre fag (Bilag 12, l. 344).

Forløbet opleves således ifølge lærer HG som givende for eleverne, hvor især deres problemløsning styrkes, hvilket er brugbart i andre fag. Forløbet har dermed fungeret som et tværfagligt forløb med

fokus på tilegnelse af generelle kompetencer, på tværs af de fag som forløbet oprindeligt var knyttet til.

Lærer HG virker samtidig ivrig og interesseret i teknikken og udtrykker flere gange, at han synes, det er sjovt at arbejde med programmering: *“det har bare været skide skægt”* (Bilag 12, l. 88), *“det bliver ret sjovt”* (Bilag 12, l. 105) og *“så det er ret spændende”* (Bilag 12, l. 146). Han udtrykker sit eget engagement for programmering gennem interviewet, men giver udtryk for, at hans egne kompetencer ikke altid er tilstrækkelige. Hans udgangspunkt for at indgå i forløbet var dog, at hvis *“elever ikke kunne, så jeg vidste jo at jeg var bedre end dem til at starte med. Så måtte vi jo bare tage den derfra”* (Bilag 12, l. 285). Hans interesse for programmeringen virker således til at overskygge bekymringen om, hvorvidt han har kompetencer til at undervise i det, idet konsulenterne kan hjælpe eleverne, hvis han ikke selv kan, og kan øge sværhedsgraden i elevernes arbejde (Bilag 12, s. 136). Han har dog kunne bidrage ligesom konsulenterne, da priserne for at få hjælp af konsulenterne og ham selv har varieret afhængig af, hvem der var bedst til de programmer, de har arbejdet med (Bilag 12, s. 136). Han giver dog udtryk for, at han ikke som udgangspunkt har kompetencerne til at undervise i programmering: *“eleverne overhaler jo mig helt vildt hurtigt. Jeg har jo ingen forudsætninger for det her, jeg synes bare det er sjovt”* (Bilag 12, l. 84). Han har af samme årsag inddraget elever fra andre klasser, som har særlige forudsætninger eller evner til at støtte i undervisningen (Bilag 12, s. 138). Han forholder sig således afslappet til sine egne kompetencer og virker ikke forstyrret af, at eleverne bliver bedre end ham, eller at han ikke altid kan hjælpe dem. Det beskriver han dog som en mulig forudsætning for undervisningen med programmeringen:

Når man arbejder med programmering, bliver du også nødt til at arbejde innovativt på en eller anden måde. Så du præsenterer dem for noget, hvilket ofte er en åben opgave, hvor du ikke ved hvordan den skal løses (Bilag 12, l. 354).

Han lægger således vægt på, at det er krævende for en lærer at arbejde med programmering, da der ikke er noget facit og elevernes arbejde udmunder i noget uforudsigeligt og individuelt. Han giver ydermere udtryk for, at dette kan være stressende for nogle lærere og foreslår, at der udformes lukkede opgaver, hvis læreren foretrækker at kende svaret på forhånd (Bilag 12, s. 145). Lærere vil ifølge ham have det bedste udgangspunkt for at undervise i programmering, hvis de ikke presses til det, men at *“jo større værktøjspakke en lærer har, jo nemmere er det at vælge, hvad de skal kunne. Og hvordan man lærer de her sider at kende”*. (Bilag 12, l. 455). Han udtrykker en interesse i at deltage på kurser, hvis han ved, at han kan dygtiggøre sig selv og udvikle sine kompetencer, men lægger meget vægt på at det skal kunne bruges i praksis og være relevant for sin undervisning (Bilag 12, l. 403).

5.2.3 LÆRERERFARINGER FRA CASE 3

I Case 3 havde de arbejdet med programmering af spil, uden brug af virksomheder, under de fire dage med Coding Class. Konsulenterne havde planlagt undervisningen, da lærerne ikke havde kompetencerne til det:

Det, som Mads og Christian kom med, var garanteret bedre end det, vi selv ville kunne opfinde. Så det har været meget dem, der skulle bestemme indholdet i og hvad de synes, for vi ved jo ikke hvad der var af muligheder. Så vi tænkte, at det ved de noget om (Bilag 13, l. 96).

Samtidig udtrykker lærerne dog, at konsulenternes planlægning ikke altid var fyldestgørende, da *“det var tit noget, der ikke var sådan vildt planlagt, når de kom”* (Bilag 13, l. 115). Lærerne havde dog ikke erfaring med programmering, hvorfor de ikke havde udtalte forventninger til forløbet og konsulenternes arbejde, men Lærer AS udtrykker, at forløbet må være vellykket så længe *“børnene har lært noget og de har haft det sjovt”* (Bilag 13, l. 388). De har således brugt konsulenterne til at overtage undervisningen og givet dem ansvaret for elevernes læring. Lærerne havde ikke selv tilmeldt sig forløbet men indvilgede i at deltage, fordi *“vi skulle jo ikke bremse noget, som eleverne kan have glæde af”* (Bilag 13, l. 14). På trods af, at eleverne i lærernes optik fik glæde af forløbet, fremhæver begge lærere, at de ikke fremadrettet ønsker at benytte samme opbygning, som ellers har fungeret som et *“frisk pust udefra”* (Bilag 13, s. 167). Dette begrundes de med, at de er underlagt faste regler om målstyring, hvor de skal indføre læringsmål og tegn på læring for hvert undervisningsforløb. Lærerne understreger flere gange gennem interviewet, at det er vigtigt at programmeringen er relevant for de matematiske fagmål (Bilag 13, s. 156, 160, 164 og 165), og de oplevede ikke, at forløbet og dets opbygning bidrog nok til det fagspecifikke udbytte:

Jeg ville ikke igen, i matematik, på min nuværende 5. årgang, bare fyre en hel dag af, for fagligt giver det ikke helt så meget. Det gør det på mange andre parametre, men vi kan ikke vinge noget af i vores stof og kompetencer (Bilag 13, l. 196).

Eleverne får således gavn af programmeringen, dog ikke med de fagspecifikke mål. Lærer JL synes derfor ikke, at forløbet har været meningsfuldt for de faglige aspekter i matematikundervisningen. Lærer JL udtrykker dog interesse i at bruge programmering i sin undervisning fremadrettet, men dette skyldes, at hun har deltaget i et kursus om brugen af programmering i matematikfaget (Bilag 13, s. 165). Kurset giver hende mulighed for at bevare det friske pust, som de fortæller, at forløbet har været for undervisningen men med fokus på fagligt relevante emner:

Så kan jeg jo, når jeg skal have et stofområde, lige starte med at kigge på kopierne, og se hvordan det nu lige var nogle gode idéer, at arbejde med kodningen indenfor det. Det kunne være en sjov effekt, at kunne give en anden vinkel på det stofområde, man underviser i (Bilag 13, s. 193).

Lærer JL ser altså en funktion og nytteværdi i at inddrage programmering i matematikundervisningen, hvis hun har konkrete redskaber til rådighed, som inddrager kodning på en måde, der er relevant for det faglige stofområde. Hun giver udtryk for, at det er gennem kurset, at hun har tilegnet sig kompetencerne til at integrere programmering i matematikundervisningen (Bilag 13, s. 160). Lærerne ser ligeledes en fordel i at inddrage programmering i relation til elevernes praksis, som syntes, det var sjovt at arbejde med vinkler, længder og rektangler i Scratch: *“det er da lidt sjovere at lave, end bare at skulle tegne den på papir (...) Det er da en leg i sig selv”* (Bilag 13, l. 208). Hertil uddyber Lærer AS, at det ligeledes er en fordel i forhold til elevernes egne præferencer:

Det passer mig rigtig godt at gøre det på den måde, at man netop finder nogle, der ikke fanges af papir, men som fanges af det andet. Så får man begge med. Så de er interesseret i det der foregår, på den måde, som de bliver fanget af det (Bilag 13, l. 212).

Lærer AS udtrykker her en forståelse for, at eleverne skal have plads til at lære på deres egne præmisser, og at elevernes fanges af noget forskelligt. Lærerne har en oplevelse af, at eleverne generelt har været optagede af forløbet, selvom det krævede meget for eleverne, og som de nævner, vil der altid være nogle, som har svært ved at koncentrere sig (Bilag 13, s. 161). Under forløbet var omkring 60 elever fra sjette årgang samlet men på trods af dette var det ikke nødvendigt for lærerne at holde øje med dem (Bilag 13, l. 304). På trods af at tre klasser var slået sammen, var det ikke udfordrende at holde dem beskæftigede og koncentrerede. Deres oplevelse var derimod, at eleverne var engagerede i forløbet:

Eleverne har været virkelig optaget af det [...]. De har haft nogle rigtig gode dage. Altså, der er nogle der næsten ikke kunne vente til næste gang. De har glædet sig meget. Jeg tror også de har hygget sig med, at lave de her spil og prøve hinandens spil af. Sådan at få det bygget op og få gjort det bedre og bedre (Bilag 13, l. 102).

Eleverne var altså i høj grad engagerede i at skabe deres spil og var især optagede af at forbedre deres spil samt fremvise og prøve hinandens. Nogle få grupper fungerede ikke optimalt, hvilket kan være fordi, de selv havde sammensat deres grupper (Bilag 13, s. 163). Lærerne nævner at ballademagerne

havde fundet sammen, hvor disse elever ville have haft svært ved at koncentrere sig uanset opgaven (Bilag 13, s. 163).

I forlængelse med forløbet arrangerede lærerne en ekstra Coding Class-dag i foråret, hvor eleverne fra sjette årgang skulle undervise eleverne fra fjerde og femte årgang i programmering. Formålet med den ekstra dag havde flere mål, og lærerne udtrykker blandt andet, at det er en fordel, at elever underviser elever, for *“så finder de ud af, hvad de selv kan”* (Bilag 13, l. 320), samt at *“det kunne være sjovt, at de kunne videregive det til 4. og 5. klasse, som skal til og ind i det der kodning også”* (Bilag 13, l. 156). Den ekstra dags formål var ligeledes rettet mod skolens lærere, blandt andet ved at *“få så få lærere til at have så mange elever som muligt”* (Bilag 13, l. 154) og fordi lærerne *“kigger jo på hvad der foregår og kan blive lidt inspirerede”* (Bilag 13, l. 435). Den ekstra dag tjener således både det formål, at eleverne, der har været igennem forløbet kan blive mere selvsikre i deres programmeringsevner, at de nye elever kan blive introducerede til det inden, de starter på deres eget forløb, at der kan spares i mandetimer i forhold til mængden af undervisning, samt at andre lærere kan blive inspirerede, når de oplever det i praksis.

5.2.4 FAKTORER DER PÅVIRKER PROGRAMMERINGS FAGLIGE RELEVANS

Lærerne i de tre cases havde forskellige kompetencer og erfaringer med programmering i deres undervisning, men flere faktorer påvirkede henholdsvis deres overordnede erfaringer med forløbet samt deres vurdering af programmerings faglige relevans for deres undervisning. Disse faktorer inkluderede forløbets opbygning, elevernes udbytte, de fagspecifikke mål samt lærernes egne kompetencer, hvilket gennemgås nedenfor.

Forløbets opbygning

På tværs af de tre cases udtaler lærerne sig positivt omkring elevernes koncentration, engagement og interesse for deres produkter. Lærerne i Case 1 og 2 var af den opfattelse, at virksomhedsopbygningen motiverede deres elever, og begge lærere gav udtryk for, at eleverne især blev motiverede af at kunne tjene penge til virksomhederne. I Case 1 ville eleverne have ekstraopgaver, så de kunne tjene flere penge til virksomheden, og i Case 2 medvirkede virksomhederne til at give de individuelle opgaver mening for eleverne. I begge cases erfarede lærerne dog, at nogle elever prioriterede at tjene penge til virksomheden frem for at arbejde med programmeringen, hvilket viser en problematik mellem elevernes egen interesse og opfyldelse af undervisningens fagspecifikke mål.

I alle tre cases benytter lærerne sig af elevundervisning på forskellige måder, hvor lærerne i Case 1 og 2 benytter ældre elever til at støtte lærerne i programmeringsundervisningen og hjælpe de lærendes elevers arbejde med kodningen, når konsulenterne ikke er til stede. I Case 3 benyttede lærerne sig af

elevundervisning under en ekstra dag uden for Coding Class-forløbets egentlige ramme. Deres hensigt var at give eleverne større selvtillid omkring deres egne kompetencer, og samtidig introducere de yngre elever til programmering. Brug af ældre elever i programmeringsarbejdet havde en støttende effekt for lærerne, da de således ikke nødvendigvis behøvede at kunne løse løbende problemer i elevernes kode.

Elevernes udbytte

Læreren i Case 2 oplevede, at eleverne blev motiveret af en høj succesrate med programmeringsopgaverne og det at skabe gode produkter. I Case 2 valgte læreren derfor at benytte nemmere programmer, hvor produkterne er gode og succesraten høj, og han har derfor større fokus på, at eleverne får succesoplevelser med programmering frem for et stort læringsudbytte. Dette viser en problemstilling i, at elevernes faglige udbytte nedprioriteres i forhold til deres engagement og motivation, hvor elevernes faglige udbytte derfor afhænger af, hvad elevernes selv finder interessant og opsøger i deres læreproces. Elevsammensætningen ses også at have en påvirkning på elevernes udbytte, hvor der både i Case 1 og 2 ses en tendens til, at enten dygtige eller interesserede elever, har størst berøring med programmeringen, hvor de andre elever i højere grad fokuserer på eksempelvis det visuelle udtryk. Der er således en problemstilling i at elevernes læringsudbytte varierer, fordi eleverne fordeler rollerne i grupperne afhængig af interesse og kompetence, hvorfor eleverne, som har sværest ved at programmere, ofte ikke får særlig stor erfaring med programmeringsopgaverne i grupperne. I Case 1 ønsker læreren fremadrettet at lave mindre og eventuelt kønsopdelte grupper for at sikre at alle eleverne får berøring med koden. I Case 3 har lærerne ikke oplevet problemer med at have overblik over tre klasser i forløbet på trods af at, de er op til seks gange så mange elever samlet som i Case 1 og 2, da eleverne ifølge lærerne har været optagede af forløbet. Lærerne oplevede dog, at enkelte af selvvalgte grupper ikke fungerede, men de mener ikke, at dette er koblet til programmeringen som element i undervisningen men derimod gruppesammensætningen, da disse eleverne generelt har svært ved at koncentrere sig. På tværs af de tre cases ses altså en tendens til at elevernes gruppesammensætninger samt rollefordelinger heri kan påvirke elevernes læringsudbytte.

Fagspecifikke mål

Lærernes erfaringer med Coding Class i relation til fagspecifikke mål varierer på tværs af de tre cases. Det er især danskfaglige mål, som synes at blive opfyldt i både Case 1 og 2, selvom Case 2 inddrager forløbet i natur/teknik og matematik. I Case 1 er forløbet koblet på danskundervisningen, men gennem virksomhedsopbygningen erfarer læreren, at der ligeledes kobles matematikfaglige mål på, især i forbindelse med virksomhedernes regnskab og selve programmeringen. Det er således de omkringliggende aktiviteter udenfor koden, der i Case 1 opfylder de danskfaglige mål, herunder mål som involverer feedbackprocesser, projektarbejde og præsentationer. I både Case 1 og 2 bliver

inddragelsen af programmering i deres undervisning således automatisk et tværfagligt forløb, hvor både naturvidenskabelige fag og humanistiske fag er repræsenteret. Læreren i Case 1 foreslår, at benytte programmering som del af den understøttende undervisning, da det ikke bidrager direkte til de danskfaglige mål, men oplever en relevans for at programmering kan understøtte undervisningens som tværfagligt element. I Case 3 er forløbet udelukkende koblet på matematikundervisningen, hvor lærerne ikke oplever, at programmeringen har bidraget til indfrielse af nogle fagspecifikke mål overhovedet, men har en oplevelse af at eleverne har lært noget og haft det sjovt. Lærerne i Case 3 ser dog en fordel ved at bruge programmering som et redskab til at arbejde med matematiske temaer, fordi den ene lærer har fået kendskab hertil gennem et kursus specifikt målrettet programmering i matematikfaget. De ønsker derfor ikke fremadrettet at arbejde med programmering i matematikundervisningen på samme måde som under Coding Class-forløbet, som de dog har brugt til inspiration for fremadrettet arbejdet med matematikfaget. Der ses således på tværs af de tre cases en problemstilling i, hvorvidt de faglige mål opnås, og om selve programmeringen har bidraget som fagligt element til lærernes undervisning. Lærerne giver udtryk for, at forløbet har gavnet elevernes mere generelle kompetencer, men at de fagspecifikke mål først opfyldes, når de kobles på helt konkrete temaer og mål for deres undervisning.

Lærernes kompetencer

Det er forskelligt, hvor stor berøring lærerne i de tre cases har med koden i Coding Class. Læreren i Case 1 ønskede eksempelvis at lære at kode selv, men der viste sig ikke at være tid til dette, hvorfor læreren følte sig afhængig af konsulenternes tilstedeværelse, selvom hun forsøgte at hjælpe eleverne, så godt som hun kunne. I Case 2 havde læreren større praksiserfaring med programmering og hjalp ligesom konsulenterne eleverne med deres kode. Han understregede dog, at hans evner hurtigt blev overhalet af elevernes, og at han ofte ikke kunne hjælpe dem, når problemer i koden opstod. I Case 3 havde lærerne ingen erfaring med programmering og havde ingen berøring med koden undervejs i forløbet, men overlod programmeringen til konsulenterne eller bad eleverne om at hjælpe hinanden. De valgte at lade konsulenterne planlægge undervisningsforløbet, og betragtede udelukkende Coding Class som inspiration til deres matematikundervisning frem for et fagligt bidrag. Nogle af lærerne var altså i høj grad afhængig af konsulenternes kompetencer, hvilket kan udgøre en fremtidig problematik, skulle de ønske at inddrage programmering i deres undervisning fremadrettet. På tværs af alle tre cases giver lærerne endvidere udtryk for en problematik i, at de ikke har et kompetencemæssigt udgangspunkt for at undervise i programmering. I Case 1 havde læreren forventet og håbet på en større personlig kompetenceudvikling samt videreuddannelse og mener ikke, at hun har evnerne til at undervise i programmering, da hun ikke er blevet uddannet til det, ligesom hun er i dansk. I Case 2 og 3 giver lærerne udtryk for, at førend de er interesserede i at deltage i et kursus, skal det have klare praktiske anvendelsesmuligheder for deres undervisning, så programmeringen kan

kobles direkte på de fagspecifikke mål og temaer. Det er således på sinde for lærerne på tværs af de tre cases, at de ønsker videreuddannelse og kompetenceudvikling, for bedre at kunne implementere programmering som fagligt element i deres undervisning.

5.3 DELKONKLUSION: CODING CLASS I UNDERVISNINGEN

Gennem analysen af læringssituationen samt lærernes erfaringer ses overordnede tendenser på tværs af de tre cases, herunder at læringssituationen er påvirket af et højt engagement blandt eleverne samt deres fordybelse i projekterne. Elevernes arbejde er ligeledes i alle cases i høj grad interessebåret med fokus på problemløsning og samarbejde, hvor gruppernes sammensætning og rollefordeling viste sig at have stor indflydelse på deres læringsudbytte. Virksomhedsopbygningen bidrager til at skabe et engagement hos eleverne og giver dem mulighed for at opsøge aktiviteter, de er motiverede til at udføre, men bevirker ligeledes en udfordring for lærerne i at styre elevernes læringsudbytte. Brug af elev-undervisere i programmeringsarbejdet, bidrager til en omfangsrig hjælp til de lærende elevers tekniske kodning, men bevirker ligeledes til en variation i de undervisende elevers engagement. Indsigten i lærernes erfaringer med Coding Class viser, at lærerne oplever besvær med at opfylde fagspecifikke mål ved inddragelse af programmering som fagligt element i deres undervisning, grundet deres begrænsede viden og kompetencer inden for feltet. Lærerne oplevede alle tvivl omkring egne kompetencer, hvor brugen af henholdsvis konsulenter og ældre elever som undervisere har den konsekvens, at lærerne støtter sig op af dem og således får mindre berøring med programmeringen. For at inddrage programmering som fagligt element i undervisningen ønskede lærerne på tværs af de tre cases videreuddannelse, hvor programmering forbindes direkte til deres undervisningspraksis og de fagspecifikke temaer og mål.

KAPITEL 6

PROGRAMMERINGS POTENTIALE

...

På baggrund af undersøgelsens analyse diskuteres i følgende kapitel indledningsvis spændingsfelter ved inddragelsen af programmering i undervisningen samt opmærksomhedspunkter, lærere skal være opmærksomme på i praksis. Herefter diskuteres potentialer og udfordringer ved at inddrage programmering i henholdsvis eksisterende fag og som selvstændigt fag i grundskolens undervisning, samt hvilke hensyn, der skal tages i arbejdet hermed. På baggrund af undersøgelsens resultater gives afslutningsvis konkrete forslag til videre undersøgelser, der kan føre debatten videre på området.

6.1 SPÆNDINGSFELTER VED PROGRAMMERING I UNDERVISNINGEN

I analysen belyses flere faktorerers indflydelse på læringssituationen samt programmeringens faglige relevans for undervisningen under Coding Class-forløbene. På baggrund af disse faktorer diskuteres, hvilke spændingsfelter som lærere står overfor i inddragelsen af programmering som fagligt element i deres undervisning, hvilket har til formål at markere hvilke opmærksomhedspunkter, der skal tages hensyn til i implementeringsarbejdet fremadrettet. Disse opmærksomhedspunkter opstilles afslutningsvis som en vejledning, så skoleledere og lærere kan træffe nogle bevidste til- og fravalg ud fra erfaringerne med Coding Class.

6.1.1 FAGSPECIFIKT KONTRA TVÆRFAGLIGT ELEMENT

Opfyldelsen af de faglige mål viser sig at være en udfordring på tværs af interviewene med de forskellige lærere i casestudiet. Ved Case 1, hvor programmeringen foregår i dansk, er det umiddelbart let, at koble danskfaglige mål på forløbet, men målene er koblet op på aktiviteter i virksomhederne, hvor læreren ikke oplever nogen direkte relation mellem programmeringen og fagets mål. De mål som blev opfyldt kunne umiddelbart være nået gennem arbejdet i virksomhederne med udarbejdelse af andre aktiviteter end programmering af spil. Aktiviteterne i virksomhederne og med programmering inddrager dog nogle matematiske mål, hvorfor forløbet automatisk blev et tværfagligt forløb selvom det kun var planlagt til danskfaget. I Case 2 er det lettere for læreren at koble danskfaglige mål på forløbet end de naturvidenskabelige fag, som forløbet er knyttet til, hvorfor det her ligeledes bliver et tværfagligt forløb i praksis. I Case 3, hvor forløbet er koblet på matematikundervisningen, lykkedes det ikke lærerne at koble matematiske fagmål på overhovedet, og de har først set programmeringens faglige relevans for matematikundervisningen gennem et videreuddannende kursus, hvor programmering blev knyttet tæt op på de fastlagte temaer i matematikfaget. De faglige mål har dog mindre betydning for læreren i Case 2 end i de resterende cases, da han oplever en større relevans for, at eleverne lærer nogle generelle kompetencer, som kan bruges på tværs af fag, eksempelvis udvikling af deres problemløsningsevner. Eftersom to ud af de tre forløb i Coding Class endte med at fungere som tværfaglige forløb, kunne det tyde på, at programmering bedre kunne bidrage som tværfagligt element og understøtte fagene på tværs herigennem. Det er således nødvendigt at tage stilling til, om brugen af programmering i undervisningen skal være et middel til at understøtte elevernes fagspecifikke kompetencer, som det ønskes i Case 1 og 3, eller mere generelle kompetencer, som ved eksemplet i Case 2. Lærerne i casestudiet udtrykker et behov for, at programmering kobles tættere sammen med de fagspecifikke

krav med tæt relation til fagenes fastlagte mål og temaer, for at programmering kan indgå meningsfuldt i de enkelte fag. Lærerne giver dog udtryk for, at deres manglende viden og kompetencer begrænser dem i, at kunne koble programmeringen aktivt til de fagspecifikke mål, hvorfor de alle ønsker videreuddannelse, hvis dette skal indfries.

6.1.2 ELEVERNES FAGLIGE UDBYTTE KONTRA ELEVERNES INTERESSE

Der hvor programmeringen bliver omtalt som en faglig fordel ved alle lærerne i casestudiet, er omkring elevernes fælles udgangspunkt for at arbejde med programmeringsopgaverne. Lærerne har en oplevelse af at forløbet med programmeringen har fanget alle deres elever og udligner forskellen mellem de fagligt stærke og de fagligt svage elever i deres undervisning. Lærerne udtrykker dog en udfordring i, at elevernes udbytte af programmeringsforløbet især afhænger af, hvorvidt eleverne udvikler en interesse for programmeringsopgaverne. I alle cases opleves stort engagement blandt eleverne, men hvor eleverne, som er gode til programmeringen eller har interesse herfor, har en tendens til at overtage kodningen i grupperne i Case 1 og 2, hvorfor ikke alle får lige meget berøring med kodningen. Andre elever arbejder derfor eksempelvis mere med virksomhedernes regnskab eller det visuelle udtryk, hvorfor eleverne i praksis beskæftiger sig med meget forskellige opgaver når programmeringsarbejdet er opbygget gennem virksomheder. Det har den konsekvens at eleverne ikke får lige meget berøring med koden og deres læringsudbytte ved den interessedrevne inddeling af roller i grupperne er forskelligt. På denne måde er elevernes udbytte i høj grad påvirket af elevernes forskellige interesser og gruppesammensætninger. Lærerne oplever her et spændingsfelt mellem elevernes læring og deres faglige målstyring, som er med til at sikre, at eleverne læring har kohærens til de læringsmål og fagmål som undervisningsministeriet har sat op for de konkrete fag. Læreren i Case 2 har eksempelvis valgt et lettere program, hvor eleverne udtrykker begejstring for produktet, og fravælger et sværere program, hvor eleverne ville blive udfordret mere fagligt. Her vægter læreren altså produktet og elevernes interesse højere end den faglige udfordring og elevernes læringsudbytte. På denne måde er det nødvendigt at vægte mellem elevernes interesse og lærernes styring af elevernes arbejde, som sikrer det fagspecifikke udbytte.

6.1.3 SELVVALGTE KONTRA LÆRERVALGTE GRUPPER

I alle cases oplever lærerne en udfordring i at elevernes faglige udbytte afhænger af deres gruppesammensætning og rollefordeling i grupperne, hvorfor det kan være relevant som lærer at gøre sig nogle bevidste overvejelser omkring, hvordan eleverne inddeles i grupper og hvordan rollerne mellem dem fordeles. Læreren i Case 1 oplevede problemer med at drengene ofte overtog programmeringsopgaverne i grupperne, hvorfor hun fremadrettet ønsker at sammensætte mindre grupper end fire elever sammen og muligvis kønsopdele dem, så hun kan sikre især pigernes berøring med programmeringsopgaverne. I Case 3 havde elev-underviserne selv inddelt sig i grupper, hvilket ifølge lærerne havde indflydelse på om grupperne var velfungerende, da ballademagerne automatisk satte sig sammen. Under observationerne viste elevernes engagement i Case 3 ligeledes meget varierende, hvilket kan være påvirket af, at antallet af elever var op til seks gange så mange, som ved de resterende cases. Det er således uvist om det er de selvvalgte grupper eller antallet af elever, som har indflydelse på, hvorvidt grupperne fungerer, men dette peger dog på, at der er behov for at tage nogle bevidst valg i forhold til hvorvidt grupperne skal sammensættes bevidst af læreren samt hvor mange elever, der skal være per gruppe samt hvor mange elever, der skal involveres i forløbet på en gang.

6.1.4 INDIVIDUELT ARBEJDE KONTRA GRUPPEARBEJDE

Ud over at overveje hvordan grupperne med fordel sammensættes, kan det ligeledes være relevant for lærere at overveje formatet for gruppearbejdet, herunder hvor længe og hvornår eleverne skal arbejde med programmering i grupperne samt graden af individuelt arbejde. I Case 1 oplevede læreren meget stor forskel på, hvad elevernes beskæftigede sig med og derfor hvad de fik ud af programmeringsforløbet, hvorfor hun ønsker afholde et længere introforløb, hvor eleverne skal arbejde selvstændigt med nogle faste opgaver og sikre deres berøring med koden, før de deles ud i mere frie projekter i grupperne. Under både Case 2 og 3 brugte henholdsvis konsulenterne og elev-underviserne tutorials til at introducere eleverne for nye programmer. Her skulle de arbejde selvstændigt med programmeringen før de arbejdede videre i grupperne, hvilket sikrede alle elever opnåede erfaring med det enkelte program samt fik berøring med koden. Det kan således være relevant at overveje, om eleverne skal opnå det samme læringsudbytte og samme praksiserfaring, hvilket kan opnås gennem større grad af individuelt arbejde. Det individuelle arbejde kan, som ved eksemplerne i Case 2 og 3, benyttes som introducerende forarbejde til elevernes mere frie arbejde i grupperne. På trods af gruppesammensætningerne, rolleinddelingerne samt forløbets opbygning, faldt elevernes engagement på tværs af de tre cases efter frokostpauserne på tværs af de tre cases, hvorfor forløbets opbygning over fire hele dage således ikke nødvendigvis, er det mest optimale for elevernes

læring. Det er således ligeledes både for lærere at tage stilling til hvordan grupperne sættes sammen, hvordan de vil balancere det individuelle og gruppeinddelte arbejde samt hvordan forløbene er opbygget.

6.1.5 EKSPERTROLLE KONTRA GUIDENDE UNDERVISER

Der ses på tværs af de tre cases en problemstilling i, at lærerne ikke føler sig kompetente til at undervise eleverne i programmering. I Case 1 og 3 har lærerne været på bar bund og overladt programmeringen under forløbet til konsulenterne fra IT-Branchen, som har kompetencerne til at arbejde med det forskellige programmeringsarbejde, hvorfor de fungerede som eksperter i undervisningen. I Case 1 og 3 har lærernes rolle derfor været mere pædagogiske, hvor de udtrykker stor afhængighed af tilstedeværelsen af de ældre elever eller konsulenterne til at håndtere den tekniske kodning. Læreren i Case 2 udtrykker stor begejstring for teknikken og vælger derfor at have praktisk erfaring med koden og de forskellige programmer, hvor han under observationerne indgik i problemløsningen sammen med eleverne. Selv her, hvor læreren er passioneret omkring den tekniske del af programmeringsopgaverne, udtrykker han en udfordring i, at han ofte ikke har evnerne til at hjælpe eleverne. Dette virkede dog ikke til at genere eleverne i Case 2, ligesom det i Case 3 ikke virkede til at eleverne fra fjerde og femte årgang var generede af at elev-undervisere fra sjette årgang, ikke var eksperter i programmering. Brugen af andre elever som undervisere bevirkede under observationsdagen blot, at de lærende elever i højere grad søgte hjælp ved hinanden, da der ikke var nogen forventning om, at de ældre elever havde svaret på alt. Dette viser at undervisere, elever som lærere, ikke nødvendigvis behøver at være alvidende, for at eleverne kan opnå progression i deres læreproces og deres konkrete arbejde med programmeringsopgaverne. Der ses således en udfordring i, at det er lærerne selv, der forventer, at de skal indgå som eksperter i klasseværelset og derfor ikke føler sig kompetente til at arbejde med programmering i deres undervisning, selvom eleverne i praksis ikke virker berørte af, at undervisere indgår i en fælles læreproces med dem. Læreren i Case 1 vil eksempelvis gerne kunne hjælpe eleverne med de opgaver, som hun stiller dem, men har ikke forudsætningerne til at gøre det eller tiden til at sætte sig tilstrækkeligt ind i programmerne. Læreren i Case 2 udtrykker dog, at forudsætningen for at arbejde med programmering er, at læreren er *“nødt til at arbejde innovativt på en eller anden måde. Så du præsenterer dem for noget, hvilket ofte er en åben opgave, hvor du ikke ved hvordan den skal løses”* (Bilag 12, l. 354). Læreren fra Case 2 er dog også den eneste af lærerne i casestudiet, som indgår i en fælles læreproces med eleverne, hvorfor det tyder på, at lærerne med passion for programmering har større tendens til at få berøring med elevernes kode i praksis og indgå eksperimenterende i programmeringsarbejdet. Lærerne i Case 1 og 3 udtrykker stor afhængighed af konsulenternes tilstedeværelse samt elevernes interne problemløsning, da de ikke har kundskaberne til at hjælpe eleverne selv. Der ses således en udfordring i hvor kompetente lærerne skal være til at guide elevernes arbejde, hvis de ikke har nogle basiskundskaber eller kendskab til kodning.

Det er således også svært for en guidende underviser, som ikke har nogen forståelse for kodning, at vurdere elevers progression og faglige udbytte med selve programmeringen, da de eksempelvis kun kan give feedback på og vurderes dette ud fra elevernes proces samt deres færdige produkter. Det er derfor relevant at overveje om lærere skal indgå som eksperter i sin egen undervisning, eller om det er tilstrækkeligt at de kan guide elevernes læreproces uden at have nogen forståelse for og erfaring med programmering i praksis.

6.1.6 VEJLEDNING TIL INDDRAGELSE AF PROGRAMMERING I UNDERVISNINGEN

På baggrund af de ovenstående udfordringer ved inddragelse af programmering som fagligt element i undervisningen udarbejdes herunder konkrete spørgsmål, der kan udgøre handleanvisninger samt overvejelser henvendt til lærere, hvis de ønsker at inddrage programmering i deres undervisning. Disse overvejelser tager hensyn til, at lærere kan have forskellige forudsætninger for programmeringsarbejdet, hvor formålet hermed således er at synliggøre spændingsfelter ved inddragelse af programmering i undervisningen, som bør tilpasses den konkrete læringssituation og lærernes egne præferencer.

OPMÆRKSOMHEDSPUNKTER TIL INDDRAGELSE AF PROGRAMMERING I UNDERVISNINGEN

- ▶ Skal programmering indgå som et fagligt element i ét enkelt fag eller indgå som et tværfagligt forløb, og i så fald med hvilke fag involveret?
 - › Et tværfagligt forløb kan give anledning til at bruge programmering som del af et større projekt eksempelvis gennem opbygning af virksomheder, hvor eleverne programmerer spil, fører regnskab og laver merchandise.
- ▶ Skal forløbet understøtte elevernes faglige udbytte eller elevernes interesser?
 - › Hvis forløbet skal sikre elevernes faglige udbytte kan du overveje, at opstille lukkede opgaver, hvilket dog kan mindske elevernes eksperimenterende og kreative udfoldelse.
- ▶ Skal eleverne selv inddele sig i grupper eller ønskes kontrol over gruppesammensætningerne?
 - › Gruppesammensætningen og antallet af elever kan have indflydelse på, hvor tæt berøring den enkelte elev får med programmering, og hvor godt grupperne fungerer.
- ▶ Skal eleverne have lige stor erfaring med kodningen, eller må de frit fordele arbejdet i grupperne?
 - › Hvis elevernes berøring med koden skal sikres, kan det være en fordel at, eleverne indledningsvis arbejder individuelt med programmering gennem eksempelvis tutorials, førend de arbejder kreativt i grupper.
- ▶ Skal du som lærer være ekspert i programmeringsarbejdet eller være en pædagogisk guide for elevernes læreproces?
 - › Hvis du indgår som ekspert kræver det, at du har praktisk erfaring med programmering og tør prøve dig frem sammen med eleverne. Hvis du vil indgå som guidende underviser, skal du acceptere, at du ikke kan hjælpe eleverne med alt og i stedet have en mere pædagogisk rolle ved eksempelvis at hjælpe elevernes problemløsning gennem tutorials på nettet.

Figur 7: Opmærksomhedspunkter til inddragelse af programmering i undervisningen.

6.2 INDDRAGELSE AF PROGRAMMERING I GRUNDSKOLEN

Med undersøgelsens udgangspunkt i det pragmatiske paradigme har følgende diskussion det formål at bidrage med erfaringer *“der kan hjælpe os til at handle på en sådan måde, at de ønskede resultater opnås”* (Kvale, 1997, s. 243). For at kunne forbedre arbejdet med at integrere it og programmering i grundskolen fremadrettet diskuteres herunder potentialerne ved at inddrage programmering i eksisterende fag og som selvstændigt fag. Dette gøres på baggrund af analysen af erfaringerne fra Coding Class samt ovenstående diskussion af udfordringerne ved at inddrage programmering som fagligt element i undervisningen samt gennem teoretiske perspektiver indenfor området.

6.2.1 PROGRAMMERING I EKSISTERENDE FAG ELLER SOM SELVSTÆNDIGT FAG

Til IT-Branchens høring d. 1. februar 2017 argumenterede flere af debattens deltagere for, at it og kodning er en del af 21. århundredes kompetencer og at det derfor skal indgå i folkeskolen som selvstændigt fag på linje med dansk og matematik (Bilag 2). Andre talte for, at it skal indgå som integreret del af folkeskolens eksisterende fag, hvor Bjørn Hansen, formand for Danmarks Lærerforening, påpegede, at der ikke er plads til flere selvstændige fag i folkeskolen: *“Folkeskolen bliver altid bedt om at rumme mere og undervise i flere ting, men der er altså ikke plads til flere fag. Derfor handler det i højere grad om at bruge og integrere it i de eksisterende fag”* (IT-Branchen, 2016). Forskeren Michael Caspersen argumenterede modsat for, at folkeskolen må tilpasse sig samfundets udvikling: *“For over 100 år siden var det dem, der kunne læse og skrive, der satte dagsordenen. Sådan bliver det også med it om lidt, og derfor er det vigtigt, at få gjort it til en del af den almene dannelse i folkeskolen”* (IT-Branchen, 2016). Michael Caspersen påpegede desuden, at it både skal indgå som eget fag og i tæt sammenhæng med andre fag i folkeskolen, hvorfor det for ham ikke er et spørgsmål om *enten eller* men *både og*.

Alexander Repenning, der forsker i computer science, er fortaler for at programmering inddrages i skolen og ser kritisk på programmering som fritidsaktivitet, da det ikke rammer en bred målgruppe og det især er drenge der er repræsenteret heri (Repenning, 2012, s. 39). Hans forskning viser størst effekt af programmeringsundervisning i indenfor skolens domæne: *“it is clear that a curriculum-integrated approach has a much higher potential for systemic impact compared to an after-school program”* (Repenning, 2012, s. 39), hvorfor han argumenterer for, at programmering og udvikling af computational thinking skal ske i skolen, så feltet kan udbredes til samt engagere og motivere en bred målgruppe. Repenning påpeger dog, at der indenfor uddannelsessystemer skal være målbare

læringsresultater og at samspillet mellem elev og lærer er et vigtigt element for læreprocessen (Repenning, 2012, s. 38). Repenning understreger derfor, at der skal være en balance mellem computational thinking i relation til pensum samt lærernes kompetencer. I casestudiet oplevede lærerne på tværs af de tre cases en udfordring i netop at balancere inddragelsen af programmering i forhold til de faglige mål, da de ikke er uddannet inden for feltet og ikke har tilstrækkeligt kendskab til programmering til at indarbejde programmering som fagligt element i deres undervisning. Lærerne skal leve op til undervisningsministeriets krav for de enkelte fag, hvorfor inddragelsen af programmering kræver et overblik over programmerings muligheder for at tilpasse undervisningen hermed til fagenes specifikke temaer og mål. Lærernes oplevede det således svært at indarbejde programmering i deres egne fag, hvis de ikke får tilbudt videreuddannelse, hvor de kan få større indsigt i hvordan programmeringen kan tilpasses til de formelle krav, som deres undervisning skal leve op til. I Case 3 ses eksempelvis, at læreren oplever det mere tilgængeligt at indarbejde programmering i sin undervisning, fordi hun har været på kursus og her har fået nogle konkrete forslag til at indarbejde programmering i relation til fagets specifikke temaer og mål (Bilag 13, s. 159). Det er her vigtigt for læreren, at hun kan tilgå de konkrete forslag fra kurset, når hun selv skal planlægge sin undervisning. Læreren fra Case 1 udtrykker ligeledes et behov for at få koblet temaer fra danskfaget på fremadrettet, men kan ikke gøre dette, fordi hun ikke har kendskab og kundskaberne til det, hvorfor hun ønsker videreuddannelse til at opnå dette (Bilag 11, s. 129). Det er således afgørende for programmerings faglige relevans for undervisningen at programmeringen kobles på fagenes temaer og mål. Der ses således her et potentiale i at inddrage programmering som del af eksisterende fag, hvis programmering kobles direkte på de fagspecifikke temaer og mål, hvilket dog kræver videreuddannelse af lærerne. Læreren i Case 1 foreslår dog ligeledes, at der uddannes lærere til at bruge it og programmering som del af den understøttende undervisning, da det *“er sådan set bedre at man har nogle lærere, der brænder for det, og kan undervise i det, eller kan komme ud, hvis der er en lærer der gerne vil bruge det i sit fag”* (Bilag 11, l. 556). Hun ser ydermere en fordel i, at lærerne kan støtte sig til lærere, der ved noget om programmering og finder det interessant, så man som lærer kan lære fra sidelinjen (Bilag 11, s. 18). Læreren i Case 1 udtrykker her en positiv indstilling for, at få it og programmering indgår i grundskolen som et fag for sig selv, så lærere således ikke står i et spændingsfelt mellem elevernes interesser og indfrielse af de fagspecifikke mål i undervisningen. I foråret 2017 er der blevet dannet en styregruppe for udarbejdelsen af et valgfag i Teknologiforståelse, som en måde, at få digital læring ind i undervisningen (Undervisningsministeriet, 2017(b)). Målet er her, at starte et valgfag fra skoleåret 2017, som et muligt alternativ til at inddrage programmering som en integreret del af eksisterende fag i grundskolen. I Case 3 nævner lærerne muligheden for, at arbejde med it og programmering i et valgfag, ligesom de i forvejen har et mediefag med programmering af LEGO Mindstorm, hvor de dog oplever at det især er de fagligt svage elever, der vælger faget (Bilag 13, s. 169). Dette indikerer samme problematik, som indikeret

ved programmering i fritidsaktiviteter af Repenning ovenfor, hvor det er en specifik målgruppe, som har interesse for feltet, hvorfor programmeringen ikke når bredt ud. Hvorvidt programmering kan indgå i grundskolen som valgfag kommer således i forlængelse af diskussionen om, hvorvidt programmering skal indgå i eksisterende fag eller som selvstændige fag, hvor det blot er et spørgsmål om, hvorvidt det gøres obligatorisk eller ej. Der opstår her et konkret valg om, hvorvidt man skal fokusere på de elever, som allerede har interesse for feltet eller man ønsker at brede kendskabet og kundskaberne ud til en større målgruppe. Ved at implementere programmering som obligatorisk fag i stedet for et valgfag kan flere elever få erfaringer med programmering og opnå kompetencer, som ifølge Jerald (2009) efterspørges på arbejdsmarkedet, hvilket indebærer kompetencer indenfor innovation, kreativitet, kritisk tænkning samt problemløsning (Jerald, 2009, s. 70). Jerald beskriver, hvorledes teknologiens eksponentielle udvikling har stor indflydelse på vores arbejdsliv og sociale liv, hvorfor borgere, som besidder disse kompetencer vil være bedre rustet til fremtiden: *“People who prefer conventional work environments are likely to see their jobs disappear. But those who are comfortable working in artistic, investigative, highly social, or entrepreneurial environments are likely to succeed”* (Jerald, 2009, s. 70). Der opstår således et valg om, hvorvidt programmeringen, eller et andet fag der har til formål at arbejde lignende kompetencer, er tiltænkt en bred målgruppe eller blot skal uddanne et udvalg af grundskolens elever med disse kompetencer. Det er dog ligeledes nærliggende at overveje effekten på elevernes motivation i forhold til om programmering indgår som et obligatorisk eller frivilligt element i grundskolens undervisning. Casestudiet viser, at lærerne oplever stor motivation hos eleverne ved programmeringsopgaverne, men her indgår Coding Class og programmering som et brud på den normale undervisning. Det kan derfor være, at elevernes motivation er særdeles høj, fordi undervisningen med programmering udgør en afveksling i forhold til deres normale undervisning i de specifikke fag. Der kan således være en risiko for, at motivationen naturligt falder, hvis programmering gøres til et obligatorisk element i grundskolens undervisning, hvor der modsat herved opstår mulighed for at give flere elever en interesse inden for feltet og opnå kompetencer, som ifølge Jerald ruste dem til deres fremtid på arbejdsmarkedet. Det er således relevant at klarlægge det konkrete mål med it og programmering som fagligt element i grundskolens undervisning før der tages en beslutning om, hvorvidt it og programmering skal indgå i eksisterende fag eller som selvstændigt fag.

6.2.2 VIDEREUDDANNELSE AF LÆRERNE

Seymour Papert har siden 1970'erne arbejdet for, at indarbejde it og programmering i grundskolen, hvor han først havde antaget, at lærerne ville være den største udfordring for at ændre skolen og integrere programmering i undervisningen (Papert, 1993, s. 57). Han erfarede dog at lærerne faktisk ønskede at lære og opnå større viden og kompetencer for at inddrage programmering i deres undervisning, men at der i praksis ikke er mulighed for dette, da: *“many aspects of School block teachers from the fulfillment of functioning in a class as co-learners”* (Papert, 1993, s. 67). Lærerne får ifølge Papert ikke muligheden for at videreudvikle sig løbende og indgå i en fælles læreproces med eleverne grundet forventningerne om at lærere har en ekspertrolle i undervisningen. I casestudiet ses lærernes egen forventning om at de skulle indgå som eksperter ligeledes som en blokering for, at de indgår i en fælles læreproces med eleverne, da de ikke føler sig kompetente til dette. Lærerne i Case 1 og 3 støttede sig eksempelvis til konsulenternes tilstedeværelse, hvorfor de ikke selv blev nødt til at prøve sig frem og få berøring med elevernes kode, da de havde nogle mere kyndige til at gøre det for dem under Coding Class-forløbet. Lærerne i Case 3 lod ligeledes konsulenterne udarbejde forløbet med programmering, da de ikke følte sig kompetente til at udarbejde et undervisningsforløb i matematik med programmering på egen hånd (Bilag 13, s. 157). Papert erfarede dog ligeledes at lærere er tilbageholdende under videreuddannende kurser, hvor *“their awareness of being teachers was preventing them from giving themselves over fully to experiencing what they were doing as intellectually exciting and joyful in its own right, for what it could bring them as a private individuals.”* (Papert, 1993, s. 72). I Casestudiet var det kun i Case 2, hvor læreren var passioneret omkring den tekniske programmering, at læreren indgik i en fælles læreproces med eleverne og fik berøring med koden. I Case 1 og 3 holder lærerne sig på afstand fra elevernes kodning og indgår således ikke i en fælles læreproces med eleverne, hvor de løbende kan udvikle deres evner, som det ses i Case 2. Her bliver lærernes manglende selvtillid og tilbageholdenhed for at få berøring med koden således en barriere for at lærernes kompetencer udvikles. Lærerne i casestudiet ønskede dog kompetenceudvikling, så de netop kunne støtte elevernes arbejde og bedre kunne håndtere inddragelsen af programmering som fagligt element i deres undervisning. Lærerne fra Case 1 og 3 var især påvirket af deres manglende viden om programmering og overblik over mulighederne hermed for forløbets opstart. Ønsket om videreuddannelse handler således både om at udvide lærernes kompetencer og kundskaber, men ligeledes medvirke til større selvtillid, hvorigennem de kan understøttes i at indgå i en fælles læreproces med eleverne. Selvom lærerne på tværs af de tre cases føler sig udfordret i forhold til deres egne kompetencer, er alle lærerne opsatte på at få kompetenceudvikling og relevant videreuddannelse for i fremtiden at kunne implementere programmering i deres undervisning. Her er udgangspunktet for lærernes ønsker for videreuddannelse, at de ligesom i deres normale fag, kan indgå i klasseværelset med en ekspertrolle, hvor de altid kan hjælpe eleverne. De ønsker derfor at få indsigt i konkrete muligheder for at inddrage

fagmålene i deres undervisning, samt at de får overblik over de konkrete muligheder, eksempelvis programmeringssprog og -programmer. Hvis programmering skal indgå i eksisterende fag er det således vigtigt at uddannelse af lærerne, som videreuddannende kursus eller som del af læreruddannelsen, er anvendelsesorienteret, hvor de faglige mål kan opfyldes og lærerne får praktisk erfaring med programmering. Det er dog ligeledes vigtigt at lærerne opfordres til at prøve sig frem og derved indgå i en fælles læreproces med eleverne.

6.3 DELKONKLUSION: PROGRAMMERINGS POTENTIALE

Undersøgelsens diskussion har bidraget med en synliggørelse af udfordringer samt potentialer ved inddragelse af programmering som fagligt element i undervisningen. Første del af diskussionen har bidraget med optegnelsen af flere spændingsfelter, som kan indtænkes ved inddragelsen af programmering i undervisning for at optimere den faglige relevans. Disse spændingsfelter omhandler blandt andet en vægtning mellem elevernes frihed til at opsøge det, de selv finder interessant og lærerens behov for at styre elevernes fagspecifikke udbytte. Der skal ligeledes overvejes, om programmeringen skal indgå som tværfagligt forløb, hvor generelle kompetencer understøttes, eller kobles tæt sammen med de fagspecifikke krav og få en tæt relation til fagenes fastlagte temaer og mål. Det er hertil relevant for elevernes udbytte, at lærerne overvejer gruppesammensætningen herunder antallet af elever, kønsopdeling og graden af indledende individuelt arbejde før arbejdet i grupper. Lærerne skal ligeledes være bevidste om deres egen rolle, der befinder sig i spændingsfeltet mellem ekspert, som kan hjælpe eleverne med den tekniske kodning, og en pædagogisk rolle, som kan guide eleverne læreproces. Anden del af diskussionen tog udgangspunkt i debatten om, hvorvidt programmering skal indgå i eksisterende fag eller som selvstændigt fag i grundskolen. Ved at indarbejde programmering som et selvstændigt fag ses et potentiale i, at lærerne ikke skal vælge mellem elevernes interesser og indfrielsen af de fagspecifikke mål i undervisningen. Det blev herfra ligeledes diskuteret om programmeringen skulle indgå som obligatorisk eller frivilligt fag i grundskolen, hvilket kræver et valg om, hvorvidt programmeringen skal udbredes til en bred målgruppe eller understøtte et begrænset antal elevers interesse indenfor feltet. Ved at indarbejde programmering som fagligt element i eksisterende fag, kræves det at programmering inddrages i tæt relation til de fagspecifikke temaer og mål, samt at lærerne både er og føler sig kompetente til at undervise i programmering. Diskussionen appellerer således til, at programmeringens faglige relevans opnås gennem uddannelse af lærere med fokus på at de får viden og kompetencer til at inddrage programmering i relation til de faglige mål og temaer.

6.4 VIDERE ARBEJDE

For at styrke undersøgelsens reliabilitet, som vedrører konsistens og troværdighed gennem mulighed for at genskabe forskningsresultaterne (Kvale & Brinkmann, 2009, s. 271), kunne der fremadrettet inddrages flere programmeringsforløb i undersøgelsen af programmerings faglige relevans for grundskolens undervisning. Undersøgelsen tager dog udgangspunkt i konkrete undervisningssituationer under forløbet Coding Class, hvorfor feltarbejdet er præget af den specifikke situations kontekst og derfor ikke er mulig at reproducere direkte. En videre undersøgelse af eksempelvis fremtidige Coding Class-forløb vil dog formentligt vise samme dominerende tendenser for læringssituationen, da undersøgelsens resultater viser nogle af de samme tendenser som andre undersøgelser inden for feltet. Som nævnt i Kapitel 2, viser andre undersøgelser eksempelvis at programmering bidrager til elevernes refleksive kompetencer og problemløsningsevner, hvor denne undersøgelse viser stor tilstedeværelse af problemløsning og samarbejde i læringssituationen. Denne undersøgelse bidrager dog ikke med indsigt i elevernes udvikling heraf, hvorfor det er relevant at følge elevers individuelle udvikling og udbytte af inddragelse af programmering i undervisningen fremadrettet. Undersøgelsen viser ligeledes variation i elevernes faglige udbytte afhængig af deres egne interesser og rollefordelinger i gruppen, hvilket ligeledes ses i Lopez et al. (2016), hvor læringsudbyttet især blev påvirket af elevernes engagement. I denne undersøgelse varierede lærernes oplevelse af elevernes læringsudbytte dog ligeledes på grund af forløbenes opbygning samt de involverede fag, hvor undersøgelsens reliabilitet kunne styrkes ved at inddrage flere eksempler på inddragelse af programmering på tværs af de forskellige fag. Det kunne således være relevant for videre arbejde, at undersøge programmeringens relevans afhængig af hvilke fag, det inddrages i og hvorvidt det bidrager ligeligt til eksempelvis humanistiske samt naturvidenskabelige fag. Undersøgelsen af Nielsen et al. (2015) udleder, som nævnt i Kapitel 2, at lærere har besvær med at inddrage programmering i overensstemmelse med de fagspecifikke mål, hvilket undersøgelsen her ligeledes peger på. Der ses således et behov for videre undersøgelse af, hvorledes programmerings relevans kan sikres i relation til de fagspecifikke mål, når programming inddrages i eksisterende fag. På baggrund af undersøgelsens resultater kan dette blandt andet forbedres gennem videreuddannelse af lærerne, hvor de undervises i at inddrage programmering i overensstemmelse med fagenes fagspecifikke mål og temaer. Der kan på denne baggrund ligeledes fremadrettet undersøges, hvordan videreuddannelse af lærere afvikles mest hensigtsmæssigt for at sikre, at de kan imødekomme undervisningsministeriets krav til deres undervisning.

I forlængelse af undersøgelsens resultater kunne det yderligere være interessant at udvikle konkrete undervisningsplaner for at inddrage programmering i de enkelte fag med fokus på indfrielse af fagspecifikke mål. Ud fra erfaringerne med Case 1 og 2 kan endvidere undersøges, hvorledes programmering kan opbygges som tværfagligt forløb, hvor der kan udarbejdes en model for at

indarbejde en virksomhedsopbygning, der inkluderer udvikling af spil, at holde regnskab og udformning af reklamemateriale i et programmeringsforløb. I casestudiet viste virksomhedsopbygningen sig både at være relevant for dansk- og matematikfaglige mål, og samtidig kan elementer fra fag som billedkunst og engelsk muligvis indtænkes. På baggrund af undersøgelsens resultater er det ligeledes relevant at følge det videre arbejde med Coding Class samt det nye valgfag i Teknologiforståelse. I relation til implementeringen af det nye valgfag er det nærliggende at undersøge elevernes faglige og generelle udbytte heraf samt undersøge udviklingen af elevernes digitale kompetencer og dannelse, som den aktuelle undersøgelsen har afholdt sig fra.

KAPITEL 7

KONKLUSION

...

Afslutningsvis præsenteres konklusionen for undersøgelsens besvarelse af undersøgelsesspørgsmålet gennem resultaterne fra undersøgelsens analyse og diskussion.

7.1 UNDERSØGELSENS RESULTATER

Undersøgelsen har haft til formål at belyse, hvordan programmering kan indgå som fagligt element i undervisningen i grundskolen med afsæt i erfaringer fra Coding Class. Med udgangspunkt i et casestudie af tre skolars involvering i pilotprojektet Coding Class har undersøgelsen givet indsigt i, hvordan inddragelsen af programmering under Coding Class-forløbene har haft indflydelse på læringssituationen i praksis i de tre cases, samt hvorledes lærerne oplever programmerings faglige relevans i deres undervisning. Herfra har undersøgelsen haft til hensigt at opstille konkrete opmærksomhedspunkter til læreres fremadrettede arbejde med at inddrage programmering i undervisningen, samt diskutere hvordan programmering kan bidrage som fagligt element i grundskolens eksisterende fag i forhold til et selvstændigt fag.

Undersøgelsens analyse af læringssituationen under Coding Class-forløbene samt lærernes erfaringer hermed viste, at elevernes engagement og fordybelse i projekterne er højt på tværs af de tre cases, hvor elevernes programmeringsarbejde er interessebåret samt fokuseret på problemløsning og samarbejde. Lærerne var af den opfattelse, at elevernes faglige udbytte ved inddragelse af programmering varierer og blandt andet afhænger af gruppesammensætning, rollefordeling i grupperne samt forløbets opbygning. Indsigten i lærernes erfaringer med Coding Class belyste yderligere, at lærerne er tilbageholdende med at få praktisk berøring med programmeringsarbejdet grundet deres begrænsede viden og kompetencer inden for feltet. Lærerne oplever forskellige grader af faglig relevans ved inddragelse af programmering i deres fag, hvorfor de ønsker videreuddannelse, hvor programmering og fagenes konkrete temaer og mål kobles sammen.

Undersøgelsens diskussion bidrager med indsigt i spændingsfelter, som lærere står overfor, hvis de ønsker at inddrage programmering i deres undervisning. Spændingsfelterne omhandler blandt andet en vægtning af lærernes faglige målstyring eller at give eleverne frihed til, at deres læring styres af deres interesser samt en vægtning af om programmering skal indgå som tværfagligt forløb, hvor generelle kompetencer understøttes, eller kobles tæt sammen med fastlagte temaer og mål, hvor fagenes fagspecifikke krav indfries. Det er relevant for elevernes udbytte, at lærere overvejer gruppesammensætning samt balancerer mellem indledende individuelt arbejde og gruppearbejde. Lærere skal ligeledes vægte hvorvidt de ønsker at have en ekspertrolle, som kan hjælpe eleverne med den tekniske kodning, eller en pædagogisk rolle, som har fokus på at guide elevernes læreproces. Diskussionen belyser ydermere potentialer og udfordringer ved at inddrage programmering som fagligt element i eksisterende fag eller som selvstændigt fag i grundskolen, hvor der ved at indarbejde programmering som selvstændigt fag ses et potentiale i, at lærerne ikke skal balancere deres undervisning mellem elevernes interessebårne læring og de fagspecifikke mål. Der skal dog yderligere tages stilling til, om et selvstændigt fag skal være obligatorisk eller frivilligt, og derfor om programmering skal udbredes til en bred målgruppe eller understøtte et begrænset antal elever.

interesse indenfor feltet. Ved at indarbejde programmering som fagligt element i eksisterende fag, kræves det modsat at programmering inddrages i tæt relation til fagspecifikke temaer og mål, samt at lærerne både er og føler sig kompetente til at undervise i programmering. Diskussionen appellerer således til, lærere skal uddannes til at inddrage programmering meningsfuldt i deres undervisning, hvis det ønskes indarbejdet i eksisterende fag.

På baggrund af undersøgelsens resultater er det nødvendigt at opnå yderligere praksiserfaringer med programmering i undervisningen samt undersøge flere aspekter af programmerings faglige relevans i grundskolen. Undersøgelsens resultater bidrager således ikke med generaliseringer, om hvordan programmering bør indgå meningsfuldt i den danske grundskole men derimod med konkret indsigt i, hvorledes programmering kan opnå faglig relevans i grundskolen. Undersøgelsen har ikke haft til hensigt at handle ud fra resultaterne men udgør derimod et videnskabeligt perspektiv i den offentlige og politiske debat, hvorfra passende beslutninger og handlinger kan træffes og udføres fremadrettet på området.

LITTERATUR

LITTERATURLISTE

- Ackermann, E. (2001). Piaget's constructivism, papert's constructionism: What's the difference. *Future of Learning Group Publication*, 5(3), 438.
- Blomberg, J., Giacomi, J., Mosher, J., & Swenton-Wall, P. (1993). Ethnographic field methods and their relation to design. In D. Schuler, & A. Namioka (Eds.), *Participatory design: Perspectives on systems design* (s. 123-154). Hillsdale: Lawrence Erlbaum Associates, Inc.
- Brinkmann, S. (2007). Motivation gennem handling og gøremål - et pragmatisk perspektiv. *KvaN - et tidsskrift for læreruddannelsen og folkeskolen*, 26(78), 91-101.
- Brinkmann, S., & Tanggaard, L. (2010). Towards an epistemology of the hand. *Studies in Philosophy and Education*, 29(3), 243-257.
- Creswell, J. W. (2003). A framework for design. *Research design: Qualitative, quantitative, and mixed methods approaches* (s. 3-26). London: Sage publications.
- Creswell, J. W. (2012). Ethnographic designs. *Educational research: Planning, conducting, and evaluating quantitative and qualitative research* (s. 461-500). Upper Saddle River, NJ: Pearson.
- Danmarks Evalueringsinstitut. (2009). *It i skolen: Undersøgelse af erfaringer og perspektiver*.
- Danmarks Vækstråd. (2016). *Rapport om kvalificeret arbejdskraft*.
- Denner, J., Werner, L., & Ortiz, E. (2012). Computer games created by middle school girls - can they be used to measure understanding of computer science concepts. *Computers & Education*, 58, 240-249.
- Dewey, J. (1980)[1912]. Uddannelse som vækst. *Demokrati og uddannelse* (s. 61-72). Aarhus: Klim.
- Ejsing-Duun, S., & Misfeldt, M. (2015). Programmering af robotenheder i grundskolen. *Læring Og Medier (LOM)*, 8(24)
- EMU. (u.å.(a)). It og medier - vejledning. Lokaliseret 10. maj 2017 på: www.emu.dk/modul/it-og-medier-vejledning
- EMU. (u.å.(b)). Vejledning for faget fysik/kemi. Lokaliseret 10. maj 2017 på: www.emu.dk/modul/vejledning-faget-fysikkemi
- EMU. (u.å.(c)). Vejledning for faget matematik: 3.4 it og medier. Lokaliseret 10. maj 2017 på: www.emu.dk/modul/vejledning-faget-matematik#
- Erickson, F. (1986). Qualitative methods in research on teaching. In M. Wittrock (Ed.), *Handbook of research on teaching* (s. 119-169). New York: MacMillan.
- Erkman, M., & Petropouleas, E. (2017). *Programmering i praksis: Didaktiske design og læringspotentialer i grundskolen*. Frederikshavn: Dafolo.
- Flyvbjerg, B. (2010). Fem misforståelser om casestudiet. In L. P. Tanggaard, & S. Brinkmann (Eds.), *Kvalitative metoder: En grundbog*. (s. 463-488). København: Hans Reitzels Forlag.

- Foerster, K. Integrating programming into the mathematics curriculum: Combining scratch and geometry in grades 6 and 7. *Sigite*, 91-96.
- Issroff, K., & Jones, A. (2007). Learning technologies - affective and social issues. In G. Conole, & M. Oliver (Eds.), *Contemporary perspectives in E-learning research. themes, methods and impact on practice* (s. 190-202). London: Routledge.
- IT-Branchen. (u.å.(a)). Coding class. Lokaliseret 27. marts 2017 på: www.itb.dk/articles/fremtidens-kompetencer-coding-class
- IT-Branchen. (u.å.(b)). Sikring af fremtidens kompetencer. Lokaliseret 27. marts 2017 på: www.itb.dk/fremtidens-kompetencer
- IT-Branchen. (2016). Uenighed om it på skoleskemaet. Lokaliseret 27. marts 2017 på: www.itb.dk/news/fremtidens-kompetencer/uenighed-om-it-pa-skoleskemaet
- IT-Branchen. (2017(a)). Hvordan skal kreativ it på skoleskemaet? Lokaliseret 5. april 2017 på: www.itb.dk/news/fremtidens-kompetencer/hvordan-skal-kreativ-it-pa-skoleskemaet
- IT-Branchen. (2017(b)). Støt op om Coding Class. Lokaliseret 5. april 2017 på: www.itb.dk/news/fremtidens-kompetencer/stot-op-om-coding-class
- James, W. (1907). What pragmatism means. *Pragmatism* (s. 17-32). NY: Dover Publications Inc.
- Jerald, C. D. (2009). *Defining a 21st century education*. (). The Center for Public Education.
- Kafai, Y. B., & Resnick, M. (1996). Introduction. *Constructionism in practice: Designing, thinking and learning in a digital world* (s. 1-8). New York: Routledge.
- Kalelioğlu, F. (2015). A new way of teaching programming skills to K-12 students: Code.ord. *Computers in Human Behavior*, 52, 200-210.
- Kalelioğlu, F., & Gülbahar, Y. (2014). The effect of teaching programming via scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education*, 13(1), 33-50.
- Kvale, S. (1997). *Interview: En introduktion til det kvalitative Forskningsinterview*. København: Hans Reitzel Forlag.
- Kvale, S. & Brinkmann, S. (2009). Den sociale konstruktion af validitet. I: Kvale, S. & Brinkmann, S., *InterView - Introduktion til et håndværk* (2, udg., s. 267-292). København: Hans Reitzels Forlag.
- Levensen, K. T. (2011). Udfordringer i netværkssamfundet: Digital kompetencer og it i nye undervisningsmiljøer. *Dansk Universitetspædagogiske Tidsskrift*, 11, 20-25.
- López, J., González, M., & Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using "scratch" in five schools. *Computers & Education*, 97, 129-141.
- Lu, J., & Fletcher, G. (2009). Thinking about computational thinking. *Proceedings of the 40th ACM Technical Symposium on Computer Science Education*, , 260-264.

- Maaløe, E. (2002). *Casestudier af og om mennesker i organisationer* (2. udgave ed.). Danmark, Viborg: Akademisk Forlag.
- Mackenzie, N., & Knipe, S. (2006). Research dilemmas: Paradigms, methods and methodology. *Issues in Educational Research*, 16(2), 193-205.
- Martin, A. & Grudziecki, J. (2006). DigEuLit: Concepts and tools for digital literacy development. *ITALICS*, 5(4), 249-267.
- Martinez, S. L., & Stager, G. (2013). *Invent to learn: Making, thinking and engineering in the classroom*. Torrance, Calif: Constructing Modern Knowledge Press.
- Mathiasen, H., & de Wit, C. K. (2010). IT i uddannelsessystemet - med hvilket sigte? *Dansk Universitetspædagogisk Tidsskrift*, 9, 12-17.
- Misfeldt, M. (2015). Mellem læringspotentiale og skuffelse – et bud på en it-didaktik for matematik. In M. W. Andersen, & P. Weng (Eds.), *Håndbog om matematik i grundskolen: Læring, undervisning og vejledning* (s. 425-473) Dansk Psykologisk Forlag.
- Nielsen, J., Pedersen, R., & Majgaard, G. (2015). 8. klasse som kreative producenter af fremtidens velfærdsteknologi - konstruktionisme, problemløsning og dialog. *Læring & Medier (LOM)*, 14
- Papert, S. (1980). Foreword: The gears of my childhood. *Mindstorms; children, computers and powerful ideas* (). New York, USA: Basic Books.
- Papert, S. (1991). Situating constructionism. In I. Harel, & S. Papert (Eds.), *Constructionism* (s. 1-12). Norwood, NJ: Ablex Publishing Corporation.
- Papert, S. (1993). *The children's machine: Rethinking school in the age of the computer* (2. edition ed.). United States: BasicBooks.
- Raudaskoski, P. (2010). Observationsmetoder (herunder videoobservation). In L. P. Tanggaard, & S. Brinkmann (Eds.), *Kvalitative metoder: En grundbog* (s. 81-96). København: Hans Reitzels Forlag.
- Repenning, A. (2012). Programming goes back to school. *Communications of the ACM*, 55(5), 38-40.
- Resnick, M. (2012). Reviving papert's dream. *Educational Technology*, 52(4), 42-46.
- Rode, J., Weibert, A., Marchall, A., Aal, K., Rekowski, T., Mimoni, H., & Booker, J. From computational thinking to computational making. *UbiComp*, 239-250.
- Smith, G., & Grant, B. (2000). From players to programmers: A computer game design class for middle-school children. *Journal of Educational Technology Systems*, 28(3), 263-275.
- Stahl, G., Koschmann, T., & Suthers, D. (2006). Computer-supported collaborative learning: An historical perspective. In K. R. Sawyer (Ed.), *Cambridge handbook of the learning sciences* (s. 409–426). Cambridge: Cambridge University Press.
- Stake, R. E. (2005). Qualitative case studies. In K. B. Denzin, & Y. S. Lincoln (Eds.), *The SAGE handbook of qualitative research* (s. 443-466) Sage Publications.
- Stake, R. E. (1995). *The art of case study research*. United States: Sage Publications.

- Tracy, S. J. (2010). Qualitative Quality: Eight “Big-Tent” Criteria for Excellent Qualitative Research. *Qualitative Inquiry* 16(10), 837-851.
- Undervisningsministeriet. (2017(a)). Om fælles mål. Lokaliseret 9. maj 2017 på: www.uvm.dk/folkeskolen/fag-timetel-og-overgange/faelles-maal/om-faelles-maal
- Undervisningsministeriet. (2017(b)). Undervisningsministeren nedsætter rådgivningsgruppe for digital læring. Lokaliseret 9. maj 2017 på: www.uvm.dk/aktuelt/uvm/udd/folke/2017/mar/170330-undervisningsministeren-nedsaetter-raadgivningsgruppe-for-digital-laering
- Wing, J. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Woo, D. (2016). Structural barriers and organizational mechanisms for training and deploying ICT champions in a school. *Education Technology Research and Development*, 64, 839-855.
- Zaharija, G., Mladenovic, S., & Boljat, I. (2013). Introducing basic programming concepts to elementary school children. *Procedia - Social and Behavioral Sciences*, 106, 1576-1584.
- Zainal, Z. (2007). Case study as a research method. *Jurnal Kemanusiaan*, 9.