

MASTEROPPGAVE

Emnekode: MAT5006

Navn: Morten Strand

Fagfornyelsens innføring av programmering i grunnskolen

- En kvantitativ studie om matematikk-læreres profesjonelle utvikling

Dato: 17.05.2022

Totalt antall sider: 83

Abstract

In recent years there has been a global push to implement programming in the curriculum. A common challenge for most countries is the teachers' lack of the necessary competence. This is no different in Norway, where many of the teachers do not have the necessary competence and have been requesting support for their professional development.

The goal of this study is to explore the following research question:

- How has mathematics teachers' professional development in relation to programming been facilitated after the implementation of a new curriculum, and what effect has it had?

This study is a quantitative study which was conducted by using an internet survey, which gathered answers from 51 mathematics teachers spread across Nordland.

The answers from the correspondents imply there has not been a push for formal programming education for teachers. 8% of the respondents have formal education related to programming, and only 22% have been able to attend university courses after the implementation. There are indications of several issues when it comes to the development of teacher competence that has taken place after the implementation. Almost 75% of the teachers feel there has been little to no developmental work related to the new curriculum conducted in their schools. Out of the four types of developmental work asked about, teachers find the university courses most useful. Teachers call for the school administrations to dedicate more time and resources to development work focused on programming. Despite this the teachers show positive development regarding programming competency, self-confidence, and motivation for teaching programming.

It is problematic that over a year after the implementation many of the teachers experience not having adequate access to teaching resources, not having adequate competency for teaching programming, and that the development work that has been done so far has not been useful. Because of this schools should make a greater effort to include teachers in planning of the development work. There was found significant differences between the teacher's assessment of their own competence before the implementation compared to now in schools with more than 200 students. There was not found a significant difference in the smaller schools, which might mean that they require additional support onward.

Sammendrag

Globalt har det vært en bølge av innføring av programmering i læreplanen. En felles utfordring for mange av landene er den manglende kompetansen hos lærerne. Dette har også vært en utfordring i Norge, der store deler av lærerne ikke har kompetanse, og har etterlyst kompetanseutvikling.

Målet for studien er å belyse følgende problemstilling:

- Hvordan har matematikklæreres utvikling i relasjon til programmering foregått etter fagfornyelsen, og hvilken effekt har det hatt?

Studien er en kvantitativ studie der det ble benyttet spørreundersøkelse som forskningsmetode, med svar fra 51 matematikklærere spredt over hele Nordland.

Svarene fra respondentene tyder på at det ikke har vært en satsing på formell utdanning innenfor programmering for lærerne. Kun 8% av de spurte har formell kompetanse fra tidligere utdanning, og i etterkant av innføringen har kun 22% av lærerne fått delta på universitetskurs. Tallene peker på at det har vært flere mangler når det gjelder utviklingen etter innføringen av læreplanen. Nesten 75% av lærerne opplever at det har vært gjennomført lite eller nesten ingenting av utviklingsarbeid knyttet til det nye læreplanverket. Av de undersøkte typene oppleggene opplever lærerne at formelle universitetskurs er den nyttigste formen for utviklingsarbeid. Lærerne etterlyser at ledelsen dedikerer mer tid og ressurser til utviklingsarbeid knyttet til programmering. Til tross for utfordringene har lærerne vist en utvikling innenfor både programmeringskompetanse, selvtillit, og motivasjon for programmeringsundervisning.

Det er problematisk at over et år etter innføringen opplever store deler av lærerne å ikke ha god tilgang på undervisningsressurser, å ikke ha et tilfredsstillende nivå av kompetanse for undervisning, og opplever at store deler av utviklingsarbeidet som er gjort ikke har vært nyttig. Med så dårlige vurderinger av arbeidet så langt bør det gjøres en større innsats for å inkludere lærerne i planleggingen arbeidet. For lærerne ved større skoler viser deres egne vurderinger av kompetanse, selvtillit, og motivasjon før innføringen og nå, en signifikant forskjell over utviklingsperioden. Denne forskjellen kan ikke ses i resultatene for lærere ved mindre skoler, noe som taler for at mindre skoler trenger ekstra støtte fra øvre organer.

Forord

Denne studien markerer slutten på et fem år langt studie for en mastergrad i Grunnskolelærerutdanningen 5.-10. trinn ved Nord Universitet. Det har vært fem spennende år der jeg har blitt kjent med mange nye folk, og fått mange nye opplevelser og erfaringer. Veien fra første praksisperiode til i dag kjennes kortere ut enn den virket som da jeg startet førsteåret, men det har vært fem fine og minneverdige år.

Arbeidet med masteroppgaven har vært både utfordrende og spennende. Gjennom arbeidet har det vært mange muligheter for refleksjoner rundt utvikling av skolen, og om viktigheten av å kontinuerlig arbeide for forbedring.

Jeg ønsker å benytte anledningen til å gi en stor takk til min veileder Asif Mushtaq ved fakultetet for lærerutdanning og kunst- og kulturfag ved Nord Universitet, for gode innspill og hjelp i arbeidet med masteroppgaven. Takk også til alle praksisveilederne jeg har hatt gjennom årene, som har stilt med både noen av de største utfordringene i studieløpet, men også de største høydepunktene. Og en stor takk til mine medstudenter, for alle gode diskusjoner og for fem flotte år på studiet. Ikke minst takk til alle som gjorde studien mulig, ved å sette av tid til å delta i spørreundersøkelsen i en ellers hektisk hverdag.

Til sist ønsker jeg å takke mine venner, både nære og fjerne. Takk både for støtten underveis, og for avbrekkene fra studiene, det har betydd utrolig mye.

Morten Strand

Bodø, mai, 2022

Innholdsfortegnelse

Abstract	i
Sammendrag	ii
Forord	iii
Innholdsfortegnelse	iv
Tabeller.....	vii
Figurer	viii
1.0 Introduksjon	1
1.1 Bakgrunn for studien	1
1.2 Fagfornyelsen	2
1.2.1 Digiutvalget	3
1.2.2 Lysneutvalget	3
1.2.3 Sanneutvalget	3
1.2.4 Digitaliseringsstrategi 2017-2021	4
1.3 Problemstilling	4
1.4 Begrunnelse for valg av problemstilling	5
1.5 Struktureringen av oppgaven	6
2.0 Teori	7
2.1 Læreplanens betydning for skolene.....	7
2.2 Programmering	8
2.2.1 Algoritmisk tenkning.....	9
2.2.2 Programmeringsspråk.....	11
2.2.3 Blokkbasert programmering.....	11
2.2.4 Fysisk programmering.....	12
2.2.5 Tekstbasert programmering.....	12
2.3 Dybdelæring	13
2.3.1 Teknologisk pedagogisk fagkompetanse	14
2.4 Skoleutvikling	15
2.4.1 Myndigheter	16
2.4.2 Ledelse	18

2.4.3 Lærere.....	19
2.5 Utvikling.....	19
2.5.1 Skaperverksteder	20
2.5.2 Lærerutdanning for programmering.....	20
3.0 Metode.....	23
3.1 Vitenskapsteoretiske betraktninger	23
3.1.1 Postpositivistisk.....	23
3.2 Forskningsdesign og metode.....	24
3.2.1 Kvalitativ og kvantitativ metode	25
3.3 Gjennomføringen av datainnsamlingen	26
3.3.1 Spørreundersøkelse som forskningsmetode	26
3.3.2 Utvalg og deltakere	29
3.3.3 Analysering av spørreundersøkelse.....	31
3.4 Kvaliteten i forskning.....	32
3.4.1 Reliabilitet	32
3.4.2 Validitet.....	33
3.4.3 Generaliserbarhet	34
3.5 Etikk	34
4.0 Funn og analyse.....	36
4.1 Respondentene	36
4.1.1 Kjønnfordeling.....	36
4.1.2 Aldersfordeling.....	37
4.1.3 Skolestørrelse	38
4.1.4 Fordeling over skoletrinn	38
4.1.5 Respondentenes kompetanse i programmering.....	39
4.2 Programmering i fag.....	40
4.3 Programmeringsspråk/programmeringsverktøy	41
4.4 Undervisningsressurser	43
4.5 Utviklingsarbeid	44
4.5.1 Grad av gjennomført utviklingsarbeid	44

4.5.2 Nyttighet og effekt	48
4.5.3 Programmeringskompetanse	50
4.5.4 Selvtillit for undervisning.....	52
4.5.5 Motivasjon for undervisning	53
5.0 Diskusjon.....	57
5.1 Drøfting av forskningsmetode.....	57
5.1.1 Spørreskjema	57
5.1.2 Kvalitet i studien	58
5.2 Drøfting av resultater	59
5.2.1 Mangel på tid og ressurser	59
5.2.2 Positiv utvikling, for større skoler	63
6.0 Avslutning	66
6.1 Problemstillingen	66
6.2 Studiens betydning og videre forskning.....	67
7.0 Litteraturliste	68

Tabeller

Tabell 1: Antall skoler spørreundersøkelsen ble sendt til	30
Tabell 2: Antall svar fra lærere i kommuner	30
Tabell 3: Fag der respondentene benytter seg av programmering	40
Tabell 4: Fag der respondentene benytter seg av programmering sortert etter skoletrinn	40
Tabell 5: Respondentenes preferanser	41
Tabell 6: Respondentenes bruk i undervisningen.....	42
Tabell 7: Respondentenes bruk i undervisningen sortert etter skoletrinn	42
Tabell 8: Respondentenes tilgang på undervisningsressurser	43
Tabell 9: Respondentenes opplevelse av mengde utviklingsarbeid sortert etter skoletrinn....	45
Tabell 10: Typer utviklingsarbeid respondentene har deltatt i.....	47
Tabell 11: Respondentenes opplevelse av typer utviklingsarbeids nyttinghet.....	49
Tabell 12: Respondentenes vurdering av egen programmeringskompetanse før og etter utviklingsarbeid.....	50
Tabell 13: Respondentenes vurdering av egen programmeringskompetanse før og etter utviklingsarbeid samlet i grupper etter deltakelse i typer utviklingsarbeid	50
Tabell 14: Respondentenes vurdering av egen programmeringskompetanse før og etter utviklingsarbeid samlet i grupper etter skolestørrelse.....	51
Tabell 15: Respondentenes vurdering av egen selvtillit for undervisning før og etter utviklingsarbeid.....	52
Tabell 16: Respondentenes vurdering av egen selvtillit for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter deltakelse i typer utviklingsarbeid	52
Tabell 17: Respondentenes vurdering av egen selvtillit for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter skolestørrelse.....	53
Tabell 18: Respondentenes vurdering av egen motivasjon for undervisning i programmering for og etter utviklingsarbeid	54
Tabell 19: Respondentenes vurdering av egen motivasjon for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter deltakelse i typer utviklingsarbeid.....	54
Tabell 20: Respondentenes vurdering av egen motivasjon for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter skolestørrelse	55

Figurer

Figur 1: “Den algoritmiske tenkeren”	10
Figur 2: “The TPACK Framework and its content knowledge”	15
Figur 3: Oversikt over svar fra kommunene	30
Figur 4: Kjønnfordeling blant respondentene på undersøkelsen	36
Figur 5: Kjønnfordeling blant lærere, Norge 2020.....	36
Figur 6: Aldersfordeling blant respondentene på undersøkelsen	37
Figur 7: Aldersfordeling blant lærere, Norge 2020.....	37
Figur 8: Respondentenes skolestørrelse målt i antall elever ved skolen	38
Figur 9: Skoletrinnene respondentene underviser ved	38
Figur 10: Respondentenes kompetanse i programmering	39
Figur 11: Respondentenes opplevelse av mengden utviklingsarbeid som har blitt gjennomført knyttet til LK20	45
Figur 12: Respondentenes opplevelse av mengde generelt utviklingsarbeid.....	46
Figur 13: Antall opplegg respondentene har tatt del i.....	47

1.0 Introduksjon

Det har de siste årene vært en sterk satsning på programmering i skolen over hele verden. Det kommer som en følge av at samfunnet preges sterkt av digitaliseringen, og da må og har også skolene en lovpålagt oppgave om å utvikles for å best mulig kunne gi elevene en utdanning som er relevant for morgendagens samfunn (Opplæringslova, 1998, § 10-8). En europeisk rapport fra 2013 konkluderte med at “European nations are harming their primary and secondary school students, both educationally and economically, by failing to offer them an education in the fundamentals of informatics” (Informatics Europe et al., 2013, s.17-18). Men med denne innføringen av programmering følger det også egne utfordringer, spesielt knyttet til skolenes beredskap, og læreres kompetanse.

1.1 Bakgrunn for studien

Den norske ideen om innføring av programmering var ikke unik, og mange land globalt har de siste årene innført programmering som en obligatorisk del av grunnskolen. Finland var tidlig ute og hadde klar en ny læreplan i 2014 som trådte i kraft fra 2016. De beregnet en to-års periode fra læreplanen var klar, til den trådte i kraft slik at kommuner og skoler skulle ha muligheten til å forberede seg. I tillegg til at skolene fikk tid til å gjennomføre lokalt arbeid knyttet til en ny læreplan ble det utviklet digitale læremateriell som støtte for både rektorer og lærere. De har også i ettertid fått på plass tre store tiltak som skal bidra til utvikling av lærernes kompetanse, to i regi av deres versjon av utdanningsdirektoratet, og et i privat regi (Bocconi et al., 2018, s.13, s.21).

I mars 2017 hadde Sverige klar en ny læreplan, som ble innført i 2018 der blant annet programmering var et nytt tillegg. I perioden mellom 2017 og 2018 var det mulig for skoler å frivillig velge å følge den nye læreplanen som “pilot-skoler”. En av utfordringene de møtte var å ha 200 000 lærere med et behov for videre profesjonsfaglig utvikling for å undervise i programmering. For å møte denne utfordringen innførte de blant annet kompetansepakker, digitale moduler som var designet for flere emner med et mål om at lærere, eventuelt flere lærere sammen kunne gjennomgå informasjonen og opplegget, og sammen få bedre kompetanse for det emnet. Prosessen med å utvikle lærerutdanningene begynte også rundt samme tid, og har satt i gang grunnleggende programmeringskurs for matematikklærere (Bocconi et al., 2018, s.13-14, s.21-22; Heintz et al., 2017, s.123-125).

Ny læreplan i Norge tredde i kraft fra 2020, blant annet med innførelsen av flere kompetansemål innenfor matematikk som gjaldt programmering. I likhet med Sverige var læreres kompetanse en utfordring, og det ble blant annet utviklet kompetansepakker som støtte til innføringen.

1.2 Fagfornyelsen

Læreplanene danner selve grunnlaget for all undervisningen som skal skje ved skolene. Fra de tidligere læreplanene har de nyere moderne læreplanene flyttet fokuser mer og mer vekk fra læreren og over til elevene. Det vil si at man i mindre grad har veldig konkrete læringsmål som legger grunnlaget for hvordan læreren skal undervise, og heller har mer overordnede kompetansemål som beskriver mer generelt hva eleven skal lære. Denne gradvise endringen skyldes samfunnsendringer, og at læreplanen utformes etter hva samfunnet har behov for. Der kunnskap tidligere ble sett på som en fast sannhet, forstår vi i dag kunnskap som noe individuelt og kontekstuel. Altså ønsker samfunnet i mindre grad å styre hvordan lærerne underviser, men heller sette mål for hva vi ønsker at elevene skal lære. Der er derimot ikke kun hvordan elevene skal lære som endres i læreplanen, samfunnets behov i form av hva elevene bør kunne endres også (Lyngsnes & Rismark, 2014, s.164-165).

Blant de moderne læreplanene har levetiden også blitt kortere og kortere frem til Kunnskapsløftet 2006 (LK06), der det kunne gjøres små endringer underveis da læreplanen i stor grad var digital. Dette skyldes at informasjon og kunnskap har i større og større grad blitt “ferskvare”, og noe som endres eller oppdateres hyppig. Med de store endringene som har skjedd som følge av den digitale revolusjonen har det også fulgt et stort behov for å utvikle læreplanen. Å utvikle en ny læreplan er derimot noe som er omfattende og tar lang tid, noe som fører med seg egne utfordringer. Spesielt når det gjelder de digitale målene i læreplanen kan det argumenteres for at den norske utviklingen av læreplanen har vært for treg (Lyngsnes & Rismark, 2014, s.164-165).

Dagens læreplan, Kunnskapsløftet 2020 (LK20), har røtter blant annet fra 2013 med Stortingsmelding 20, der det blir påpekt at grunnopplæringen skal gi elevene den kompetansen som er nødvendig for fremtidens samfunn. Der blir det foreslått å sette i gang en prosess der et utvalg får mandat til å utrede hvilke kompetanser og ferdigheter de mener er viktige for å møte fremtidens kompetansebehov (Kunnskapsdepartementet, 2013, s.66-67).

1.2.1 Digiutvalget

Planen har også rot i arbeidet fra Digiutvalgets utredning fra 2013. Digiutvalget ble i 2011 bedt om å identifisere og kartlegge barrierer for digital verdiskapning. De ble også bedt om å vurdere tiltak iverksatt i andre land, og komme med innspill til potensielle politiske grep. De mente da at programmering burde innføres i skolen tidligere enn i den videregående utdanningen. Ved at elevene i 2013 hadde få eller ingen muligheter til å lære programmering i grunnskolen mente de at elevene ble lært til å bli konsumenter, og ikke skapere av digitale tjenester. De peker på at det beste ville vært å tilby undervisning i programmering allerede fra barneskolen (Kommunal- og distriktsdepartementet, 2013, s.105).

1.2.2 Lysneutvalget

Lysneutvalget ble oppnevnt i 2014 med et mandat om å utrede digitale sårbarheter i en digitalisert verden, men ser også kompetanse og bygger på Digiutvalgets utredning. Utredningen ble levert i 2015. De er enige i Digiutvalgets vurdering om at nye generasjoner må kunne skape, ikke bare forbruke teknologi. De drar frem at en rekke andre land i EU enten har innført, eller planlegger å innføre opplæring i programmering i skolen. Det ville imidlertid ikke vært realistisk å forvente at lærerne skulle inneha den nødvendige kompetansen til å undervise i programmering. Finland dras frem som et eksempel på en løsning, der de har arbeidet mot tilstrekkelig kompetanse i et samarbeid mellom det offentlige og den private sektoren (Justis- og beredskapsdepartementet, 2015, s.225).

1.2.3 Sanneutvalget

Sanneutvalget ble oppnevnt i 2016 med et mandat om å foreta en gjennomgang av teknologi i grunnopplæringen. De argumenterer for at i skolens oppgave om å gi elevene en teknologisk allmenndannelse ligger det at elevene skal kunne bruke, forstå og å utvikle omgivelsene. Dette krever kunnskap om teknologi, hvordan det er satt sammen, og hvordan det fungerer. De påpeker at teknologi innføres som et fag i flere land, og et flere land integrerer programmering i sine læreplaner. I 18 av 21 undersøkte europeiske land har de på dette tidspunktet enten allerede gjort, eller har planer om å innføre programmering i læreplanen. Norge skiller seg ut som et av tre land som da ikke har slike planer. De mener det er rart at teknologi vurderes som vanskelig å integrere i skolen. De påpeker at de tydeligste utfordringene er at lærerne opplever å ikke ha tilstrekkelig kompetanse til å undervise, at det ikke er spesifisert antall timer til teknologi, og for tett integrering med sterkere fag som for eksempel naturfag. På denne måten

blir emnet nedprioritert selv med gode hensikter i læreplanene. Ting å arbeide med videre vil være å tydeliggjøre læreplanene, og å arbeide med lærernes kompetanse (Sanne et al., 2016, s73-76).

1.2.4 Digitaliseringsstrategi 2017-2021

I digitaliseringsstrategien fra 2017 legger regjeringen frem forslag til tiltak for å utnytte de digitale mulighetene i dagens skole. I tiltakene finner vi blant annet et forslag om å inkludere programmering og algoritmisk tenkemåte i læreplanen, og særlig i læreplanen for naturfag og matematikk (Kunnskapsdepartementet, 2017a, s.18). For læreres kompetanse setter de et delmål om at lærer og skoleledere skal ha høy profesjonsfaglig digital kompetanse, og det skal ligge til rette for etter- og videreutdanning. Noen av tiltakene regjeringen foreslår for å møte dette målet er i å utvikle et nettbasert videreutdanningstilbud for lærere. Utdanningen skal gjelde profesjonsfaglig digital kompetanse, og skal utvikles i sammenheng med videreutdanningsstrategien “Kompetanse for Kvalitet” (Kunnskapsdepartementet, 2017a, s.22-23). I dag ligger det nettbasert videreutdanning innenfor programmering på 15 studiepoeng ved 4 forskjellige universiteter tilgjengelig for lærere under Kvalitet og Kompetanse hos Utdanningsdirektoratet (Utdanningsdirektoratet, 2022).

1.3 Problemstilling

Det er nå snart to år siden innføringen av den nye læreplanen tredde i kraft. Da kom også innføringen av programmering som obligatorisk del i matematikken som for mange lærere og elever er et helt nytt tema. Det vil si at over de to årene har lærerne i større grad blitt kjent med temaet programmering, enten basert på eget arbeid, eller via utviklingsarbeid i regi av skolen. Målet for denne oppgaven er å utforske hva som har blitt gjort i skolene, og effekten dette har hatt på lærernes profesjonelle utvikling.

Denne oppgaven har som mål å ta for seg utviklingen etter innførelsen av programmering i ny læreplan. Som steg mot dette tar den for seg en problemstilling, med to støttende forskningsspørsmål:

- Hvordan har matematikklæreres utvikling i relasjon til programmering foregått etter fagfornyelsen, og hvilken effekt har det hatt?
 1. Hvordan har det blitt lagt til rette for lærernes profesjonelle utvikling etter fagfornyelsen?

2. Hvordan har matematikk-læreres situasjon utviklet seg etter fagfornyelsen?

Problemstillingen begrenser seg til lærere i matematikk, og som er ansatt i grunnskolen. Geografisk begrenser oppgaven seg til lærere som arbeider ved skoler som ligger i kommuner i Nordland.

1.4 Begrunnelse for valg av problemstilling

I likhet med andre land er programmering et nytt emne for mange av lærerne. Allerede før innføringen ble det dratt frem at selv om lærerne var positive til innføringen, manglet de den nødvendige kompetansen for å kunne undervise i programmering. Allerede i 2019 ble det fremhevet at lærere ønsket å lære med om hvordan de underviser i programmering (Berggren & Jom, 2019). Selv om det i flere år hadde vært tydelige signaler om at programmering ble et nytt tema i matematikk ble det ikke satt noen krav om videreutdanning av lærerne. Blant 450 lærerstudenter i Tromsø hadde kun 10 litt kjennskap til koding (Bjørkeng, 2020). Også etter innføringen er det flere lærere som opplever å ikke ha fått en tilstrekkelig innføring i programmering. I en NRK-artikkel kommer det frem at lærere ber om hjelp fra eksterne organisasjoner som LearnLink for hjelp til undervisning i koding. LearnLink som eksempel kontaktes flere ganger i uka, av lærere som fortsatt ikke har den nødvendige kompetansen til å undervise i programmering (Moreau, 2021).

To masteroppgaver skrevet i 2021 tok for seg emner relatert til denne oppgaven. De ble skrevet samme skoleår som den nye læreplanen ble gjeldende. Den første undersøkte: “Hva kjennetegner matematikklæreres holdninger rundt undervisning av programmering i teoretisk matematikk”, altså tok den for seg lærere ved videregående skole. Den fant at lærerne så nytteverdien og var motiverte for undervisningen videre. Det var derimot en del usikkerhet knyttet til eksamensformen, og manglende kompetanse blant lærere og elever. Det var også usikkerhet knyttet til lite samarbeid i kollegiet og noe frustrasjon på grunn av sen kursing i programmering (Zukanovic, 2021, s.57-58). Den andre oppgaven undersøkte “Hvordan er lærernes synspunkter og holdninger til innførelsen av programmering i skolen i lys av fagfornyelsen?”. Den tok for seg 15 lærere, syv fra ungdomsskolen, og åtte fra videregående skole. Her påpekes også manglende programmeringskompetanse blant lærere som en av utfordringene. Lærerne her er også misfornøyde med støtten som er gitt i sammenheng med fagfornyelsen, og mener at “det har vært lite tid til forberedelser, utilstrekkelig kursing, og en mangel på didaktikk som er tilpasset fagene”. Til tross for det er lærerne også her motiverte for undervisningen videre i programmering (Stenlund, 2021, s82).

Disse utfordringene har også vært tydelig gjennom tidligere praksisperioder i utdanning. Mangel på programmeringsundervisning i vårt kulls studieløp har vært diskutert både innad i studentgruppen, og har blitt luftet som frustrasjon ovenfor ansatte ved universitetet. Alle de nevnte faktorene pluss en personlig interesse i utvikling av organisasjoner og tjenester er bakgrunnen og motivasjonen for denne studien, der målet er å undersøke lærernes utvikling over de siste årene.

1.5 Struktureringen av oppgaven

Oppgaven er delt inn i 6 kapitler. Kapittel 1 består av introduksjon av oppgaven og problemstillingen. Kapittel 2 presenterer relevant teori og tidligere forskning, den nye læreplanen fra 2019, og om skoleutvikling. I kapittel 3 beskrives valget av metode, og hvordan datainnsamlingen og analysen er gjennomført. Funnene fra studien presenteres i kapittel 4. I kapittel 5 diskuteres funnene fra kapittel 4 opp mot teorien som er presentert i kapittel 2. Helt til sist kommer avsluttende refleksjoner og forslag til videre forskning i kapittel 6.

2.0 Teori

I dette kapittelet presenteres en gjennomgang av bakgrunnsarbeidet for at programmering nå er havnet i læreplanen, og på hvilken måte det har endret læreplanen. Fra læreplanen blir det sett på kompetansemålene, punktet om dybdelæring, og kravet om skoleutvikling. I sammenheng med kompetansemålene er det relevant å se på hva programmering innebærer, og de forskjellige formene programmering kan ta i skolen. Dybdelæring blir sett opp mot “Technological Pedagogical Content Knowledge”-modellen (Koehler et al., 2013; Mishra & Koehler, 2006), og skoleutvikling blir sett opp mot inkludering av programmering.

2.1 Læreplanens betydning for skolene

1. september 2017 ble overordnet del fastsatt. Overordnet del er en del av læreplanverket som definerer verdier og prinsipper for grunnopplæringen. Den beskriver det grunnsynet som skal være grunnlaget for all opplæring i grunnskolen, og gjør det tydelig hva som er skolens ansvar. Den definerer opplæringens verdigrunnlag, prinsipper for læring, utvikling og dannelse, og prinsipper for skolens praksis (Kunnskapsdepartementet, 2017b).

Fra høsten 2017 til våren 2018 ble det arbeidet med å utvikle kjerneelementene for den nye læreplanen. For matematikk ble de valgte kjerneelementene utforskning og problemløsning, modellering og anvendelser, resonnering og argumentasjon, representasjon og kommunikasjon, abstraksjon og generalisering, og matematiske kunnskapsområder. Tanken er at elevene skal jobbe mer med metodene og tankegangen bak, slik at de får en større forståelse. Særlig for utforskning og problemløsning er aktuelt i sammenheng med programmering og algoritmisk tankegang. Spesielt algoritmisk tankegang er en viktig del av å lære seg å bryte ned problemer i delproblemer og så finne løsninger (Kunnskapsdepartementet, 2019b).

Fra høsten 2018 til høsten 2019 gjaldt arbeidet det å utvikle læreplanene. Utdanningsdirektoratet samarbeidet med lærere og pedagoger for å utvikle læreplanene. Underveis ble skisser av læreplanene sendt ut og det var mulig å komme med innspill til arbeidet videre (Utdanningsdirektoratet, 2021).

De norske læreplanene delegerer flere plikter til skoleeierne. Pliktene at skoleeier har ansvar for at det som skjer i skolen skjer i tråd med læreplanene og deres hensikter og mål. Med det menes både det faglige som foregår i klasserommet, og hvordan skolen som helhet skal operere.

Fra læreplanen vil punktet om kompetansemålene i programmering i matematikkfaget, dybdelæring, og behovet for skoleutvikling være relevant for denne oppgaven. For å oppnå disse målene vil det innebære omfattende utviklingsarbeid lokalt ved skolene. Både med et mål om å komme frem til en felles praksis for hvordan man kan gå frem for at elevene skal nå kompetansemålene, og om å sørge for at lærerne er i stand til å følge det som blir skolens praksis. I sammenheng med en ny læreplan vil dette være både arbeid både på kort og på lang sikt, med et mål om kontinuerlig å bedre kvaliteten i undervisningen (Lyngsnes & Rismark, 2014, s.177). Kunnskapsløftet har en kollektiv vinkling på skoleutvikling, der det spesifiseres at alle ansatte i skolen har en plikt til å delta aktivt i utviklingen av det profesjonelle læringsfelleskapet i skolen, og dermed også å delta aktivt i utviklingen av skolen (Kunnskapsdepartementet, 2017b).

Det mest konkrete som er innført i læreplanen angående programmering er kompetansemålene. Her er det matematikk som har fått flest kompetansemål som omhandler programmering. Her er det kommet inn kompetansemål som gjelder programmering i alle klassetrinn fra og med 2. trinn. Det viser at myndighetene ønsker en tidlig satsing på programmeringsundervisning, på et nivå som er passende for elevene.

- I 2. til 3. trinn er målene basert på det å lage enkle algoritmer i sammenheng med lek og spill. Altså skal elevene ikke enda programmere på datamaskiner, men begynner allerede her arbeidet med algoritmisk tenkning.
- I 4. og 6. trinn er fokuset på å lage og å programmere algoritmer ved bruk av enkle programmeringsstrategier.
- I 7. til 10. trinn skal elevene bruke programmering til å utforske, simulere, og beregne modeller og egenskaper, og de skal teste og forbedre egne algoritmer. (Kunnskapsdepartementet, 2019b)

Kompetansemålene definerer hva elevene skal lære i løpet av skolegangen, og legger dermed også føringer for hvilken kompetanse lærerne bør inneha.

2.2 Programmering

I kompetansemålene brukes ordet programmering flere ganger. “Programmering” som et større konsept består av flere elementer, blant annet algoritmisk tenkning og koding som er relevant for denne studien. For å forstå ordet programmering kan man ta utgangspunkt et “program”. Et program er et sett med instruksjoner som trinnvis beskriver en fremgangsmåte, for eksempel kan man sammenligne en matoppskrift med et program. Prosessen med å skrive et slikt program

som en maskin kan forstå, kan kalles programmering (Bueie, 2019, s.22-23). I motsetning til noen som følger en oppskrift kan ikke en datamaskin kunne gjøre endringer underveis, da den kun gjør det vi gir den beskjed om gjennom oppskriften (Haraldsrud et al., 2020, s.17-18).

Programmering er på mange måter ulikt det det man opplever i hverdagen ellers, og det kan bidra til at introduksjonen av programmering kan være vanskelig for mange. Til tross for erfaring med andre digitale verktøy kan både algoritmisk tenkning og koding være vanskelig å lære seg. Man kan tenke at å lære seg programmering er som å lære seg et nytt språk, men et språk som har høye krav for rettskriving da små “grammatikkfeil” kan gjøre at datamaskinen ikke er i stand til å lese programmet (Kluge, 2021, s.132-133). Om man kommer over disse hindringene kan programmering gi oss nye muligheter for utforskning og forståelse av konsepter. Programmering i skolen kan på denne måten bidra til økt dybdelæring med hvordan algoritmisk tenkning, problemløsningskompetanse og mulighetene programmering gir på tvers av fag (Haraldsrud et al., 2020, s.14; Kluge, 2021, s.144). Men når programmering skal innføres i matematikkfaget, bør man utnytte mulighetene programmering gir for problemløsning og utvikling av nye metoder, og målet i undervisningen bør være arbeide mot en generelle problemløsningsmetoder (Gjøvik & Torkildsen, 2019).

2.2.1 Algoritmisk tenkning

Digital og algoritmisk tenkning henger tett sammen med hva man tenker at man lærer i sammenheng med å lære programmering. Digital tenkning kan ses på som et litt større begrep enn algoritmisk tenkning, og innebærer et mer overordnet syn. Digital tenkning kan innebære å vite hvordan man velger rett digitalt verktøy for en oppgave, og å kunne se hvordan man kan anvende de digitale verktøyene til å løse nye utfordringer. I dagens digitale samfunn kan man si at det å beherske digital tenkning er viktig for å kunne øke forståelsen av verden rundt oss, og for å kunne reflektere rundt ny teknologi (Kluge, 2021, s.142-143).

Algoritmisk tenkning er litt vanskeligere å definere, noe som kan gjøre det vanskeligere å undervise eller vurdere. Men man kan si at det handler mer om å kunne ta et større problem og dele det inn i flere del-problem. Så skal man ta de små del-problemene og lage stegvise løsninger, som kan gjennomføres et steg av gangen for å løse problemene. På den måten ønsker man å kunne ta et større problem og bryte det ned i flere mindre, men mer løsbare problemer (Kluge, 2021, s.142-143). Videre kan en algoritme defineres som en ordnet liste med instruksjoner eller steg, for å finne et svar, eller løse et problem. Med bakgrunn i det kalles denne tankegangen

der man ønsker å bryte et større problem ned i mindre og stegvis løsbare problemer for algoritmisk tenkning (Bueie, 2019, s.28). Elever kan derimot ha vansker med denne arbeidsmåten, noe som kan skyldes lite erfaring i å arbeide med denne typen problemløsning. Det kan også skyldes lite erfaring eller kompetanse i programmeringen selv, noe som også vises i at det oppfattes som at det er en sterk kobling mellom algoritmisk tenkning og programmering (Crick, 2017, s.13).

Når man tenker algoritmisk vurderer man hvilke steg man må gjennomføre for å kunne løse et problem, og kunnskap om hva et digitalt verktøy vil kunne hjelpe med. Det er en problemløsende og eksperimenterende prosess der man å være åpen for nye fremgangsmåter og alternative løsninger. Det innebærer også å ha verktøyene som trengs for å kunne finne og korrigere eventuelle feil i “algoritmen”. I figuren “Den algoritmiske tenkeren” presenterer UDIR de viktigste elementene ved algoritmisk tenkning. Den inneholder en oversikt over både viktige begrep, og viktige arbeidsmåter. Disse punktene er med på å definere hvilke ideer lærerne må være i stand til å formidle i undervisning, og definerer kompetansene de må ha.



Figur 1: “Den algoritmiske tenkeren” (Utdanningsdirektoratet, 2019)

2.2.2 Programmeringsspråk

I tillegg til algoritmisk tenkning er programmeringsspråk en stor del av “programmering”. Programmeringsspråk er en betegnelse på et skriftlig språk som brukes til å instruere maskiner. Språkene består av mulige instruksjoner, og grammatiske regler for hvordan man må strukturere dersom man skal sette instruksene sammen til et større program slik at det kan forstås og leses av en maskin. Språkene som er relevant for denne oppgaven er Scratch, Micro:bit, Python, og Minecraft Education Edition.

2.2.3 Blokkbasert programmering

En populær tilnærming til “enklere” programmering enn tradisjonell tekstbasert programmering er blokkbasert programmering. Her programmeres det ved hjelp av blokker istedenfor via tekst, og man kan fokusere mer på konseptene i programmeringen, heller enn syntaksen og det å få “kodingen” rett (Haraldsrud et al., 2020, s.87-88). I blokkbasert programmering har man et grensesnitt med “blokker” som tilsvarer programmeringsfunksjoner. Så heller enn å skrive koden, manipulerer man blokker for å endre koden. Det gjør det enklere for nybegynnere både å arbeide med den algoritmiske tankegangen, og til å eksperimentere med kodingen. Fokuset blir på å forstå hvordan programmet og programmeringen virker, og å se sammenhengen mellom koden og hva programmet gjør. Ved blokkbasert programmering får man altså ned terskelen da det ikke er nødvendig å kunne skrive koden, og man kan fokusere på tankegangen bak kodingen (Kluge, 2021, s.143). Men den forenklingen kan man enklere la elevene utforske og prøve seg frem. Det ikke er nødvendig for elevene å ha inngående kunnskap om hvordan koden må struktureres, siden blokkene har en mer intuitiv og visuell struktur, og innlagte begrensninger på hvordan forskjellige kodeblokker kan kobles sammen. Blokkbasert programmering er altså et pedagogisk praktisk verktøy for undervisning i grunnleggende programmering. Det er spesielt relevant for lavere trinn, da frem til elevene er klare for å gjøre overgangen til tekstbasert programmering (Piedade et al., 2020, s.3).

Det fins i dag flere gode eksempler på verktøy hvor man kan arbeide med blokkbasert programmering. En av de mest populære er Scratch, og er utviklet av MIT, med en tanke om nettopp å være en inngang til programmering. På den måten kan man få utforske tankegangen bak programmering, men uten å måtte forholde seg til et tekstbasert språk (Bueie, 2019, s.26-27). Scratch har flere faktorer som gjør at det passer fint til bruk i skolen, blant annet ligger det enkelt tilgjengelig på nett. Det har også en fullt norsk oversettelse, og på nett fins det mange lett tilgjengelige undervisningsopplegg for undervisning i Scratch. Et annet verktøy som brukes

er Micro:bit, som er en liten programmerbar datamaskin. Den kan programmeres både via tekstbasert MicroPython, eller en blokkbasert tilnærming. Den er også fysisk enhet og gir muligheter knyttet til fysisk programmering (Monk, 2017, s.14-22).

2.2.4 Fysisk programmering

I likhet med at blokkprogrammering får ned terskelen for bruk av programmering i undervisningen kan fysisk programmering både få ned terskelen, og øke motivasjonen og engasjementet blant elevene. Fysisk programmering innebærer programmering av små digitale enheter der man kan se og direkte identifisere hva koden gjør (Kluge, 2021, s.135). Man har for eksempel “Bee-bot”, som er en enkel robot med syv knapper på ryggen. Her har man fire knapper for bevegelse, en knapp for forover, en for bakover, en knapp på hver side som får roboten til å vri seg 90 grader i den retningen. De tre siste knappene er “kjør”, som kjører gjennom det elevene har programmert, “slett” som fjerner all kode, og “pause” som setter programmet på pause. Med disse enkle kommandoene kan elever programmere en robot til å for eksempel følge en løype som er markert på gulvet, og kan oppleves av elevene både som spennende, motiverende, og direkte givende da de enkelt ser resultatet. For elever ved høyere trinn vil bruk av Micro:bit være mer relevant. Micro:bit som maskin har mange muligheter. På selve maskinen finner du 25 små programmerbare lys, programmerbare knapper, et innebygd akselerometer, kompass, og en liten radioenhet som kan brukes til å kommunisere mellom to enheter. Med disse mulighetene er det mye rom for forskjellige utfordringer og oppgaver som kan løses, og er derfor et godt verktøy til bruk i skolen (Monk, 2017, s.14-22).

Videre kan elever ved bruk av fysiske roboter til programmeringsundervisning få en bedre innsikt i problemløsningsprosessen. De kan se og oppfatte problemet fysisk, så arbeide med å abstrahere vekk slik at de sitter igjen med essensen av problemet, og så gå i gang med problemløsningsarbeidet. Når elevene starter på arbeidet med å utvikle planer, algoritmer og et ferdig program, vil det også i bruk av roboter være mer visuelt og håndfast hvor et eventuelt problem i programmet er. På bakgrunn av dette kan bruk av fysisk programmering være et godt startpunkt i tidlig programmeringsundervisning (Piedade et al., 2020, s.8).

2.2.5 Tekstbasert programmering

Python har flere pedagogiske og faglige styrker som bidrar til at det kan være et naturlig språk å velge for programmeringsundervisning i skolen. For det første er Python gratis, enkelt

tilgjengelig, og det kan kjøres på Windows, Mac, og Linux. For det andre er Python et lettlest språk noe som gjør det til et bra språk for nybegynnere, da det er ganske intuitivt å vite hva koden gjør dersom man leser koden (Haraldsrud et al., 2020, s.14). For det tredje er Python også et språk som elevene kan ha bruk for videre, da det også brukes utenfor skolen. Det gjør også at det finnes et godt utvalg med ferdigskrevne koder, noe som er nyttig i naturfag eller matematikk, der det kan være behov for komplisert kode som gjør veldig spesifikke ting (Bueie, 2019, s.30-31). Flere av disse faktorene gjør også Python til et godt egnet språk for overgangen fra blokkprogrammering til tekstprogrammering. Da den største endringen blir behovet for å selv sørge for at koden skrives rett, blir det mye nytt for elever og lærere (Haraldsrud et al., 2020, s.102-103). Med Python som lettlest kode kan det bli en enklere overgang, der elevene fortsatt kan fokusere mer på problemløsningen og tankegangen, heller enn kun fokusere på syntaksen (Piedade et al., 2020, s.15). Da får også elevene mer øving i å lese kode, og kan arbeide mer sammen i både lesingen og tolkningen av koden (Crick, 2017, s.9-10).

2.3 Dybdelæring

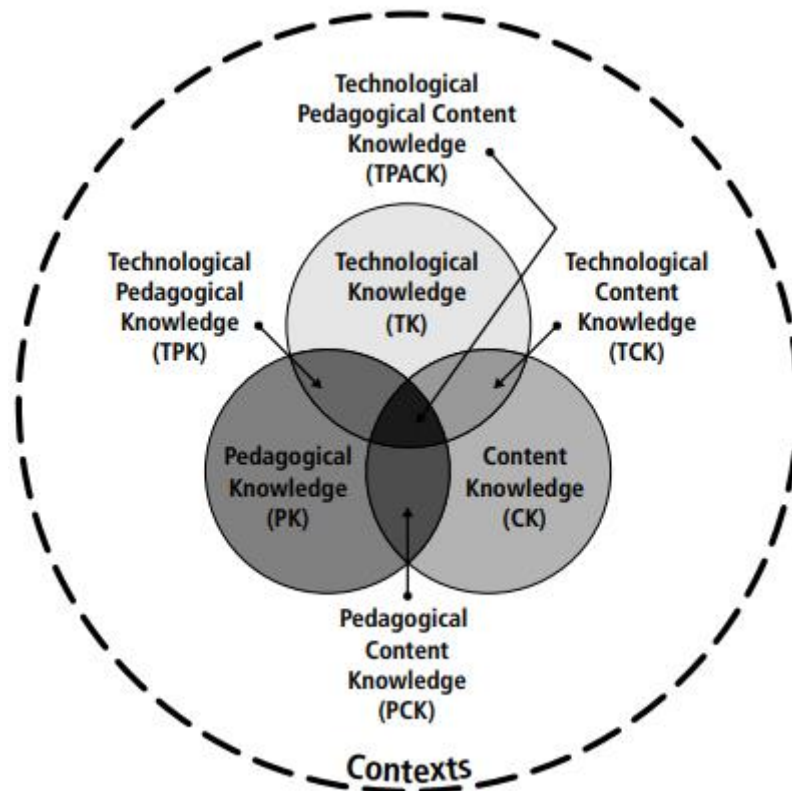
I tillegg til endringene i læreplanen for fagene, er det også nye elementer i overordnet del. I punkt 2.2 i overordnet del, Kompetanse i fagene, dras dybdelæring også inn som et viktig element i dagens skole. Skolen skal legge til rette for dybdelæring for elevene, slik at de kan oppnå en mer helhetlig forståelse av emnene. Dybdelæring vil si å utvikle varig kunnskap og kompetanse i fag. Basert på det kan man oppnå en varig forståelse av fagene, og av sammenhenger mellom fag og mellom fagområder, og dermed oppnå den mer helhetlige forståelsen. På den måten utvikler man også en dypere faglig forståelse, slik at man kan bruke kunnskapen sin i både kjente og ukjente situasjoner (Kunnskapsdepartementet, 2017b). God dybdelæring er derimot krevende for skolene å få til å realisere. God dybdelæring stiller nemlig høye krav både til lærernes faglige kompetanse, deres pedagogiske kompetanse, og til hvordan lærerkollegiet sammen utvikler lærernes arbeid med elevene (Kunnskapsdepartementet, 2014, s.41).

Spesielt i sammenheng med programmering er det utfordrende med dybdelæring. Selv om elevenes læringsutbytte øker gjennom dybdelæringen, er dette et krevende fagområde i dagens skole, og med begrenset plass i timeplanen (Kluge, 2021, s.145-147). Altså er programmering et område der mange lærere mangler den nødvendige kompetansen til å kunne legge til rette for dybdelæring, og det byr på flere utfordringer. Læreres kompetanse og materielle tekniske utfordringer vil variere fra skole til skole, og vil gjøre kvaliteten i undervisningen svært variert

(Sentance & Csizmadia, 2017, s.479-481). Teknologi og programmering i skolen er generelt temaer som er utfordrende både å forske på, utdanne, og å vurdere læreres praksis i (Crick, 2017, s.3-4). Det fins også som en konsekvens lite forskning på god pedagogisk bruk av verktøy, både fysiske og digitale, i programmeringsundervisning (Crick, 2017, s.17-18). Det å utvikle den nødvendige IKT-kompetansen lærere trenger for å kunne legge til rette for dybdeløring i programmering kan altså være utfordrende (Nilsen, 2010, s.107). Når programmering også bare er en del av matematikkfaget og ikke et eget fag må det også konkurrere med tradisjonell matematikkundervisning om tiden, og for lærere med svak programmeringskompetanse kan programmering fort gå mindre tid programmeringsundervisning til fordel for mer inngående undervisning i tradisjonell matematikk. Lærere trenger altså mer kompetanse, mer øving, og mer støtte for utvikling av lærernes pedagogiske evner for dybdeløring i programmering (Sentance & Csizmadia, 2017, s.489).

2.3.1 Teknologisk pedagogisk fagkompetanse

For å vurdere lærernes pedagogiske evner i sammenheng med teknologi kan vi benytte Technological pedagogical content knowledge (Teknologisk pedagogisk fagkompetanse) som rammeverk. Det beskriver de tre viktigste kompetansedimensjonene for teknologiundervisning. Den bidrar til å synliggjøre den komplekse sammenhengen av kompetanser som lærere må inneha for å kunne drive god pedagogisk undervisning med bruk av IKT. Modellen ble utviklet for å synliggjøre hvilke kompetanser lærerne er nødt til å utvikle (Nilsen, 2010, s.109-110). Kompetansedimensjonene den tar hensyn til er “Content knowledge”, eller fagkompetanse (C). Dette er den faglige kompetansen læreren skal inneha for å undervise i sitt fag. Det andre er “Pedagogical knowledge”, eller pedagogisk kompetanse (P). Dette er den kompetansen læreren trenger for å kunne drive god undervisning (Mishra & Koehler, 2006, s.1021). Sist har vi “Technological knowledge”, eller teknologisk kompetanse. Denne omhandler kompetansen som gjelder bruk av teknologi, da teknologien endrer grunnlaget for undervisningen. Den vil altså ha påvirkning på hvordan undervisningen kan gjøres (Mishra & Koehler, 2006, s.1025).



Figur 2: "The TPACK Framework and its content knowledge" (Koehler et al., 2013, s.15)

I skjæringspunktene mellom disse tre kompetansedimensjonene finner vi diverse sammensetninger. Teknologisk pedagogisk kompetanse handler om å vite hvordan man bruker teknologi på en pedagogisk måte i undervisningen. Teknologisk fagkompetanse handler om hvordan teknologi kan bidra til faget, og hvordan teknologien kan brukes i faget. Pedagogisk fagkompetanse handler om hvordan man underviser i et spesifikt fag eller emne, og hvordan man gjør undervisningen forståelig for elevene. I midten av disse tre finner vi teknologisk pedagogisk fagkompetanse. Dette omhandler hvordan man kan gjennomføre god pedagogisk undervisning, med god faglig kompetanse, og kompetanse i hvordan man kan anvende teknologi som en del av hele undervisningen. Altså omhandler det om evnen til å gjennomføre pedagogisk undervisning på et godt nivå, med bruk av teknologi (Koehler et al., 2013, s.16; Mishra & Koehler, 2006, s.1028-1029). Disse aspektene legger grunnlaget for hva lærere bør kunne for teknologisk undervisning, også undervisning i programmering.

2.4 Skoleutvikling

For at skolene skal kunne nå målene som er satt i forhold til undervisning av programmering, og dybdelæring er skolene også pålagt å drive skoleutvikling. Skoleutvikling som begrep kan

tolkes i flere retninger. Det kan referer både til utviklingen på nasjonalt nivå i form av læreplaner, til et lands skolehistorie, eller det kan gjelde det lokale utviklingsarbeidet som skjer i skolene (Mausethagen & Helstad, 2019, s.17). Skoleutvikling og utviklingsarbeid har som mål å gjøre skolen til en lærende organisasjon. En lærende organisasjon er en organisasjon der medlemmene og organisasjonen som helhet har et ønske, og gjennomfører en kontinuerlig utvikling (Nilsen, 2010, s.108, s.114-115). Skolen i dagens samfunn har mange endringer den skal forholde seg til. Teknologi, globalisering, og internasjonalt samarbeid er alle nye utfordringer som skolen må forholde seg til, og finne løsninger som resulterer i positiv endring for elevene (Imsen, 2016, s.540-541). Det er også poengtert i læreplanen, der det spesifiseres at skolen i et profesjonsfaglig felleskap skal drive utvikling (Kunnskapsdepartementet, 2017b).

Dette utviklingsarbeidet har to dimensjoner, profesjonalisering ovenfra, og profesjonalisering innefra. Profesjonalisering ovenfra har sammenheng i øvre nivåers ønsker og mål om utvikling, og kan ses på som bestemt eller påtvungent utviklingsarbeid. Her vil læreplanen være et godt eksempel, der endringer i læreplanen skal føre til endringer i læreres praksis. Denne typen profesjonalisering handler om at man tenker at det finnes en sammenheng mellom tiltak som gjøres politisk nivå, og arbeidet som gjøres i skolene (Dons, 2010, s.85). Profesjonalisering innefra handler derimot om et ønske fra medlemmene i et "nivå" for utvikling (Mausethagen & Helstad, 2019, s.17). Utviklingsarbeid er derimot sjelden enten eller, men ofte en blanding av profesjonalisering ovenfra, og innenfra. For eksempel vil lokalt arbeid med læreplaner ofte ha utspring i læreres eget ønske om å bearbeide og lage en felles forståelse av læreplanene. Lærerne har også rapportert et ønske om mer utviklingsarbeid og faglig oppdatering, og det gjør det viktig at skolene og utdanningssystemet legger til rette for dette (Kunnskapsdepartementet, 2008, s.40). Dersom man eksklusivt har profesjonalisering ovenfra kan det oppleves som spesielt påtvunget av lærerne, og man kan risikere at de ikke får et eierforhold og en personlig investering i arbeidet, og det kan påvirke motivasjonen for arbeidet (Imsen, 2016, s.542-548). Dersom profesjonalisering skjer kun innenfra kan det oppleves som vanskeligere å definere og å bli enige om klare felles mål.

2.4.1 Myndigheter

Myndighetenes ansvar i skoleutvikling ligger i stor grad i å legge mål og rammer for utviklingen. Politiske endringer og reformer av læreplanen er viktige virkemidler som myndighetene har for å påvirke skolens praksis. Myndighetene har også prøvd å delegere ansvaret for skoleutviklingen i større grad eksklusivt over på kommunene og skolene, men har

senere sett behovet for mer nasjonal støtte, og tatt en mer aktiv del i skoleutviklingen (Kunnskapsdepartementet, 2016, s.94-96). I Kvalitet for skolen argumenteres det for at denne endringen ble gjort fordi skolene selv ikke hadde den nødvendige kompetansen for å nå målene som var satt i den nye læreplanen (Kunnskapsdepartementet, 2008, s.7). Mangelen på nødvendig kompetanse blir spesielt en utfordring for mindre skoler, der man naturlig vil ha et mindre profesjonsfelleskap, som gir mindre muligheter for profesjonsutvikling i kollegiet. Mindre skoler er tradisjonelt også landlige, noe som det mer krevende både i form av tid og ressurser for lærerne herfra å delta på kurs eller andre opplegg (Abalde, 2014, s.15). Den typen profesjonalisering ovenfra som myndighetene kan bidra med har tradisjonelt sett hatt et mål om å eksistere sammen med, og å støtte profesjonaliseringen innenfra (Mausethagen & Helstad, 2019, s.17). I sammenheng med den nye læreplanen ble det utviklet kompetansepakker som støtte for emnene. Her ble også programmering inkludert. Disse var tilgjengelige digitalt og fungerer som støtte til profesjonalisering innenfra. De er også digitale som gjør at de kan oppdateres i samarbeid med eksterne kompetansemiljøer med relevant fagkompetanse (Dolonen et al., 2019, s.21).

Lærerutdanningene er også et virkemiddel myndighetene har for å påvirke utviklingen av skolene (Imsen, 2016, s.542-548). Lærerutdanningene bør sikre at alle lærerstudenter har profesjonsfaglig digital kompetanse, og at de som skal undervise i fag der programmering skal brukes også har vært gjennom formelle kurs (Kunnskapsdepartementet, 2017a, s.22-23). Å inkludere programmering i fremtidige lærerstudenters utdanning kan også bidra til høyere selvtillit, høyere programmeringskompetanse, og høyere problemløsningskompetanse, som alle er relevante for undervisning i programmering (Bati, 2021, s.13).

For lærere som allerede er i arbeid er det viktig at det legges til rette for etter- og videreutdanning i emnet. Etter kunnskapsløftet i 2006 ble det satt av midler for å styrke lærernes digitale kompetanse, men dette ble i stor grad brukt på etterutdanning som ikke ga formell kompetanse. Utvikling av uformell kompetanse har vist seg å ofte resultere i mindre endringer som varer kortere. Her har utvikling av formell kunnskap i et profesjonsfelleskap ved skolene har vist bedre resultater (Kunnskapsdepartementet, 2008, s.39-40). Det er allikevel viktig med målrettede statlige bidrag av ressurser til videreutdanning.

2.4.2 Ledelse

Ledelsens ansvar vil blant annet være å fungere som et mellomledd mellom myndighetene og lærerne selv. Ledelsen har ansvaret for at det som skjer i skolen skjer i tråd med læreplanene, blant annet via å legge til rette for lokalt utviklingsarbeid for å kunne utvikle skolens praksis til å være i tråd med nye nasjonale læreplaner (Mausethagen & Helstad, 2019, s.17). I å legge til rette for arbeidet ligger det flere hovedaspekter som er viktig for god utvikling.

Det første er å ha konkrete målbare mål for arbeidet. I dette arbeidet bør lærerne selv inkluderes tidlig i prosessen for å komme sammen om felles mål for utviklingen. Her kan man også delegerer noe av kommunikasjonen mellom lærere og ledelse til teamledere, og på den måten få ledelsen nærmere lærerne (Mausethagen & Helstad, 2019, s.175). Med denne tilnærmingen kan man i større grad involvere lærerne, og øke deres personlige eierskap, og deres følelse av personlig ansvar for utviklingen. At lærerne har kjennskap til, og føler et eierskap til målene er en viktig faktor for suksess i utviklingsarbeidet (Nilsen, 2010, s.114-115; Imsen, 2016, s.542-548).

Det andre er å legge til rette for arbeidet. Her ligger både delegering av tid, og å sørge for at ressursene for arbeidet er til stede (Dons, 2010, s.85; Irgens, 2010, s.141-142). Delegering av tid for utviklingsarbeid er noe både lærerne og rektor opplever som en utfordring. Ledelsen på sin side kan oppleve at det er vanskelig å legge til rette for felles tid på en slik måte at lærerne også får den tiden de trenger til deres individuelle arbeid. Lærerne selv kan ha vidt forskjellige timeplaner noe som kan gjøre det vanskelig å sette av felles tid (Irgens, 2010, s.128-130). Denne utfordringen er noe lærere kjenner godt til, da arbeidstiden ofte kan være presset, med undervisning som skal planlegges, og gjennomføres, og det etterarbeidet som følger (Mausethagen & Helstad, 2019, s.168). Den typen opplegg som oftest lar seg gjennomføre er fellesmøter og teammøter. Teammøter opplever lærere oftere som mer produktiv enn fellesmøter. I teammøter kan man ha tettere samarbeid om mindre mål, noe som kan føre til en mer konsentrert innsats for utvikling. Fellestiden kan derimot risikere å bli oppfattet som en påtvungen lite produktiv bruk av tiden. Dersom ledelsen ikke har konkrete mål, og en plan for å inkludere lærerne aktivt i utviklingen, risikerer man at lærerne føler at man i fellesmøtene bare ønsker å “fylle tiden”, uten noe større mål for arbeidet (Eftedal & Damsgaard, 2014, s.72-73).

Å sørge for tilgjengelige ressurser er som nevnt også en viktig del av tilrettelegging for utviklingsarbeid, spesielt for utvikling for bruk av IKT i undervisningen. Den materielle tilgangen på utstyr vil kunne ha konsekvenser for utviklingsarbeidet. Uten tidlig tilgang på det utstyret som skal brukes i undervisningen vil det være veldig vanskelig for lærere å tilegne seg den kompetansen som skal til for nettopp bruken i klasserommet. Den typen utstyr som kan være relevant kan være datamaskiner, roboter, eller lærebøker (Imsen, 2016, s.542-548). Det er også viktig at lærerne får god pedagogisk kompetanse for bruk i klasserommet, da tilgang på utstyr ikke er nok i seg selv (Dolonen et al., 2019, s.11).

2.4.3 Lærere

Som en følge av at lærernes kunnskapsgrunnlag i stor grad er basert på erfaringsbasert kunnskap, noe som er personlig har utviklingsarbeid blitt sett på som et personlig ansvar som hver enkelt lærer har. Denne typen organisering av utviklingsarbeid legger til rette for en dårligere samarbeidskultur innad i organisasjonen, og gir dårligere resultater for felles læring i kollegiet (Kunnskapsdepartementet, 2004, s.27). Lærerne eksisterer i kollektive miljøer innad i skolen, og jobber dermed også ofte i team i det daglige arbeidet (Lyngsnes & Rismark, 2014, s.193-194). Som følge av dette har både politikere, forskere, og lærerne i nyere tid søkt større grad av profesjonalisering i form av akademisk kunnskap og gjennomføring av felles utviklingsarbeid i skolene (Mausethagen & Helstad, 2019, s.17). For dette er det viktig at endringskompetanse løftes frem hos dagens og fremtidens lærere (Lyngsnes & Rismark, 2014, s.182-183). Dette felleskapet kan også inkludere eksterne ressurspersoner som har den nødvendige kompetansen til å fungere som støtte for utviklingsarbeidet dersom man får til et samarbeid mellom partene. Å få behovene til skolene og tilbudet fra eksterne ressurspersoner kan derimot være komplisert, men vil være avgjørende for resultatene av arbeidet (Mausethagen & Helstad, 2019, s.167-181).

2.5 Utvikling

Som nevnt er utviklingen av lærerutdanningen et viktig virkemiddel i skoleutviklingen. Under lærerutdanning går både den grunnleggende lærerutdanningen, og etter- og videreutdanning av praktiserende lærere. Fra Japan kan man se at ved innføringen av programmering var det blant lærerne over 80% som var positive til innføringen, mens over 60% opplevde lite selvsikkerhet og selvtillit knyttet til innføringen. Lærerne hadde ikke nok erfaring med bruk av IKT i skolen,

var sterkt knyttet til tradisjonell undervisning, og hadde lite tid til å drive utviklingsarbeid. Dette er tre faktorer som trekkes frem som utfordringer som må håndteres i skolene for å få en god innførelse av programmering (Ohashi, 2017, s.133-136).

2.5.1 Skaperverksteder

Innføringen av programmeringen skjer i hele verden, også i våre naboer i de nordiske landene. I Sverige og Finland har det vært gjennomført et opplegg i samarbeid mellom lærere, lærerutdanninger, og såkalte “Makerspaces”, eller skaperverksteder. Skaperbevegelsen bygger på en ide om at man lærer når man får prøve, utforske, og ta del i praktiske oppgaver. Skaperverksteder er fysiske steder som legger til rette for denne typen arbeid, med verktøy, materialer, og kompetansepersonell som grunnlag for læringen. Bruken av denne typen opplegg i lærerutdanningene hadde positive resultater for lærerstudentene og lærerne som deltok. Oppleggene var praksisnære, og det kan bidra til mer engasjement fra lærerne (Kjällander et al., 2018, s.25-28; Kong & Lao, 2019, s.981). Kursene som kan arrangeres i samarbeid med skaperverkstedene kan ta for seg forskjellige kompetanseområder, og kan bidra til å forsterke noe av den videreutdanningen som allerede er tilgjengelige for praktiserende lærere.

2.5.2 Lærerutdanning for programmering

Som en del av programmeringskursene i utdanningen er det flere viktige elementer som inkluderes. For eksempel er algoritmisk tenkning et viktig tema, både hva det er, men også hvordan det kan bidra til undervisningen, også for andre områder enn programmering (Sands et al., 2018, s.161-163). For eksempel bør programmering inkluderes også i kurs som gjelder digitale ferdigheter generelt. Dette har en positiv effekt også på den digitale kompetansen som gjelder andre emner enn programmering (Woodrow, 1992, s.214-216). For de lærerne som derimot ser på programmering som et eget separat emne, og ikke som et verktøy som kan inkluderes i faget som en helhet, oppleves planlegging som en spesiell utfordring (Rouhani et al., 2020, s.54-56). Lærernes deltakelse i videreutdanninger med fokus på programmeringskonsepter og god pedagogisk praksis kan føre til store forbedringer i lærernes forståelse av teknologi og programmering. Spesielt med fokus også på god pedagogisk praksis kan det bidra til store positive resultater for alle elementene i TPCK-modellen (Kong et al., 2020, s.16). Lærere som oppnår den nødvendige grunnkompetansen og som selv kan eksperimentere med programmering vil ha en sterkere forståelse for hvilke muligheter

programmering kan gi, og for hvordan man kan gjøre eksperimenterende undervisning (Rich, 2017, s.13).

Det er også nyttig å inkludere bruk av fysisk programmering i kursene, da dette er noe som brukes i skolene. I sammenheng med lærerne og lærerstudentenes utvikling bør det altså dedikeres noe tid til pedagogisk bruk av programmerbare roboter i undervisningen (Sisman & Kucuk, 2018, s.317-318). En god modell for et slikt arbeid kan være å ta utgangspunkt i opplegg som benytter seg av fysisk programmering, og så arbeide ut fra det (Piedade et al., 2020, 9). Dette bidrar både til en bedre pedagogisk praksis når det kommer til undervisningen, men det har også positive effekter på deltakernes evne til algoritmisk tenkning og feilsøking, noe som begge er viktige kompetanser som læreren bør ha også for undervisningen i programmering generelt (Angeli, 2021, s.7-8).

Men i utformingen av kursene kan det også være relevant å vurdere hvem kurset skal designes for. I et gitt kurs vil det være vanskelig å dekke alt innenfor programmering, så man må gjøre vurderinger på hvilken kompetanse lærerne trenger for å sikre et godt nivå på undervisningen (Dağ, 2019, s.304-308). Dette kan for eksempel deles inn basert på hvilken tidligere kompetanse deltakerne har, da noen former undervisning kan for eksempel ha en bedre effekt for nybegynnere enn for viderekommende. Det kan også deles inn etter hvilke trinn lærere skal undervise ved, da en lærere ved barnetrinnet vil ha andre krav enn en lærere ved ungdomsskolen (Angeli, 2021, s.7-8; Vinnervik, 2022, s.21-22). For de laveste trinnene kan kursing i bruk av Scratch og Micro:bit være tilstrekkelig, mens fra mellomtrinnet og opp vil Python i større grad være naturlig. For noen konsepter innenfor programmering vil bruk av Scratch eller Micro:bit kunne bidra til nesten lik forståelse etter kurs som ved bruk av Python, og vil dermed kunne være tilstrekkelig for de trinnene der tekstbasert programmering ikke skal undervises (Changpetch et al., 2022, s.9-16). For noen konsepter kan også deler av undervisningen skje helt uten bruk av datamaskiner eller programmering (Sands et al., 2018, s.161-163).

Gjennom kursene må lærerne, på tross av deres varierende utgangspunkt, kunne få den nødvendige kompetansen til å undervise i programmering. Dette inkluderer å variere undervisningen, da forskjellige arbeidsmåter har forskjellige fordeler også innenfor programmering (Erümit, 2020, s.1032-1034). For å bidra til dette kan er det veldig aktuelt med kurs som tar utgangspunkt verktøy som er tilgjengelige for de aktuelle lærerne, og som selv tilbyr støttemateriell til undervisningen. For eksempel har code.org, Scratch, og Micro:bit alle

støttmateriell til undervisning, og et kurs for lærere som ser på hvordan disse kan brukes i undervisningen vil bidra til å utvikle deres pedagogiske forståelse av undervisning i programmering. Her kan eksisterende lærerutdanninger og videreutdanninger være mangelfulle, da god pedagogisk bruk fortsatt er noe som det trengs mer forskning rundt (Falkner & Vivan, 2015, s.396-398; Menekse, 2015, s.22). Lærere må også kunne kjenne på en tilstrekkelig kompetanse, motivasjon, og selvtillit i egne evner til å kunne undervise. Både i sammenheng med emner i lærerutdanninger og formelle kurs gjennomført for lærere har det blitt observert en positiv effekt (Dağ, 2019, s.304-308; Rouhani et al., 2020, s.54-56). Gruppene rapporterte om økt selvtillit for egen kompetanse og undervisning, og økt motivasjon, som en følge av undervisningen. Kvaliteten på kurset er også viktig, da dårlige kurs kan føre til negative resultater i form av mindre motivasjon eller interesse (Woodrow, 1992, s.216-217).

Utdanningene bør gå over tid, da kurs over lengre perioder viser større positive endringer i resultatene (Menekse, 2015, s.22). Dette er noe av problemet, da det tidligere har blitt nevnt hvordan lærernes hverdag allerede er presset for tid. Dette gjør det også ekstra vanskelig å drive etterutdanning av alle lærerne som i dag er i arbeid. Her blir problemet både å få tilstrekkelig med tid til etterutdanningen, men også de manglende ressursene som er nødvendig for den omfattende videreutdanning som kreves, spesielt om man setter et mål om å gi alle formell kompetanse (Kjällander et al., 2018, s.25-28). Om universiteter og andre høyere utdanningsinstitusjoner legger til rette for et godt samarbeid, kan det gjøre det enklere å få gjennomført utdanningen ifølge en tidsplan som fungerer også for skolene. Det mangelfulle samarbeidet er en av de store utfordringene som partene må løse for å sikre lærerne gode muligheter for profesjonell utvikling (Menekse, 2015, s.22).

3.0 Metode

I dette kapittelet tas det utgangspunkt i problemstillingen “Hvordan har matematikklæreres utvikling i relasjon til programmering foregått etter fagfornyelsen, og hvilken effekt har det hatt?” og hvordan forskningen har blitt gjennomført for å svare på problemstillingen. I dette kapittelet presenteres først informasjon om den vitenskapsteoretiske tilnærmingen som er tatt for denne studien, og forskningsdesign og valget av metode. Etter det presenteres datainnsamlingen som ble gjort, og kvalitet i forskning.

3.1 Vitenskapsteoretiske betraktninger

For denne studien ble det vurdert å starte fra et fenomenologisk eller et mixed-method vitenskapsteoretisk ståsted. Der en fenomenologisk tilnærming ville fokusert på hvordan innføringen opplevdes fra lærernes perspektiv, og beskrevet lærernes opplevelser og erfaringer knyttet til fenomenet (Postholm, 2011, s.41-43). En mixed-method tilnærming ville kombinert to forskjellige vitenskapsteoretiske tilnærminger, en kvalitativ og en kvantitativ, og gjennomført forskning basert på elementer fra begge to. Med data fra to forskjellige tilnærminger kan man potensielt få et bedre overblikk fra den kvalitative tilnærmingen, og en bedre forståelse fra den kvalitative (Siraj, 2013, s.8-9).

3.1.1 Postpositivistisk

Med utgangspunkt i spissformuleringen av problemstillingen og forskningsspørsmålene har det derimot blitt tydelig at en postpositivistisk tilnærming til forskningen er mer passende. I likhet med tanken om “the scientific method” tenker man fra et postpositivistisk at bak alle resultater finnes det faktorer som kan ha påvirkning for det endelige resultatet. Altså at det kan være en sammenheng mellom enkelte faktorer, og hvordan det endelige resultatet ser ut. Det er dette som ligger til grunn for tanken bak postpositivistisk forskning, da man ønsker å identifisere og vurdere de faktorene som kan ha påvirkning for et resultat eller en hendelse. Den mest klassiske måten å gjøre slik forskning på er via et “eksperiment” der man har kan endre spesifikke faktorer for å se på eventuelle endringer i resultatet. I postpositivistisk forskning prøver man altså å redusere situasjoner til et gitt antall bestemte variabler, slik at det er mulig å forske på de forskjellige variablenes påvirkning på resultatet. Dersom vi kan redusere en situasjon til et begrenset antall muligheter, er det mulig å endre, manipulere, eller sammenligne situasjoner for de variablene, og dermed kunne registrere og observere eventuelle endringer i resultat (Creswell & Creswell, 2018, s. 44).

I Creswell og Creswell beskriver de 5 standpunkt for postpositivistisk tenkning:

1. Kunnskap kan aldri være absolutt, og forskning er feilbarlig. Derfor kan man ikke bekrefte en hypotese, men det kan være at det ikke er mulig å forkaste en nullhypotese.
2. Forskning er en prosess der man lager påstander, og deretter ønsker å justere, eller avkrefte dem. Målet er så å komme frem til sterkere påstander som er vanskeligere å avkrefte.
3. Data, beviser og rasjonelle slutninger er det som former kunnskap. Forskeren samler inn informasjon fra måleinstrumenter, fra deltakere, eller via gjennomførte observasjoner. På bakgrunn av denne dataen ønsker altså forskeren å trekke rasjonelle slutninger som kan forme ny kunnskap.
4. Forskning ønsker å utvikle sanne utsagn som kan forklare situasjoner eller årsakssammenhenger. For kvantitativ forskning kan det være å finne sammenhenger mellom bestemte variabler og resulterende hendelser.
5. Å være mest mulig objektiv er en viktig del av forskningen. Forskere må prøve å identifisere potensielle predisposisjoner de selv sitter på, og som kan oppstå ved valg av forskningsdesign.

(Creswell & Creswell, 2018, s. 45)

3.2 Forskningsdesign og metode

I arbeidet med denne studien ble flere forskjellige forskningsmetoder vurdert. Basert på problemstillingen og forskningsspørsmålene ble det til slutt valgt å gå for spørreundersøkelse. Her vil valget av metode, fordeler, og ulemper ved spørreundersøkelse kontra andre forskningsmetoder, bli diskutert og begrunnet som valg for denne studien.

Valget av datainnsamlingsmetode skal og burde være bestemt av hvordan problemstilling man har, og hva man ønsker å undersøke. Valget bør vurderes, og man burde kunne begrunne valget av forskningsmetode. Dersom man ønsker å undersøke meninger og trender hos en større gruppe mennesker, kan spørreundersøkelse være aktuelt da å intervju en stor gruppe som ofte vil være altfor tidkrevende. Om man vil undersøke atferd eller interaksjoner hos mennesker, vil observasjoner eller etnografiske studier være mer relevant. Dersom problemstillingen omhandler personers opplevelse som deltakere i en situasjon kan intervju være den som er mest relevant (Brinkmann & Kvale, 2021, s. 20-22). En av avveiningene man gjør før man gjennomfører en studie er valget av hva eller hvem som skal studeres. For en intervjustudie vil

et viktig element å ha med i utvalgsprosessen være å sørge for at alle deltakerne har opplevelser rundt fenomenet man ønsker å undersøke, og for en studie som bruker spørreundersøkelse vil måten man samler respondenter være en viktig faktor å vurdere (Postholm, 2011, s. 43-44).

Første steg i designingen av metoden for forskningen var å identifisere hva målet for studien var, hvilke spørsmål og faktorer skal undersøkes. For denne undersøkelsen var målet å undersøke hvordan effekt utviklingsarbeidet eller andre faktorer har hatt for lærernes profesjonelle utvikling, og dermed er det nyttig å se på trender i større datasett. Først ble både kvalitative og kvantitative tilnærminger vurdert, men basert på problemstillingen, ble spørreundersøkelse trukket frem som en mulig forskningsmetode, da man på denne måten kan ha mulighet for å oppdage de større sammenhengene. Hadde målet med undersøkelsen derimot vært å undersøke læreres egen opplevelse av utviklingsarbeidet, og hvor nyttig de selv erfarte at oppleggene var, ville intervju vært mer naturlig. Det hadde også vært rom for å gjennomføre en mixed-method studie der både spørreundersøkelse og intervju ble brukt som forskningsmetode, og da kunne sett på trendene, og undersøkt læreres egne erfaringer.

3.2.1 Kvalitativ og kvantitativ metode

I samfunnsvitenskap ønsker man å koble empiri sammen med ideer, teorier, eller hypoteser. Dette gjelder for både kvalitative og kvantitative studier. Men metoden vi bruker for datainnsamling gir oss forskjellige datasett, som kan benyttes på forskjellige måter innen forskningen. Det er tre hovedforskjeller som skiller kvalitativ og kvantitativ forskning fra hverandre.

Først ut er bruken av tid. For kvantitative studier er det å velge variabler, definere hvordan variablene skal måles, og å finne ut hvordan man skal kategorisere og vurdere resultatene noe som i stor grad skjer før datainnsamlingen. I kvalitative studier derimot skjer mye av kodingen av svar, og vurderinger av målinger enten underveis i innsamlingen eller etter innsamlingen (Neuman, 2014, s. 204-205).

Enda en forskjell er typen empiri som samles inn. I kvantitative studier samles det inn empiri i form av tall og målbare enheter. For eksempel kan en spørreundersøkelse være et kvantitativt verktøy som resulterer i ren data i form av tall eller definerte variabler. Dette standardiserer og gjør det mulig å representere empirien på en konkret og forståelig måte. Slik standardisert data er det også mulig å kjøre analyser og sammenligninger med. For kvalitative studier kan man

også få empiri i form av tall, men det er vanligere med tekst, handlinger, lyder, bilder eller andre mer fyldige datapunkter. Intervju er en vanlig metode å bruke innenfor kvalitative studier, da søker man å oppnå en forståelse for en situasjon eller opplevelse. Da er det mindre nyttig med standardiserte tall, mens utfyllende beskrivelser og forklaringer kan være nyttig empiri. Kvalitative studier ønsker altså ikke å standardisere empirien, men ønsker å beholde variasjonen og egenartene ved empirien (Neuman, 2014, s. 204-205).

Sist er hvordan vi kobler konsepter opp mot empirien. Som i bruk av tid gjør kvalitative studier mye av arbeidet tidlig i prosessen. Måleinstrumenter for å standardisere alt fra harde data til abstrakte ideer blir utformet tidlig, slik at innsamlet empiri kan måles og sorteres med en gang. Selv om endringer kan skje underveis er det meste av denne typen arbeid gjort i forkant av forskningen innenfor kvantitativ forskning. For kvalitativ forskning gjør man også ulike vurderinger i forkant av datainnsamlingen, men mange av variablene og målingene blir til underveis i innsamlingen eller i analysen. Kvalitative forskere har større muligheter til å vurdere og eventuelt gjøre endringer underveis i forskningsarbeidet. Kvalitativ datainnsamling er altså en mer kontinuerlig prosess (Neuman, 2014, s. 204-205).

3.3 Gjennomføringen av datainnsamlingen

Utgangspunktet for valget av metode ligger i denne problemstillingen med tilhørende forskningsspørsmål:

- Hvordan har matematikklæreres utvikling i relasjon til programmering foregått etter fagfornyelsen, og hvilke faktorer kan ha hatt påvirkning?
 - Hvordan har det blitt lagt til rette for lærernes profesjonelle utvikling etter fagfornyelsen?
 - Hvordan har matematikk-læreres situasjon utviklet seg etter fagfornyelsen?

Med bakgrunn i spissingen av problemstillingen, nødvendig tid og arbeidsmengden som kreves for å benytte flere forskningsmetoder, ble det valgt spørreundersøkelse som forskningsmetode for denne studien.

3.3.1 Spørreundersøkelse som forskningsmetode

Spørreundersøkelser er blant en av de mest brukte forskningsverktøyene. De kan brukes til å samle store mengder målbare og analyserbar empiri, og føyer seg altså inn i en positivistisk tilnærming. Man kan via spørreundersøkelser samle empiri om alt fra oppførsel og egenskaper,

til meninger og kunnskap. Ideen er at dersom vi har et stort antall personer, og samler empiri knyttet til de samme variablene fra alle, vil vi kunne kjøre sammenligninger basert på de forskjellige variablene. Da er ønske å kunne analysere empirien nærmere, og se om det kan være en sammenheng mellom de forskjellige man eventuelt finner. Ved bruk av spørreundersøkelser kan man si at man ser etter sammenhenger og ikke årsakssammenhenger, da man ikke kan endre variabler underveis og se etter årsakssammenheng. Dette er derimot ting man prøver å ta høyde for når man designer undersøkelsen (Neuman, 2014, s. 360-361).

Ved bruk av spørreundersøkelser er det vanlig med strukturerte spørsmål med tydelige og eksklusive svaralternativer. Denne typen utforming av svaralternativene vil gi målbar empiri som er mulige å analysere (Dalland, 2012, s.112). Her kommer en stor del av arbeidet i forkant av datainnsamlingen, da utformingen av spørreundersøkelsen krever tid og planlegging. Man må ta hensyn til hvilke datapunkter og spørsmål som kan være relevant for problemstillingen. Spørsmålene må skriftliggjøres før undersøkelsen sendes ut

Potensielle feilkilder knyttet til undersøkelsen kan ofte ha en sammenheng med forarbeidet. Eksempler på feilkilder kan være utvelgelsen av respondenter, og feil knyttet til besvarelse av enkeltspørsmål. Man kan risikere at en gruppe svarer sjeldnere på undersøkelsen, men en annen er overrepresentert som kan skape et feil bilde dersom det ikke tas hensyn til i forkant i form av kontrollvariabler. Enkeltspørsmål kan også være dårlig formulert, eller være åpne for tolkning, noe som kan resultere i at respondenter svarer på forskjellige ting i samme spørsmål. Man bør unngå ha spørsmål som omhandler mer enn et tema, da ved spørsmål som “Har dere epler og pærer i kantina?”. En respondent kan være i en situasjon der de kun har epler, eller kun har pærer i kantina, men vil da risikere å ikke kunne gi korrekte svar, og man ender opp med dårligere data (Neuman, 2014, s. 322-325). Andre elementer som organisering av spørsmålene, og rekkefølgen de stilles i kan påvirke resultatet, og må tas med i betraktningen under utformingen av undersøkelsen (Leavy, 2017, s.107). For å sikre at spørreundersøkelsen blir forstått og gir de resultatene man er ute etter, kan man lage en “pilotundersøkelse” der man sender et utkast av undersøkelsen til en liten gruppe mennesker. Denne pilot-undersøkelsen ønsker man skal være et grunnlag for tilbakemeldinger på både formuleringen av spørsmålene, og på innholdet i undersøkelsen. Ved å få andres perspektiver på det, kan man oppnå en mer forståelig og dekkende undersøkelse (Leavy, 2017, s.116).

For prosessen med utformingen av selve spørreskjemaet var det naturlig å benytte en digital løsning, for å enkelt kunne nå ut til flest mulig på kortest mulig tid. For det ble nettskjema.no brukt, som er Nord Universitets foretrukne verktøy for elektroniske spørreskjemaet. Det er flere fordeler med bruk av elektroniske spørreskjema over for eksempel spørreskjemaer sendt i posten. De største utfordringene for spørreskjemaer sendt i posten er lav svarprosent, og at det er en tidkrevende prosess. En slik undersøkelse har innebygd treghet i systemet da svarene går gjennom posten. Man har heller ingen kontroll over hvilken situasjon deltakerne er i når de svarer på undersøkelsen. En lærer som svarer i rolige omgivelser på eget kontor vil kanskje kunne gi bedre data enn en lærer som for eksempel svarer på undersøkelsen i morgenstresst rett før undervisning.

Elektroniske spørreskjemaer er derimot betydelig raskere, både i distribuering av undersøkelsen og i mottagelse av svar. Svarprosessen er også enklere og krever mindre egeninnsats fra deltakerne, da alt gjøres på en datamaskin, og skjemaet kan automatisk sendes inn når den er ferdig utfylt. Det at elektroniske spørreskjemaer er enkle å lage, er både en fordel og en ulempe. Fordelen er at utforming av spørreskjemaer går fort, og prosessen er godt tilgjengelig for de fleste. Ulemper kan være at nettopp fordi utformingen kan skje så fort, så er det enklere at noe kan formes litt for fort, og at kvaliteten kan bli litt lavere. For utvalg av et større antall deltakere i en slik masterstudie kan det også være vanskeligere med elektroniske spørreskjemaer å kunne gjøre et bevisst utvalg av deltakere, og å dermed få et representativt utvalg (Neuman, 2014, s. 348).

For å undersøke læreres profesjonelle utvikling ble det tatt utgangspunkt i tre hovedvariabler som skulle undersøkes. Dette var motivasjon for undervisning i programmering, selvtillit for undervisning i programmering, og kompetanse i programmering. Det var også ønskelig å kunne sammenligne disse verdiene mellom før og etter utviklingsarbeid skolene har gjort i sammenheng med innføringen av programmering fra høsten 2020. Spørsmålene i undersøkelsen ble altså formulert slik at de krevde to svar, et for før-verdien, og et for etter-verdien. For å undersøke disse variablene var det viktig å finne uavhengige variabler som kan ha hatt påvirkning på lærernes utvikling.

Noen av de uavhengige variablene som spørres etter i undersøkelsen er:

- Hvilket skoletrinn læreren underviser ved
- Størrelse på skolen
- Tidligere kompetanse i programmering
- Hvilke programmeringsspråk/verktøy som brukes ved skolen
- Hvilke fag læreren underviser i
- Hvilke typer utviklingsarbeid læreren har deltatt i

Tanken bak disse uavhengige variablene var de representerer noen av faktorene som kan ha påvirket lærerens utvikling. Hvilket skoletrinn kan ha påvirkning i at lærere ved høyere trinn vil kunne ha behov for bedre kompetanse i programmering, og det er mulig at disse lærerne har blitt prioritert eller vært fokuset på kompetansehevingen. Størrelse på skolen definerer også størrelsen på lærerfellesskap, og rammene for kompetanseutviklingen som kan skje innad i lærerfellesskapet ved den skolen. Ved å få inn empiri om disse variablene, er det mulig å se utviklingen for de tre hovedvariablene delt inn i grupper basert på disse uavhengige variablene. Man kan altså se utviklingen i motivasjon for undervisning i programmering for gruppene lærere som underviser på ungdomstrinnet mot lærere som underviser på mellomtrinnet. På denne måten kan man beskrive utviklingen for de forskjellige gruppene, og man kan sammenligne ulike typer utviklingsarbeid, størrelse på skoler, skoletrinn, og mer.

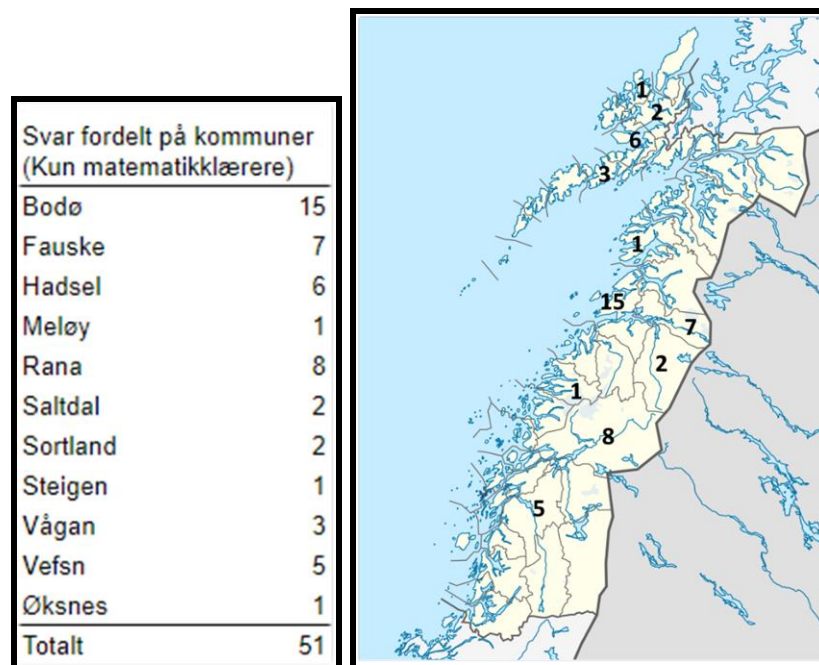
3.3.2 Utvalg og deltakere

For utvelgelsen av deltakere til denne studien måtte det først identifiseres hvem som skulle undersøkes. Problemstillingen tar for seg matematikklærere, og på bakgrunn av begrensninger kort beskrevet i kapittel 1.2 er det spesifikt lærere som underviser i matematikk, på grunnskolenivå, og som underviser i Nordland fylke. Disse avgrensningene er de eneste begrensningene som ble lagt til grunn i henvendelsen til mulige deltakere, og avgrensningen gjør at det er et lavere behov for antall svar enn dersom man skulle gjennomført en studie på landsbasis. Da spørreskjemaet skulle sendes ut elektronisk til så mange lærere som mulig ble det naturlig å henvende seg til administrasjonen ved skolene, da administrasjonen ved skolene ville ha muligheten til å sende ut undersøkelsen til lærerne ved deres skole. Link til spørreundersøkelsen ble sendt ut i en epost sammen med informasjon om hva undersøkelsen gjaldt, og en kort beskrivelse av hvilke lærere som det var ønsket med svar fra.

Alstadhaug	Andøy	Bodø	Brønnøy	Fauske	Hadsel	Hemnes	Meløy
2	2	9	4	4	4	6	5
Narvik	Rana	Saltdal	Sortland	Steigen	Vågan	Vefsn	Øksnes
3	9	2	2	1	4	5	2

Tabell 1: Antall skoler spørreundersøkelsen ble sendt til

Undersøkelsen ble de sendt ut til 63 skoler fordelt over hele Nordland. I utsendingen ble det lite mulighet til å selv velge hvilke lærere som svarte, og utvelgelsen ble da heller indirekte gjort basert på hvilke lærere som tok seg tid til å svare. Dette er ikke en optimal måte å gjøre et utvalg på, men med begrenset tid og tilgang på individuelle lærere ble denne løsningen benyttet.



Tabell 2: Antall svar fra lærere i kommuner (Til venstre)

Figur 3: Oversikt over svar fra kommunene (NordNordWest, 2012) CC BY-SA 3.0 / Med antall svar lagt til (Til høyre)

Etter første utsendingen av skjemaet ble det også sendt ut to nye eposter som en påminnelse, og med et mål om å øke svarprosenten. Dette ble gjort parallelt med annet arbeid, og undersøkelsen stod åpen over en lengre periode. Omtrent to måneder etter første utsending ble undersøkelsen lukket, med et endelig antall svar fra 56 individuelle lærere. Blant de 56 svarene hadde derimot 5 av lærerne ikke markert for at de underviser i matematikk, og de blir derfor utelukket fra studien.

3.3.3 Analysering av spørreundersøkelse

For analyseringen av datamaterialet fra spørreundersøkelsen brukes SPSS (Statistical package for the social sciences, eller Statistisk pakke for samfunnsvitenskapen) (IBM, n.d.). Datasettet fra Nettskjema.no ble lastet ned etter spørreundersøkelsen ble lukket, og det ble åpnet i Excel og SPSS for rydding i datasettet. Det kan involvere å konstruere nye bredere variabler, dersom noen underviser i programmering i matematikk og/eller i naturfag kan man lage en variabel som er “underviser i realfag”, der de som underviser i matematikk eller naturfag begge markeres positiv. Man kan derimot ikke lage tynnere kategorier, da man ikke har den nødvendige informasjonen til det (Neuman, 2014, s. 397).

Det kan også innebære å se etter feil utfylling og korrigere så langt man har mulighet til det. I denne studien blir lærerne spurt om hvilke typer utviklingsarbeid de har deltatt i, og også hvor nyttige de opplevde de forskjellige typene utviklingsarbeid. Dersom noen har markert at “mentor-ordning” var en nyttig form for utviklingsarbeid, men ikke har markert at de har deltatt i en mentor-ordning, er det en mulig feil i datasettet. I denne studien har da de mer spesifikke svarene blitt prioritert. Dersom de har en mening på hvor nyttig et opplegg var, kan man på et tryggere grunnlag gå ut ifra at de har deltatt. Mens dersom det oppgis at de har deltatt har vi ingen måte å vite hvor nyttig de opplevde opplegget når det ikke er markert for. Det vil med andre ord si at dersom det er markert at et opplegg var nyttig, men det er ikke markert for deltakelse i opplegget, er det omgjort til at det er markert for deltakelse. Dersom det er motsatt, at det er markert at de har deltatt i et opplegg, men ikke for hvor nyttig de opplevde det, er det omgjort til ikke deltakelse.

Deskriptiv statistikk

For analysearbeidet vil det først være presentasjon av deskriptiv statistikk slik at man kan få et innsyn i hvordan datamaterialet ser ut, og hva man kan lese ut av det. Det innebærer å blant annet å representere data i tabeller eller grafer, og å fremheve sentralmålene (Johnson & VanderStoep, 2009, s. 92-93). Det er også lurt å se på hvem som har besvart undersøkelsen og hvordan utvalget stemmer overens med den større populasjonen med lærere i Norge. Det kan være nyttig å starte med å beskrive hvilke typer utviklingsarbeid som har vært mest vanlig, hvordan situasjonen er i forhold til undervisningsmateriale, eller å få en oversikt over i hvor mange fag lærere underviser i programmering. Slik deskriptiv statistikk bidrar til å danne et bilde av hvordan empirien er (Neuman, 2014, s. 396).

Analytisk statistikk

Etter å ha presentert empirien i form av deskriptiv statistikk kan man bruke SPSS sine analytiske verktøy til å se etter statistisk signifikante sammenhenger via ulike statistiske tester. Om noe er statistisk signifikant er det sannsynlig at variasjonene i resultatene skyldes faktiske forskjeller, og ikke tilfeldige variasjoner. Dersom man måler 50 tilfeldige menn og 50 tilfeldige kvinner vil man sannsynligvis finne en statistisk signifikant forskjell mellom høydene for gruppene, altså er det sannsynlig at det er en faktisk forskjell i variabelen høyde for gruppene menn og kvinner. Fra disse analysene kan man derimot bare lese sannsynligheter for sammenheng, og en test kan ikke garantere at det er en.

Når man gjennomfører en slik test tar man utgangspunkt i to hypoteser, der den ene er en nullhypotese man prøver å avkrefte. En nullhypotese er en hypotese som forutser at det er ingen sammenheng mellom variablene. Den andre hypotesen kalles en alternativ hypotese. Den kan da formuleres som at det er en sammenheng mellom variablene. I en slik hypotesetest bruker man empirien til å undersøke om man kan forkaste eller om man må beholde nullhypotesen. Forkaster man nullhypotesen har man ikke bevist at den alternative hypotesen er sann, men man kan si at det er statistisk sannsynlig at det er en sammenheng (Neuman, 2014, s. 185). I en slik test er det vanlig å sette signifikansnivået til 0,05, eller 5%. Dersom resultatet av en slik test havner på eller under 0,05 betyr det at i kun 5 av 100 tilfeller skyldes variasjonene i variablene ren tilfeldighet. Med andre ord er det 95% sannsynlighet for at resultatene ikke er tilfeldige, men representerer populasjonen korrekt (Neuman, 2014, s. 423).

3.4 Kvaliteten i forskning

Kvaliteten i forskning handler om hvor pålitelige og gyldige resultatene fra forskningen er, og kvaliteten uttrykkes i form av reliabilitet og validitet.

3.4.1 Reliabilitet

Reliabilitet handler om at forskningsmetoden skal gi svar som er konsekvente. Et verktøy som produserer forskjellige resultater med samme utgangspunkt vil gi studien en lavere reliabilitet. En badevekt som viser vilt forskjellige målinger dersom du måler deg tre ganger på rad har liten reliabilitet. Reliabilitet handler derimot ikke om hvor korrekte målingene er, dersom badevekten gir samme feil resultat hver gang, har den allikevel høy reliabilitet. Dette handler om reliabilitet i målingsverktøyet, og handler om å få samme resultat med samme test (Neuman, 2014, s. 212).

En test med høy reliabilitet bør kunne gjentas med samme gruppe mennesker på et annet tidspunkt, og få like resultater. Høyere likhet i resultatene, vil bety at testen har høyere test-retest reliabilitet. Å kunne fått gjennomført en undersøkelse før fagfornyelsen ble innført, og så kunne gjennomført en ny undersøkelse nå i ettertid hadde vært nyttigere, men da det er to år etter innføring blir dette vanskelig. Det kan føre til litt lavere reliabilitet da deltakerne er nødt til å selv prøve å huske hvordan situasjonen var for to år siden. Det kan bety at dersom du hadde gjennomført samme test med samme deltakere en annen dag, er det mulig at de husker andre ting, eller legger vekt på andre minner.

Dersom faktorer som henger sammen viser lignende resultat bidrar det til parallel-forms reliabilitet. I undersøkelsen for denne studien handler et av spørsmålene om hvilke fag læreren bruker programmering i, og et annet om hvor ofte læreren bruker programmering i de ulike fagene. Dersom mange svarer at de bruker programmering i naturfag, men flere av de samme markerer at det er "ikke relevant" om hvor ofte de bruker programmering i naturfag, har vi to faktorer som ikke henger sammen. Dette kan bidra til lavere reliabilitet innad i testen, dersom målinger som burde vise sammenheng ikke gjør det.

Inter-rater reliabilitet handler om hvorvidt andre som ser samme empiri vil tolke funnene likt. Dersom en test har en høy inter-rater reliabilitet burde en annen forsker kunne bruke samme test, og få like resultater. Dersom du har to personer som bytter på å se gjennom samme kikkert i samme posisjon, men ser forskjellige ting, så har kikkerten lav inter-rater reliabilitet (Johnson & VanderStoep, 2009, s. 64-65).

3.4.2 Validitet

Validitet handler om å avgjøre hvorvidt det man måler er sant. Blant annet vil det gå ut på å se etter momenter som avkrefter funnene, å være kritisk til egne funn og tolkninger, for så å kunne vurdere om det man har funnet ut faktisk er sant. En test har høy validitet dersom den faktisk måler det den er ment å måle, og den måler det rett.

Content validitet handler om hvorvidt de elementene som måles via testen representerer konseptet som skal undersøkes godt nok. Dersom denne studien ønsket å undersøke læreres profesjonelle utvikling, men spørsmålene kun handlet om hvilket utstyr de hadde tilgjengelig på skolen, ville testen hatt lav content validitet. For at testen skal kunne dekke de elementene

som er en del av “profesjonell utvikling” er det nødvendig å definere og tilpasse testen til å måle det som er relevant. I denne oppgaven har utviklingen innenfor kompetanse, motivasjon, og selvtillit for undervisning vær de elementene som måles for å dekke “profesjonell utvikling”, og med spørsmål som dekker de variablene vil testen ha høyere content validitet. Dersom man kun hadde spørsmål relatert til kompetanse ville man også kunne argumentere for at det oppfyller profesjonell utvikling, selv om man måler færre områder for profesjonell utvikling (Johnson & VanderStoep, 2009, s. 59-60).

Hvorvidt en test klarer å få en korrekt måling av det den ønsker å måle er construct validitet. Dersom spørreundersøkelsen klarer å få korrekte målinger om læreres kompetanse, motivasjon og selvtillit vil den ha høy construct validitet. Da vil altså lærere som har hatt høy grad av profesjonell utvikling få gode målinger innenfor flere av de valgte elementene. Ved å sørge for å ha både god content og construct validitet sørger man for at man måler det man ønsker, og at målingene er korrekte (Johnson & VanderStoep, 2009, s. 59-60).

3.4.3 Generaliserbarhet

Generaliserbarhet handler om hvor generelle resultatene kan sies å være. Dersom man kan ta resultatene fra studien og overføre det til andre situasjoner eller grupper og resultatene stemmer fortsatt, vil testen være mer generaliserbar. Et representativt utvalg for en definert gruppe vil bidra til å gjøre resultatene generaliserbare for den gruppen (Brinkmann & Kvale, 2021, s. 355). I denne studien har svarene en ganske lik spredning mellom de forskjellige aldersgruppene som spredningen generelt blant lærere i Norge. Kjønnbalansen blant svarene er også nært kjønnbalansen blant lærere generelt, som også øker generaliserbarheten til studien. Dersom testen derimot hadde tatt for seg gutter i Nordland, er det vanskelig å tenke at resultatene kan generaliseres til kvinner i Oslo, og det kan derfor være vanskeligere å generalisere slik.

3.5 Etikk

I all samfunnsvitenskap er det viktig at det tas hensyn til personene som involveres i forskningen. En av de viktigste momentene angående etiske overveielser før datainnsamlingen for både kvalitativ og kvantitativ forskning, vil være at informantene er så informerte og klar over hva prosjektet går ut på, at de kan gi et informert samtykke. Dette betyr at de vet akkurat hva de takker ja til, og hva de begir seg ut på (Postholm, 2011, s. 145-147). Det er altså nødvendig at forskerne informerer deltakerne tilstrekkelig over hva deltakelsen innebærer.

Elementer deltakerne bør informeres om for å kunne oppfylle kravene for et informert samtykke inkluderer formålet med forskningen, risikoer og potensielle positive resultater fra forskningen, og at deltakerne kan trekke seg fra forskningen underveis. Det bør også legges til rette for at deltakerne skal kunne stille spørsmål dersom behovet skulle oppstå. Det er viktig at deltakernes samtykke er helt frivillig, og skal ikke være påvirket av noen form for tvang, eller ufrivillighet (Oldendick, 2012, s. 24).

I innsamlingen av empiri fra deltakerne er det også etiske betraktninger man bør ta hensyn til. For at deltakerne skal kunne gi et informert samtykke krever det at de er selvstendige aktører som er i stand til å ta egne vurderte valg. Siden denne studien ikke henvender seg spesifikt til for eksempel barn, er dette et mindre viktig element i innsamlingen av empiri, men er noe man bør ha i mente.

De mer alvorlige etiske vurderingene er mer relevant for kvalitative studier, der menneskene er kjent for forskerne, og kan i verste tilfelle identifiseres i studien. Men selv i en anonym spørreundersøkelse er de etiske vurderingen viktig å ta hensyn til, da flere av dem er gjeldende selv om deltakerne ikke kan identifiseres. Blant annet vil kravet om at deltakerne skal kunne gi et informert samtykke fortsatt være høyst relevant. Da vil det i informasjonen som gir før deres deltakelse være viktig å inkludere elementene som kan være relevant selv i en anonym spørreundersøkelse. Det kan være å informeres om eventuelle risikoer ved deltakelse, som kan inkludere spørsmål om såre eller potensielt traumatiske hendelser. Det kan også være å spesifisere at all dataen er automatisk anonymisert, og kan derfor ikke spores tilbake til deltakerne slik at sosiale eller rettslige skader ikke er en risiko (Oldendick, 2012, s. 25).

I utformingen av spørreundersøkelsen til denne studien ble flere slike etiske vurderinger tatt i betraktning i utformingen av spørreskjemaet, og av innsamlingen av informasjonen i forkant. Det viktigste kom først i spørreskjemaet der ble det tydeliggjort hva spørreundersøkelsen gikk ut på, og hvilke kategorier med spørsmål lærerne kunne forvente i løpet av spørreundersøkelsen. Det ble informert om antall spørsmål hele spørreundersøkelsen inneholdt, og om cirka hvor lang tid undersøkelsen kom til å ta. Det ble også i introduksjonen til spørreundersøkelsen tydeliggjort at undersøkelsen er frivillig, og at før svaret sendes inn kan de når som helst trekke seg. Det ble informert om at på tross av at svarene er anonyme og det ikke er mulig å spore svarene til individene, vil det være noe begrensede personlige faktorer som de ville bli spurt om. En epost for eventuelle spørsmål ble vedlagt fremst i undersøkelsen.

4.0 Funn og analyse

I dette kapittelet skal datamaterialet presenteres. Først vil dataen om respondentene og deres situasjon i skolen presenteres, og så vil dataen om respondentenes erfaringer rundt utviklingsarbeid presenteres. Datamaterialet ble samlet inn via Nettskjema.no, og bearbeidet ved bruk av SPSS (Statistical Package for the Social Sciences) (IBM, n.d.).

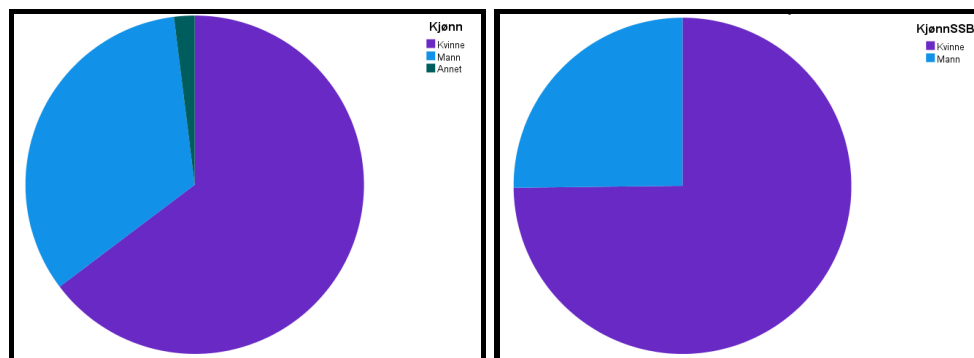
For alt av tabeller som presenteres i dette kapittelet vil de to høyeste prosentverdiene/andelene være **markert** for å være litt enklere å identifisere.

4.1 Respondentene

Først i spørreundersøkelsen er det noen spørsmål knyttet til å definere hvordan utvalget som har svart på undersøkelsen ser ut. Faktorer som kjønn, alder, og hvor stor skole de underviser ved.

4.1.1 Kjønnfordeling

For å sikre et representativt utvalg kan det være nyttig å se på enkle faktorer som beskriver lærerne i Norge som gruppe, og som kan påvirke resultatene fra studien. Her er kjønnfordelingen en av faktorene som kan ha en påvirkning, og som er verdt å sammenligne mellom respondentene og lærerne i Norge.



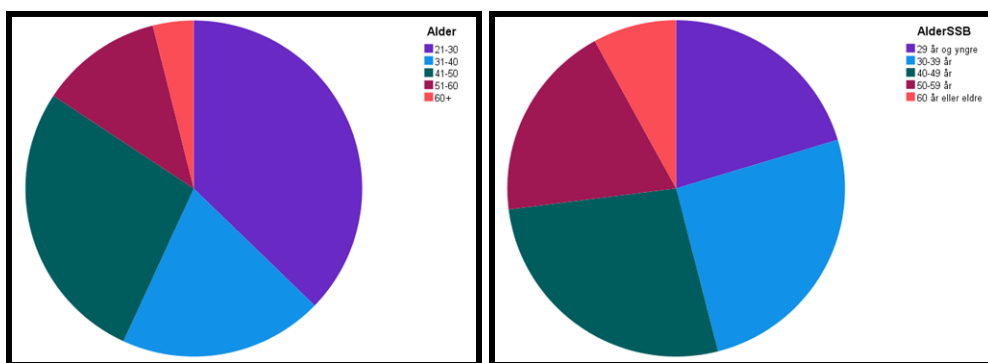
Figur 4: Kjønnfordeling blant respondentene på undersøkelsen (Til venstre)

Figur 5: Kjønnfordeling blant lærere, Norge 2020 (Statistisk sentralbyrå, 2021) (Til høyre)

For spørsmålet om kjønn identifiserer cirka 65% av respondentene seg som kvinner, 33% som menn, og 2% som noe annet. Denne kjønnfordelingen er ganske lik kjønnfordelingen blant lærere generelt i Norge. Menn er derimot litt sterkere representert i denne studien sett opp mot andelen menn blant lærerne generelt.

4.1.2 Aldersfordeling

En annen faktor som kan være relevant og som kan påvirke svarene er respondentenes alder. En sammenligning mellom aldersfordelingen til respondentene og lærerne i Norge kan også være viktig.



Figur 6: Aldersfordeling blant respondentene på undersøkelsen (Til venstre)

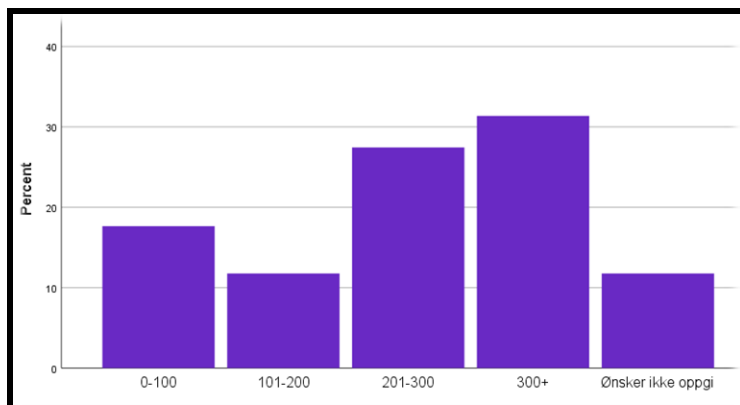
Figur 7: Aldersfordeling blant lærere, Norge 2020 (Statistisk sentralbyrå, 2021) (Til høyre)

For aldersfordelingen blant respondentene er slik at nesten 40% er under 30 år, og nesten 60% er under 40 år. Blant svarene er også kun 15% over 50 år. Dette skiller seg fra aldersfordelingen blant lærere generelt i Norge.

Blant respondentene er aldersgruppen 21-30 sterkt overrepresentert i forhold til gruppen 29 år og yngre, og gjør opp nesten en dobbelt så stor andel blant respondentene som blant lærerne generelt i landet. De eldre derimot er sterkt underrepresentert da de over 50 utgjør rundt 30% generelt i landet, men kun halvparten så stor andel blant respondentene. At undersøkelsen ble gjort elektronisk, kan ha vært med på å forme hvilke aldersgrupper som valgte å delta. Temaet for undersøkelsen kan også ha hatt en påvirkning da temaet er både nytt og stort, og kan dermed virke avskrekkende.

4.1.3 Skolestørrelse

Skolestørrelse påvirker også størrelsen på lærerkollegiet, og kan da ha en påvirkning på blant annet utviklingsarbeidet som blir gjennomført, og kan derfor være verdt å ha med i studien.

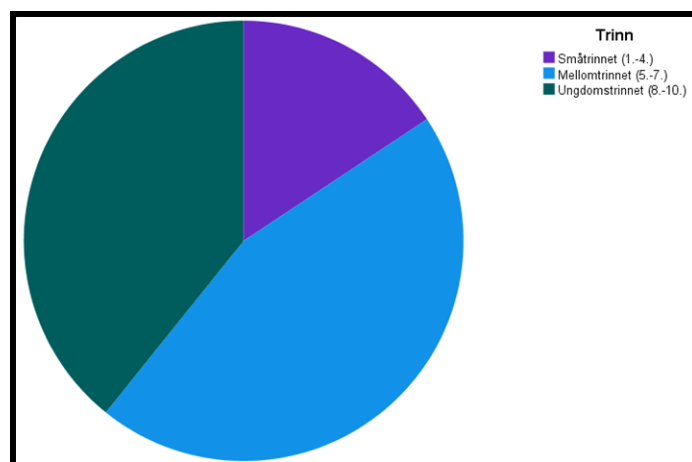


Figur 8: Respondentenes skolestørrelse målt i antall elever ved skolen

Majoriteten av respondentene underviser ved større skoler. Cirka 30% underviser ved skoler som har fra 201 til 300 elever, og cirka 30% underviser ved skoler som har flere enn 300 elever. Respondentene hadde muligheten til å ikke oppgi hvor stor skolen var, da det i for eksempel små kommuner kunne bidratt til å identifisere hvilken skole, og hvilken lærer som hadde besvart undersøkelsen. 6 personer, eller cirka 10% ønsket da ikke å oppgi antall elever ved skolen.

4.1.4 Fordeling over skoletrinn

Fordelingen mellom skoletrinnene lærerne underviser ved kan ha påvirkning i form av hvilken kompetanse, og på hvilket nivå som er nødvendig for lærerne.

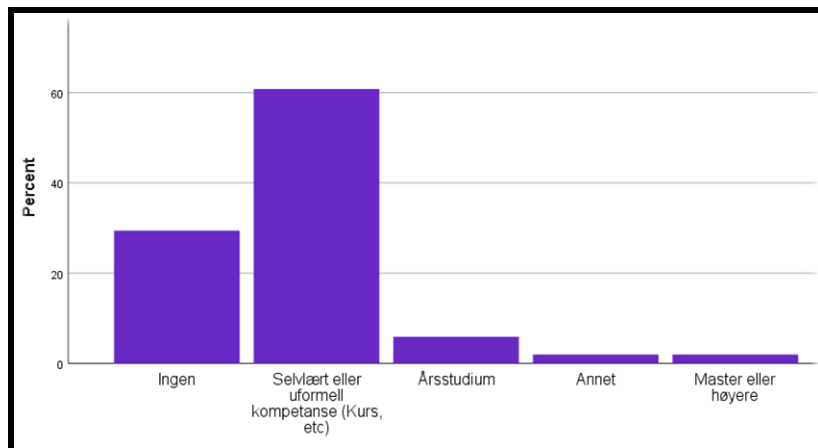


Figur 9: Skoletrinnene respondentene underviser ved

Fra forskjellige skoletrinn er det flest besvarelser fra lærere som underviser på mellomtrinnet, tett fulgt av lærere som underviser ved ungdomstrinnet. Det kan være både positivt og negativt for denne studien. Lærere ved høyere skoletrinn har høyere behov for fagkompetanse og kan dermed også ha høyere krav til kompetanseutvikling. Med en slik gruppe respondenter kan det altså hende at man får et tydeligere resultat for hvordan utviklingen har gått, og hvilket behov som finnes for videre utviklingsarbeid. Det betyr også at datamaterialet er skjevfordelt og vil sterkere representere erfaringene fra lærere ved høyere skoletrinn enn lærere fra lavere skoletrinn.

4.1.5 Respondentenes kompetanse i programmering

Lærernes tidligere kompetanse i programmering er med på å danne grunnlaget for den videre profesjonelle utviklingen de kan ha innenfor temaet, og er derfor en viktig å undersøke.



Figur 10: Respondentenes kompetanse i programmering

Blant respondentene er det klart vanligste å være selvlært, eller å ha uformell kompetanse fra kurs eller andre opplegg. Mens 60% er selvlært, er det derimot nesten 30% som oppfatter selv å ha ingen programmeringskompetanse. Kun 8% av respondentene har formell kompetanse, hvorav 6% har årstudium, og 2% har en mastergrad eller høyere. Blant de som har valgt “annet” har en tatt IKT ved lærerhøgskole, en annen har 15 studiepoeng i programmering fra NTNU, og en tredje holder på å ta 15 studiepoeng.

4.2 Programmering i fag

Da undersøkelsen tar for seg matematikklærere er det gitt at lærerne underviser i matematikk. Ved å undersøke hvor mange andre fag respondentene underviser i, er det mulig å se om det å undervise programmering i flere fag har noen effekt på utviklingen.

Programmering i fag	Prosent
Matematikk	100,0%
Naturfag	37,3%
Kunst og Håndverk	21,6%
Musikk	9,8%
Andre fag	21,6%

Tabell 3: Fag der respondentene benytter seg av programmering

Da målet for denne studien var å nå matematikklærere og å undersøke deres utvikling er det naturlig at respondentene underviser i matematikk. Naturfag, kunst og håndverk, og musikk er også fag der programmering har spesifikke mål på læreplanen. Det er derimot tydelig at det ikke nødvendigvis er de samme som underviser i matematikk som i naturfag, kunst og håndverk, eller musikk. I andre fag som oppgis der respondentene underviser i programmering er valgfag programmering, samfunnsfag, KRLE, engelsk, og norsk. Det er altså flere av respondentene som velger å bruke programmering også utenfor de fagene der de har spesifikke kompetansemål som elevene skal nå.

Programmering i fag	Småtrinnet (1.-4.)	Mellomtrinnet (5.-7.)	Ungdomstrinnet (8.-10.)
Matematikk	100,0%	100,0%	100,0%
Naturfag	12,5%	43,5%	40,0%
Kunst og Håndverk	12,5%	39,1%	5,0%
Musikk	0,0%	13,0%	10,0%
Andre fag	37,5%	17,4%	20,0%

Tabell 4: Fag der respondentene benytter seg av programmering sortert etter skoletrinn

Dersom man deler inn respondentenes svar etter skoletrinn ser det ut som at på småtrinnet er der respondentene bruker programmering mest i fag som ikke har spesifikke kompetansemål.

Da det på småtrinnet ofte er enkeltlærere som følger klassen i et større utvalg fag, kan det være at dersom du bruker programmering i matematikk, er det enklere å overføre det til de andre fagene du underviser i. Lavere bruk av programmering i naturfag blant respondentene på småtrinnet kan skyldes at naturfag ikke har spesifikke kompetansemål knyttet til programmering før etter 7. trinn. Musikk har ikke kompetansemål knyttet spesifikt til programmering før 10. trinn, selv om digitale verktøy er nevnt tidligere. Kunst og håndverk har ikke kompetansemål som nevner programmering eksplisitt, men nevner bruk av digitale verktøy etter 7. og 10. trinn (Kunnskapsdepartementet, 2019a; Kunnskapsdepartementet, 2019c; Kunnskapsdepartementet, 2019d). På mellomtrinnet har respondentene da tatt i bruk programmering i større grad i flere av fagene med kompetansemål, spesielt kunst og håndverk og naturfag har store økninger. De andre fagene uten spesifikke kompetansemål faller derimot bak og ser mindre bruk av programmering. På ungdomstrinnet faller kunst og håndverk av igjen, mens naturfag holder seg stabil. Disse tallene betyr derimot ikke at dette er graden av undervisning i programmering som skjer ved disse trinnene, men heller noe om hvor matematikklærerne underviser og benytter seg av programmering. Spesielt ved ungdomstrinnet kan det altså heller være et tegn på at lærerne i større grad forholder seg til sitt eller sine fag i flere klasser, heller enn å følge samme klassen i mange forskjellige fag.

4.3 Programmeringsspråk/programmeringsverktøy

For å avgjøre hva som bør satses på av kursing, er det relevant å se på hvilke språk/verktøy lærerne selv foretrekker, da det kan være mengde erfaring som påvirker preferansene.

Programmeringsspråk/ Programmeringsverktøy	Andel som liker språket/verktøyet
Python	11,8%
Micro:bit	54,9%
Scratch	43,1%
Minecraft Education Edition	37,3%
Annet	9,8%

Tabell 5: Respondentenes preferanser

Blant respondentene er det Micro:bit og Scratch som er de foretrukne programmeringsspråkene/verktøyene, med Minecraft education edition på en nær tredjeplass.

Under annet blir Lego EV3 eller lego-roboter nevnt av tre lærere. En lærer nevner code.org som et verktøy. En lærer nevner HTML. Micro:bit, Scratch, og Minecraft education edition har alle tre enklere grensesnitt enn Python, da man i alle tre kan bruke blokkprogrammering over tekstprogrammering, noe som kan være lettere tilgjengelig for lærere.

Programmeringsspråk/ Programmeringsverktøy	Andel som bruker i undervisningen
Python	7,8%
Micro:bit	51,0%
Scratch	47,1%
Minecraft Education Edition	39,2%
Annet	15,7%

Tabell 6: Respondentenes bruk i undervisningen

For bruk i undervisningen er tallene ganske like hva respondentene foretrekker. Det kan tyde på at lærerne enten selv velger hva de bruker i undervisningen, eller at de blir mer komfortable med de verktøyene de bruker. Under annet nevnes de samme tingene som på forrige spørsmål, i tillegg til roboten Bee-Bot og Lego WeDo. Bee-bot er en enkel robot som programmeres via knapper på toppen, med kommandoer som fremover, bakover, eller snu 90 grader til venstre eller høyre. Lego WeDo er en samling med legoklosser, og et programmeringsverktøy for PC eller nettbrett. Med disse bygger man en robot som kan programmeres via et blokkbasert grensesnitt.

Programmeringsspråk/ Programmeringsverktøy	Småtrinnet (1.-4.)	Mellomtrinnet (5.-7.)	Ungdomstrinnet (8.-10.)
Python	0,0%	4,3%	15,0%
Micro:bit	25,0%	52,2%	60,0%
Scratch	25,0%	52,2%	50,0%
Minecraft Education Edition	62,5%	52,2%	15,0%
Annet	37,5%	17,4%	5,0%

Tabell 7: Respondentenes bruk i undervisningen sortert etter skoletrinn

Hvis vi deler inn bruken basert på skoletrinn, kommer det frem at ved småtrinnet er det oftere mer aktuelt med verktøy som ikke har tekst-programmering som fokus. Enten ved at det er noe som “spillifiserer” programmeringen som Minecraft, eller andre mer fysiske verktøy som fokuserer mer på den algoritmiske tankegangen bak programmering, som Bee-Bot eller Lego Wedo. Når du kommer til mellomtrinnet brukes det i større grad verktøy som Scratch og Micro:bit. Micro:bit egner seg spesielt godt til undervisning på mellomtrinnet da det som verktøy kan brukes til å tilrettelegge for overgangen fra blokkbasert til tekstbasert programmering. Micro:bit har også fordelen med å ha et fysisk element til programmeringen, som kan bidra som et godt verktøy i elevenes arbeid. Python er nesten fraværende fra respondentenes bruk ved alle tre skoletrinnene, men ser litt bruk fra mellomtrinnet og oppover.

4.4 Undervisningsressurser

Undervisningsressurser som skolebøker, fysiske roboter, nettbaserte verktøy eller arbeidsrom kan ha en påvirkning både på lærerens utvikling, men også på hvordan undervisningen kan legges opp for elevene. Disse fire faktorene er det som tas med i begrepet “undervisningsressurser”.

Tilgjengelighet av undervisningsressurser	Skolebøker	Fysisk utstyr	Nettbaserte ressurser	Arbeidsrom
Ingen	25,5%	19,6%	2,0%	23,5%
Lav	41,2%	31,4%	31,4%	41,2%
Tilfredsstillende	21,6%	21,6%	35,3%	15,7%
God	7,8%	17,6%	23,5%	11,8%
Svært god	3,9%	9,8%	7,8%	7,8%

Tabell 8: Respondentenes tilgang på undervisningsressurser

Jevnt over har respondentene lav tilgang på det meste av undervisningsressurser. Skolebøker er fraværende hos 25% av respondentene, og over 40% opplever å ha lav tilgang. Kun litt under 35% opplever å ha tilfredsstillende eller bedre tilgang på relevante skolebøker for undervisning i programmering.

Situasjonen er noe bedre for tilgang på fysisk utstyr, der 20% opplever å ha ingen tilgang til fysisk undervisningsmateriale, og 30% har lav tilgang. Det betyr at cirka 50% opplever å minst

ha tilfredsstillende tilgang på fysisk undervisningsmateriale. En av respondentene skriver i et av fritekst-svarene at:

- “De har vært opptatt av programmering og iPad som læringsverktøy de siste 10+ årene. Nå er det ingen utvikling, eller kursing, men noen lærere som enda brenner for programmering som underviser på denne måten. Vi har derimot dårlige rammer som få/ingen rom tilgjengelig og lite utstyr (Micro:bit) som også er godt brukt/ødelagt. Vi har bestilt nytt, men korona hindrer tilførsel av deler som behøves i produksjonen.”

For noen er det altså en utfordring at verktøyene for undervisning ikke er til stede, og det problematiserer programmeringsundervisningen.

Nettbaserte ressurser er det respondentene opplever å ha best tilgang til, der nesten 70% opplever å ha tilfredsstillende tilgang. Blant fritekst-svarene fremheves code.org som noe flere av respondentene bruker aktivt. En av respondentene skriver derimot i et fritekstsvaer:

- “Da vi skulle gjennomføre programmering-minikurs i kollegiet ble det veldig tydelig at nettet på skolen ikke er dimensjonert for dette. Det blir derfor også håpløst å skulle gjøre dette i klasse med feks 20 elever hvor halvparten vil oppleve å ikke komme inn, det henger seg opp, og man dermed ikke får gjennomført opplegget man har planlagt”

Om infrastrukturen hos noen skoler ikke er dimensjonert for å håndtere at klasser skal drive undervisning fra for eksempel code.org vil dette sette en bremse både på undervisningen, men som respondenten skriver setter dette også en bremse for lærernes arbeid med programmering.

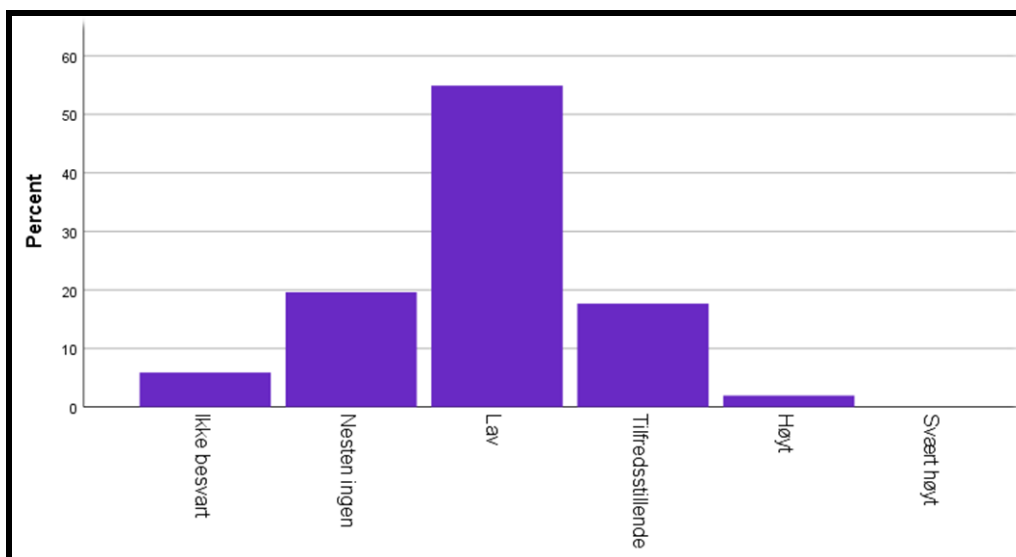
Tilgang på arbeidsrom er også mangelfull for de fleste av respondentene, der nesten 65% svarer at de har ingen eller lav tilgang på arbeidsrom for å drive programmeringsundervisning.

4.5 Utviklingsarbeid

Utviklingsarbeid er en viktig del av å ivareta yrkesutøveres profesjonelle utvikling. Dette gjelder også for lærere. Så en del av spørreskjemaet tar for seg forskjellige aspekter ved utviklingsarbeid gjennomført i skolen.

4.5.1 Grad av gjennomført utviklingsarbeid

Lærernes opplevelse av mengden utviklingsarbeid som har blitt gjennomført kan bidra til å illustrere hvor stor satsing det har vært på utviklingsarbeid i skolene, noe som er en viktig vurdering for å kunne si noe om eventuelle grep videre.



Figur 11: Respondentenes opplevelse av mengden utviklingsarbeid som har blitt gjennomført knyttet til LK20

Jevnt over opplever respondentene at det har vært en lav grad av gjennomføring av utviklingsarbeid knyttet til nytt læreplanverk. Nesten 75% opplever at det har vært gjennomført lite eller nesten ingenting av utviklingsarbeid knyttet til nytt læreplanverk.

I hvilken grad utviklingsarbeid knyttet til ny læreplan har vært gjennomført	Småtrinnet (1.-4.)	Mellomtrinnet (5.-7.)	Ungdomstrinnet (8.-10.)
Nesten ingen	12,5%	19,0%	26,3%
Lav	50,0%	66,7%	52,6%
Tilfredsstillende	37,5%	14,3%	15,8%
Høyt	0,0%	0,0%	5,3%
Svært høyt	0,0%	0,0%	0,0%

Tabell 9: Respondentenes opplevelse av mengde utviklingsarbeid sortert etter skoletrinn

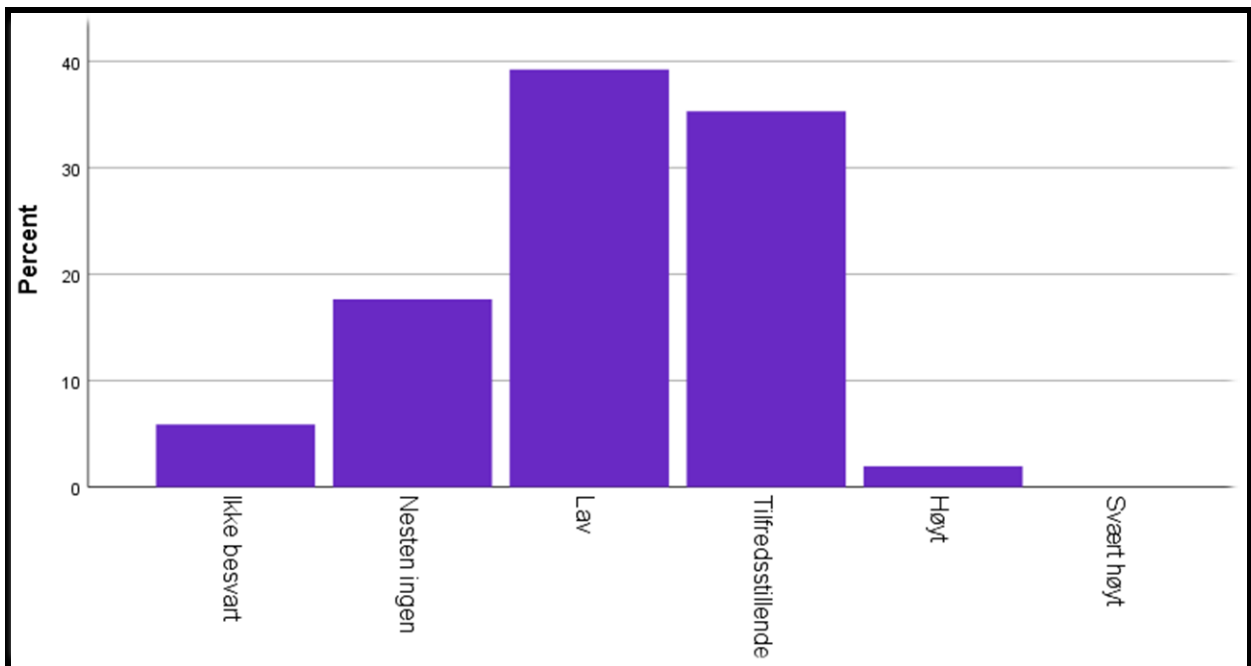
Deler man inn etter skoletrinn kan det se ut som at respondentene som underviser ved småtrinnet er litt mer fornøyd med mengden utviklingsarbeid som er gjennomført. Respondentene ved mellomtrinnet og ungdomstrinnet oppfatter det derimot som at ingen eller en lav mengde utviklingsarbeid har blitt gjennomført. Respondentene ved ungdomstrinnet er de som flest opplever at det ikke har vært gjennomført noe utviklingsarbeid knyttet til læreplanen. Tallene betyr ikke at det ikke har vært gjennomført noe utviklingsarbeid knyttet til læreplanen, men kun respondentenes oppfattelse av tiden dedikert til utviklingsarbeid.

I et av fritekst-svarene fremhever en respondent rollen ledelsen har i arbeidet:

- “Det er kanskje relevant for undersøkelsen å spørre om hvilken holdning skoleledelsen har til kompetanseheving, tidsbruk og økonomi. På min skole er ledelsen positiv, noen har fått kurs, men kun matte-lærere.”

Videre er det mer avsatt tid spesifikt til utviklingsarbeidet er det klart vanligste respondentene etterlyser:

- “Viktig at det blir forankret hos ledelsen, slik at det blir avsatt tid.”
- “Det har vært lite fokus på at programmering er en del av fagfornyelsen. Vi har jobbet lite med det i utviklingstiden/fellestiden.”
- “Må settes av tid til kompetanseheving i kollegiet.”
- “Vi har ikke jobbet noe med programmering i fagfornyelsen. Og har jobbet lite med fagfornyelsen etter koronapandemien.”
- “Da også med tanke på å få godt nok kvalifiserte, og motiverte, lærere som har fått nok tid til å arbeide med det til at det blir en naturlig del av faget.”



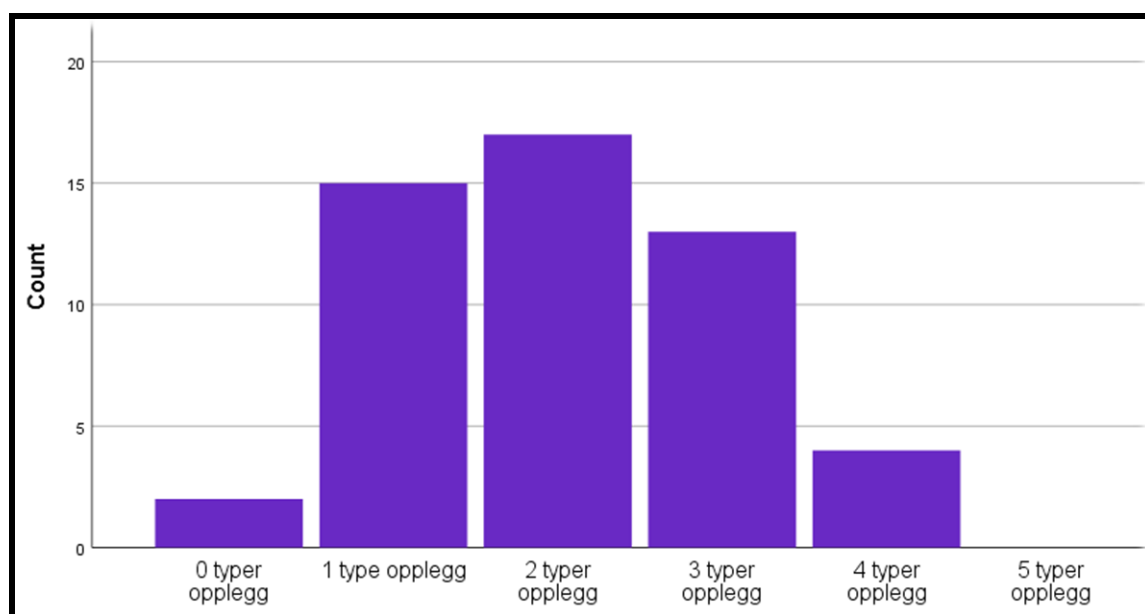
Figur 12: Respondentenes opplevelse av mengde generelt utviklingsarbeid

I sammenheng med generelt utviklingsarbeid er tallene litt bedre, men det er fortsatt rundt 60% av respondentene som føler at skoleutviklingen har vært mindre enn tilfredsstillende.

Type utviklingsarbeid	Totalt	Småtrinnet (1.-4.)	Mellomtrinnet (5.-7.)	Ungdomstrinnet (8.-10.)
Kompetansepakker	56,9%	50,0%	65,2%	50,0%
Felles utviklingsarbeid	66,7%	75,0%	60,9%	70,0%
Uformelle kurs	37,3%	25,0%	39,1%	40,0%
Universitetskurs	21,6%	25,0%	17,4%	25,0%
Andre opplegg	11,8%	25,0%	13,0%	5,0%

Tabell 10: Typer utviklingsarbeid respondentene har deltatt i

Som en helhet har rundt halvparten av respondentene deltatt i opplegg knyttet til kompetansepakker, også på tvers av skoletrinnene. Mellom 60% og 75% har arbeidet i fellesskap med utviklingsarbeid. Færre har deltatt på uformelle kurs, der på mellomtrinnet og ungdomstrinnet har rundt 40% deltatt, mens kun 25% har deltatt på småtrinnet. Rundt 12% oppgir å ha deltatt i andre opplegg. I fritekst-svarene fremheves opplegg gjennom Osloskolen i 2020, utforsking i Minecraft med hjelp fra en facebookgruppe, og opplegg på vitensenter eller newtonsenter. Tallene sier ingenting om hvor ofte eller i hvor stor grad deltakerne har deltatt på disse typene opplegg.



Figur 13: Antall opplegg respondentene har tatt del i

I svarene kan man også lese ut hvor mange typer opplegg respondentene har deltatt i. Her kan man lese av at rundt 60% av respondentene har deltatt i en eller to typer opplegg, mens nesten 90% har deltatt på mellom en og tre typer. For de fleste skoler har det altså i stor grad vært fokus på bruk av kompetansepakker og/eller felles utviklingsarbeid som metoder, da dette både er de opplegge flest lærere har deltatt i, og det er den kombinasjonen som dukker opp flest ganger i datamaterialet (19,6% har tatt del i kun kompetansepakker og felles utviklingsarbeid, mens 43,1% har tatt del i de to oppleggene pluss en eller to andre).

Fra fritekstsvar kommer det frem at noen av respondentene ønsker flere muligheter for kompetanseheving, både formell og uformell:

- “Har søkt videreutdanning i programmering to år på rad, men det er ikke prioritert på vår skole.”
- “Jeg vil gjerne ha info om hvordan man kan få mer kompetanse på området, her har jeg kun vært innom code.org.”

Der rundt 20% har vært gjennom universitetskurs for kompetanseheving i programmering kan det være flere som denne respondenten som kunne ønsket å få mer formell kompetanse, men der det heller ikke blir prioritert i forhold til bruk av tid og ressurser.

4.5.2 Nyttighet og effekt

For å kunne vurdere utviklingsarbeidet er det ikke bare nok å se på i hvor stor grad det har vært gjennomført, men også på hvilken type utviklingsarbeid som har vært gjennomført. Både respondentens egen oppfattelse av utviklingsarbeidet, og før og etter verdier, som det gjøres statistisk analyse av, blir spurt om i undersøkelsen. På den måten kan lærernes egen oppfattelse av hva som er nyttigst sammenlignes opp mot hva som ser ut til å ha størst effekt ifølge den statistiske analysen.

Nyttighet	Kompetansepakker fra UDIR	Felles utviklingsarbeid	Uformelle kurs	Universitetskurs
Ikke nyttig	0,0%	2,9%	0,0%	0,0%
Litt nyttig	44,8%	38,2%	26,3%	18,2%
Tilfredsstillende	17,2%	20,6%	15,8%	9,1%
Nyttig	13,8%	17,6%	36,8%	45,5%
Svært nyttig	24,1%	20,6%	21,1%	27,3%

Tabell 11: Respondentenes opplevelse av typer utviklingsarbeids nyttinghet

Denne tabellen tar for seg respondentenes egen opplevelse av nyttinghet knyttet til de forskjellige typene utviklingsarbeid. Respondentene er relativt enige i at ingen av formene for utviklingsarbeid ikke er nyttige. Det er kun cirka 3% som mener at felles utviklingsarbeid ikke er nyttig, ellers er alt minst litt nyttig. For kompetansepakker og felles utviklingsarbeid synes cirka 45% av respondentene at den typen arbeid er litt nyttig. Mens for begge er det altså cirka 55% som synes den typen utviklingsarbeid er tilfredsstillende nyttig eller bedre. For uformelle kurs er samlingen litt sterkere i toppen, der over 55% synes det er nyttig eller svært nyttig. Den mest positive vurderingen finner vi ved universitetskurs, der over 70% synes at det er nyttig eller svært nyttig. Blant disse gruppene finner statistiske sammenligninger derimot ingen signifikant forskjell bortsett fra mellom hvor nyttig respondentene synes kompetansepakker fra UDIR og felles utviklingsarbeid er, der respondentene synes utviklingsarbeid (Median=3,5) er signifikant mer nyttig enn kompetansepakkene (Median=2,5) ($df=21$, $Z=-2,326$, $p=0,020$).

4.5.3 Programmeringskompetanse

Et viktig aspekt ved læreres undervisning, er deres faglige kompetanse, og legger selve grunnlaget for undervisningen. I spørreundersøkelsen blir respondentene spurt om å vurdere egen programmeringskompetanse i før- og etter-verdier, for å kunne sammenligne utviklingen etter innføringen.

Programmeringskompetanse	Før utviklingsarbeid	Etter utviklingsarbeid
Ingen	43,1%	11,8%
Noe	33,3%	52,9%
Tilfredsstillende	19,6%	23,5%
Høy	3,9%	9,8%
Svært høy	0,0%	2,0%

Tabell 12: Respondentenes vurdering av egen programmeringskompetanse før og etter utviklingsarbeid

På spørsmål om egen programmeringskompetanse før arbeidet svarer mer enn 70% at de har lavere enn tilfredsstillende programmeringskompetanse, derav har cirka 45% ingen programmeringskompetanse, og cirka 30% at de har noe kompetanse. Etter arbeidet har tallene bedret seg noe da kun 12% vurderer egen kompetanse til ingen, og cirka 50% til noe. Og fra før-verdiene til etter-verdiene finner vi en signifikant endring ($df=50$, $Z=-4,849$, $p<0,001$).

Programmeringskompetanse	Før arbeid	Etter - arbeid med kompetanse-pakker	Etter - felles utviklingsarbeid	Etter - uformelle kurs	Etter - universitetskurs
Ingen	43,1%	10,3%	11,8%	5,3%	9,1%
Noe	33,3%	51,7%	50,0%	52,6%	36,4%
Tilfredsstillende	19,6%	27,6%	26,5%	15,8%	18,2%
Høy	3,9%	6,9%	8,8%	21,1%	27,3%
Svært høy	0,0%	3,4%	2,9%	5,3%	9,1%

Tabell 13: Respondentenes vurdering av egen programmeringskompetanse før og etter utviklingsarbeid samlet i grupper etter deltakelse i typer utviklingsarbeid

Her er respondentene samlet i grupper etter hvilke typer utviklingsarbeid de har deltatt i. Gruppene er ikke eksklusive, altså en person som har deltatt i arbeid med kompetansepakker og på uformelle kurs telles med i begge gruppene. Sammenligner man de forskjellige typene arbeid mot før-verdiene, har alle ført til bedre tall. Før arbeidet vurderte over 75% av respondentene egen kompetanse til noe eller ingen, men etter arbeidet er tallene for gruppene gått ned til mellom 60% og 45%. Forskjellen mellom før-verdien og etter kompetansepakker ($df=28$, $Z=-3,900$, $p<0,001$), etter utviklingsarbeid ($df=33$, $Z=-3,710$, $p<0,001$), etter uformelle kurs ($df=18$, $Z=-2,673$, $p=0,008$), og etter universitetskurs ($df=10$, $Z=-2,646$, $p=0,008$) er alle signifikante. Altså har respondentene som gruppe hatt en signifikant positiv endring av programmeringskompetansen fra alle fire typene utviklingsarbeid.

Nivå av programmeringskompetanse	Før arbeid	0-100 elever	101-200 elever	201-300 elever	300+ elever
Ingen	43,1%	0,0%	0,0%	21,4%	18,8%
Noe	33,3%	88,9%	50,0%	50,0%	31,3%
Tilfredsstillende	19,6%	0,0%	33,3%	21,4%	31,3%
Høy	3,9%	11,1%	16,7%	0,0%	18,8%
Svært høy	0,0%	0,0%	0,0%	7,1%	0,0%

Tabell 14: Respondentenes vurdering av egen programmeringskompetanse før og etter utviklingsarbeid samlet i grupper etter skolestørrelse

Om vi samler respondentene i grupper basert på skolestørrelse blir gruppene eksklusive, da respondentene tilhører kun en av kategoriene. Hos alle gruppene her ser vi en stor nedgang i antall respondenter som rapporterer å ha ingen programmeringskompetanse. Ved skolene med 0-100 elever rapporterer derimot nesten 90% å kun ha noe kompetanse, og viser ikke en signifikant forskjell fra før-verdiene ($df=8$, $Z=-1,732$, $p=0,083$). Gruppen 101-200 elever viser heller ikke en signifikant forskjell fra før-verdiene ($df=5$, $Z=-1$, $p=0,317$). Gruppene 201-300 elever ($df=13$, $Z=-2,646$, $p=0,008$) og 300+ elever ($df=15$, $Z=-2,828$, $p=0,005$) viser begge en signifikant endring fra før-verdiene.

4.5.4 Selvtillit for undervisning

Et til aspekt ved læreres kompetanse er selvtillit i egen undervisning i programmering. I spørreundersøkelsen blir respondentene spurt om å vurdere egen selvtillit i før- og etter-verdier, for å kunne sammenligne utviklingen etter innføringen.

Selvtillit for undervisning i programmering	Før utviklingsarbeid	Etter utviklingsarbeid
Ingen	41,2%	23,5%
Noe	29,4%	35,3%
Tilfredsstillende	21,6%	23,5%
Høy	7,8%	17,6%
Svært høy	0,0%	0,0%

Tabell 15: Respondentenes vurdering av egen selvtillit for undervisning før og etter utviklingsarbeid

På spørsmål om respondentenes selvtillit for undervisning i programmering svarte kun 30% at de hadde tilfredsstillende eller høyere selvtillit før gjennomført utviklingsarbeid, og 40% svarte at de hadde ingen selvtillit. Etter utviklingsarbeidet har de tallene endret seg til 40% med tilfredsstillende eller høyere selvtillit, og 25% hadde ingen selvtillit. Endringene fra før-verdiene til etter-verdiene var for hele gruppen signifikante ($df=50$, $Z=-3,443$, $p=0,001$).

Selvtillit for undervisning i programmering	Før arbeid	Etter – kompetanse-pakker	Etter - utviklingsarbeid	Etter - uformelle kurs	Etter - universitetskurs
Ingen	41,2%	17,2%	41,2%	21,1%	9,1%
Noe	29,4%	41,4%	17,6%	31,6%	45,5%
Tilfredsstillende	21,6%	24,1%	32,4%	15,8%	9,1%
Høy	7,8%	17,2%	8,8%	31,6%	36,4%
Svært høy	0,0%	0,0%	0,0%	0,0%	0,0%

Tabell 16: Respondentenes vurdering av egen selvtillit for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter deltakelse i typer utviklingsarbeid

Om man samler respondentene i grupper basert på hvilke opplegg de har deltatt i, kan vi se en forbedring i andel som kjenner på ingen selvtillit for undervisning for nesten alle gruppene. Hos

de som har deltatt i felles utviklingsarbeid, er det nøyaktig samme andel som kjenner på ingen selvtillit etter arbeidet som før. Ellers er det stort sett positive endringer for alle gruppene. Andelen som har høy selvtillit går fra cirka 8% før arbeidet til nesten 20% for de som har arbeidet med kompetansepakkene, og over 30% for de som har deltatt på uformelle kurs eller universitetskurs. Mellom før-verdiene og etter-verdiene er det signifikante forskjeller for gruppene som har arbeidet med kompetansepakker (df=28, Z=-3,419, p=0,001), har deltatt i felles utviklingsarbeid (df=33, Z=-2,780, p=0,005), og har deltatt på universitetskurs (df=10, Z=-2,121, p=0,034). For gruppen som har deltatt på uformelle kurs (df=18, Z=-1,540, p=0,124) er det derimot ikke en signifikant forskjell mellom før- og etter-verdiene.

Selvtillit for undervisning i programmering	Før arbeid	0-100 elever	101-200 elever	201-300 elever	300+ elever
Ingen	41,2%	11,1%	16,7%	35,7%	18,8%
Noe	29,4%	44,4%	33,3%	28,6%	43,8%
Tilfredsstillende	21,6%	22,2%	33,3%	21,4%	18,8%
Høy	7,8%	22,2%	16,7%	14,3%	18,8%
Svært høy	0,0%	0,0%	0,0%	0,0%	0,0%

Tabell 17: Respondentenes vurdering av egen selvtillit for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter skolestørrelse

Dersom vi samler gruppene basert på skolestørrelse ser vi en forbedring i antallet som kjenner på ingen selvtillit, fra 40% før, til under 20% for alle gruppene unntatt gruppen 201-300 elever. Der er det fortsatt en liten forbedring ned til 35%, bare ikke like drastisk som hos de andre gruppene. Videre ser vi en økning i antall som kjenner på høy selvtillit fra 8% opp til cirka 15% eller høyere for alle fire gruppene. Mellom før- og etter-verdiene er det ikke signifikante forskjeller for gruppene 0-100 elever (df=8, Z=-1,141, p=0,102), og 101-200 elever (df=5, Z=-0,447, p=0,655). Det er signifikante forskjeller for gruppene 201-300 elever (df=13, Z=-2,449, p=0,014) og 300+ elever (df=15, Z=-2,271, p=0,023).

4.5.5 Motivasjon for undervisning

Motivasjonen for undervisning er en viktig brikke for å sikre god undervisning, da en motivert lærer vil ha bedre forutsetning for å ville lære mer, og å planlegge god undervisning. Respondentens blir spurt om å vurdere egen motivasjon i før- og etter-verdier, for å kunne sammenligne utviklingen etter innføringen.

Motivasjon for undervisning i programmering	Før utviklingsarbeid	Etter utviklingsarbeid
Ingen	19,6%	2,0%
Noe	35,3%	25,5%
Tilfredsstillende	23,5%	33,3%
Høy	19,6%	35,3%
Svært høy	2,0%	3,9%

Tabell 18: Respondentenes vurdering av egen motivasjon for undervisning i programmering før og etter utviklingsarbeid

På spørsmål om respondentenes motivasjon for undervisning i programmering finner vi de høyeste målingene for de tre faktorene, kompetanse, selvtillit, og motivasjon. Allerede før utviklingsarbeidet ser vi at nesten 50% følte på tilfredsstillende eller høyere motivasjon for undervisning. Etter utviklingsarbeidet har andelen som kjenner på tilfredsstillende eller høyere motivasjon økt til over 70%. Forskjellene mellom før- og etter-verdiene er også signifikant forskjellige (df=50, Z=-5,504, p<0,001).

Motivasjon for undervisning i programmering	Før arbeid	Etter – kompetanse-pakker	Etter - felles utviklingsarbeid	Etter - uformelle kurs	Etter - universitetskurs
Ingen	19,6%	0,0%	2,9%	0,0%	0,0%
Noe	35,3%	27,6%	20,6%	21,1%	27,3%
Tilfredsstillende	23,5%	37,9%	38,2%	31,6%	18,2%
Høy	19,6%	34,5%	35,3%	47,4%	54,5%
Svært høy	2,0%	0,0%	2,9%	0,0%	0,0%

Tabell 19: Respondentenes vurdering av egen motivasjon for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter deltakelse i typer utviklingsarbeid

Dersom man samler respondentene i grupper basert på hvilke opplegg de har deltatt i ser vi positive endringer for alle gruppene. For alle gruppene bortsett fra de som har deltatt i felles utviklingsarbeid er andelen som kjente på ingen motivasjon for undervisning gått ned til 0%. Det har også vært en stor nedgang i andelen som målte til noe motivasjon, der alle gruppene ligger på under 30%, og gruppene deltatt i felles utviklingsarbeid og på uformelle kurs har hatt

en nedgang til 20%. Andelen som kjenner på høy motivasjon gått opp for alle gruppene, til rundt 35% for gruppene med kompetansepakker og felles utviklingsarbeid, og til cirka 50% for gruppene som har deltatt i uformelle kurs eller universitetskurs. Forskjellen mellom før- og etter-verdiene for gruppene som har deltatt i arbeid med kompetansepakker (df=28, Z=-3,704, p<0,001), i felles utviklingsarbeid (df=33, Z=-3,874, p<0,001), på uformelle kurs (df=18, Z=-2,652, p=0,008), og på universitetskurs (df=10, Z=-2,070, p=0,038) er alle signifikante.

Motivasjon for undervisning i programmering	Før arbeid	0-100 elever	101-200 elever	201-300 elever	300+ elever
Ingen	19,6%	0,0%	0,0%	0,0%	0,0%
Noe	35,3%	44,4%	16,7%	42,9%	12,5%
Tilfredsstillende	23,5%	33,3%	0,0%	21,4%	43,8%
Høy	19,6%	22,2%	66,7%	28,6%	43,8%
Svært høy	2,0%	0,0%	16,7%	7,1%	0,0%

Tabell 20: Respondentenes vurdering av egen motivasjon for undervisning i programmering før og etter utviklingsarbeid samlet i grupper etter skolestørrelse

Samler man respondentene i grupper basert på skolestørrelsen ser man at blant alle disse fire gruppene er det etter arbeidet 0% som måler egen motivasjon til ingenting. Det er derimot et par respondenter som ikke ønsket å oppgi hvor mange elever som er på den skolen de underviser ved, og derfor får man den ulikheten mellom 2,9% som kjenner på ingen motivasjon etter felles utviklingsarbeid mot 0% blant alle fire gruppene her. Ellers kan vi se at det er kun for gruppene 101-200 elever og 201-300 elever at noen respondenter rapporterer svært høy motivasjon, selv etter aktuelt utviklingsarbeid. Alle gruppene har derimot en større andel respondenter som rapporterer høy motivasjon etter arbeidet, da spesielt gruppene 101-200 elever og 300+ elever. For gruppene 0-100 elever (df=8, Z=-1,633, p=0,102), og 101-200 elever (df=5, Z=-1,732, p=0,083) kan vi allikevel ikke finne en signifikant forskjell mellom før- og etter-verdiene. Men det gjør vi derimot for gruppene 201-300 elever (df=13, Z=-2,251, p=0,024) og 300+ elever (df=15, Z=-2,739, p=0,006).

For å se nærmere på den kombinerte effekten av oppleggene hadde det vært spennende å kunne sortere gruppen inn i flere undergrupper, for hvilke opplegg respondentene hadde deltatt i. For eksempel en gruppe for de som har deltatt kun i arbeid med kompetansepakker, en gruppe for

de respondentene som har deltatt i uformelle kurs og universitetskurs, men ikke deltatt i arbeid med kompetansepakker og felles utviklingsarbeid, en gruppe for de som har deltatt i alle de fire typene utviklingsarbeid, etc. Det er dessverre for få respondenter for å kunne gjøre en slik inndeling, og det kan derfor ikke tas ekstra hensyn til at flere av respondentene har deltatt i flere opplegg, og det kan ikke gjøres statistiske analyser av effekten av flere kombinerte opplegg.

5.0 Diskusjon

I dette kapitlet vil det først diskusjon om metoden og gjennomføringen av forskningen, for så å se på resultatene som ble presentert i forrige kapittel og se det opp mot teori presentert i kapittel 2. Diskusjonen av resultatene vil ta utgangspunkt i de to forskningsspørsmålene som er sentrale for oppgaven.

5.1 Drøfting av forskningsmetode

I dette del-kapitlet vil metodologiske faktorer diskuteres, i form av både vurdering av gjennomføringen av forskningen og kvaliteten i studien.

5.1.1 Spørreskjema

Forskningsmetoden som ble valgt for denne studien var spørreskjema med lukkede og åpne spørsmål. Målet med denne struktureringen var å få data som kunne analyseres ved hjelp av SPSS for å se trender i dataen, og å bruke fritekst-svarene til å fylle inn og kanskje forklare trendene. Dataen som ble samlet inn fylte rollen som var tenkt, og det var mulig å se trender i datasettene, og å få gode utfyllinger for det meste i fritekst-svarene. Dette gjelder både utfordringer knyttet til tid for å gjennomføre utviklingsarbeid, informasjon om opplegg i regi av vitensenter eller newtonsenter de har deltatt på, og noe om hva respondentene ønsker fremover i form av videreutdanning eller info om hvordan man kan få mer kompetanse på området.

Spørreskjemaet ble sendt ut til 63 skoler spredt over 16 kommuner i Nordland (Tabell 1; Figur 3). Fra de skolene ble det endelige tallet besvarelser på 56 lærere, hvorav 5 ikke var matematikklærere og ble vurdert delvis inkludert, men endte utelatt fra studien da problemstillingen spesifikt gjelder matematikklærere (Tabell 2). I datainnsamlingen var det ønskelig med så mange besvarelser som mulig, men det viste seg å være utfordrende. Det er nok flere bakenforliggende årsaker, deriblant koronapandemien, og trykket med at alle ved lærerutdanningene for første gang i år skriver masteroppgaver. Koronapandemien kan ha ført til et større sykefravær, noe som igjen har ført til et større arbeidspress på lærerne som var på jobb, noe som kan forklare at deltakelse i masterstudier måtte nedprioriteres. Det store presset som kom i form av at alle skriver masteroppgaver kan også ha kommet noe uventet på, da vi er første kull etter at alle lærerutdanninger ble omgjort til å være femårige masterstudier.

I utformingsperioden av spørreskjemaet var problemstillingen enda ikke helt fastlåst, så noe av spørreskjemaet er orientert bredere enn nødvendig, og kunne vært mer målrettet. Gjennom arbeidet med oppgaven har det dukket opp flere situasjoner og refleksjoner om endringer i spørreskjemaet som kunne forbedret forskningen. Mer målinger spesifikt om i hvor stor grad respondentene føler de mestrer forskjellige elementer innenfor programmering, for eksempel algoritmisk tenkning, bruk av fysiske elementer, blokk- og tekst-programmering kunne vært inkludert som elementer under “programmeringskompetanse”. Det hadde gitt muligheten for å i større grad undersøke lærernes kompetanse opp mot TPCCK-modellen, og på den måten avgjort hvilke elementer av lærernes kompetanse som trenger mest oppmerksomhet videre (Koehler et al., 2013; Mishra & Koehler, 2006). I sammenheng med hvilke typer utviklingsarbeid respondentene har deltatt i, kunne det vært nyttig å etterlyse effektiviteten av de forskjellige typene arbeid i relasjon til forskjellige emner relatert til programmering. Det kan være at felles utviklingsarbeid er bedre for å utvikle pedagogisk praksis, mens kompetansepakkene fungerer bra til å utvikle lærernes forståelse av algoritmisk tenkning, og uformelle eller formelle kurs hadde best resultater i forhold til egen evne til å kode. Da alle delene er viktig for god programmeringsundervisning ville det vært nyttig å sammenligne de forskjellige typene arbeid opp mot hverandre, for å kunne undersøke mer enn at arbeidet har hatt en effekt (Kunnskapsdepartementet, 2019b). Å undersøke hvorvidt respondentene opplever at skolen har den nødvendige kompetansen til å selv drive utvikling for programmering i skolen, eller om de opplever det som nødvendig å innhente ekstern kompetanse for å kunne utvikle lærernes kompetanse hadde også vært interessante elementer for videre utviklingsarbeid i skolene.

5.1.2 Kvalitet i studien

Da respondentene ble spurt om å vurdere egen kompetanse, selvtillit, og motivasjon i dag og før innføringen av læreplanen gjør det studien utsatt for minne-bias. Det vil si at respondentene ikke kan forventes å nødvendigvis huske ting korrekt. Dette gjør studien litt mindre reliabel, men det ville heller ikke vært mulig å gjøre målinger både før og etter innføringen i ettertid. Noe som bidrar til sterkere reliabilitet var at i det ferdige datasettet var det nesten ingen motsigelser i besvarelser, kun to respondenter hadde huket av for at de underviste i programmering i fag, men som de ikke markerte for hvor ofte de underviste i programmering. Spørsmålene i undersøkelsen dekte et bredt spektrum av elementer innenfor programmering i skolen og lærernes utvikling i relasjon til programmering, med et mål om å ha høyere validitet når det gjelder å undersøke læreres utvikling nettopp i relasjon til programmering i skolen. Studien har flere elementer som taler for at resultatene er litt mer generaliserbare innenfor

Nordland. Deriblant er aldersfordelingen, og kjønnsfordelingen blant respondentene relativt lik fordelingen blant lærerne generelt i Norge (Figur 4; Figur 5; Figur 6; Figur 7). Respondentene er også spredt ut over store deler av Nordland, og fra skoler i forskjellige størrelser, som gir en bredere geografisk forankring (Figur 3; Figur 8). Det hadde derimot vært ønskelig med flere respondenter, da det relativt lave antallet respondenter gjør det vanskelig å gjøre definitive generaliseringer, selv om man kan se trender ut fra datasettet, både i relasjon til utviklingen blant lærere og i forhold til utfordringer i utviklingsarbeidet. Funnene i studien forklares godt av tidligere forskning, noe som kan tale for at resultatene ville vært relativt like dersom man gjennomførte en lignende studie i samme området igjen.

5.2 Drøfting av resultater

I dette del-kapittelet skal resultatene drøftes opp mot teorien med mål om å svare på forskningsspørsmålene.

5.2.1 Mangel på tid og ressurser

Det første forskningsspørsmålet som tas opp i oppgaven er “Hvordan har det blitt lagt til rette for lærernes profesjonelle utvikling etter fagfornyelsen?”. Inn under dette spørsmålet ligger det både de materielle forholdene, og de organisatoriske forholdene. De materielle forholdene dekker tilgangen på undervisningsressurser og ressurser til å drive utviklingsarbeid, mens de organisatoriske forholdene handler om hvorvidt det er lagt til rette for utviklingen.

Ressurser

Innføringen av fagfornyelsen, har ført til at programmering skal implementeres i skolen, likevel er det fortsatt utfordringer tilknyttet undervisningsressurser, som trengs for å gjennomføre undervisning om programmering. Mer enn 60% av respondentene opplever selv å ikke ha tilfredsstillende eller bedre tilgang på relevante skolebøker for undervisningen (Tabell 8). Skolebøker er en stor ressurs for undervisning, og vil være et godt støtte-verktøy for lærernes planlegging og gjennomføring av undervisning. Relevant undervisningsmateriale om programmering kan med andre ord bidra til at lærere kan bruke mindre tid på å planlegge undervisningstimene når de legger opp til programmering. Dette vil igjen gi lærerne bedre tid til å kunne sette seg inn i ulike programmeringsprogram på egenhånd før undervisningen (Imsen, 2016, s.542-548).

Respondentene rapporterer å ha litt bedre tilgang på fysisk utstyr, der 50% opplever å ha tilfredsstillende eller bedre tilgang på utstyr (Tabell 8). Tilgang på utstyr i seg selv er ikke nok til å sikre god utvikling, da lærerne trenger god kunnskap i hvordan de kan gjennomføre pedagogisk undervisning med støtte i utstyret. (Menekse, 2015, s.22; Crick, 2017, s.17-18; Sisman & Kucuk, 2018, s.317-318). Men det betyr derimot også at 50% opplever at tilgangen på fysisk utstyr til bruk i undervisningen er mangelfull. Men en slik mangel på fysiske undervisningsmateriell som eksempelvis roboter, går man glipp av en god mulighet for engasjerende undervisning for spesielt de yngre elevene (Piedade et al., 2020, s.8), men også for eldre elever i form av Micro:bit som byr på flere muligheter (Monk, 2017, s.14-22). Dette er en viktig del av ledelsens ansvar, både i forhold til bruk for læreres utvikling, men også til bruk i undervisningen. På tross behov for mer forskning på pedagogisk bruk, fins det grunnlag for å si at bruk av fysisk utstyr har en positiv effekt på utviklingen av algoritmisk tenkning og feilsøking i programmer ved å enklere kunne se programmet stegvis og identifisere hvor problemet er (Bueie, 2019, s.28; Angeli, 2021, s.7-8). Det vil altså være gunstig dersom skolene oppnår bedre tilgang på fysisk utstyr, og det blir viktig at lærerne har den nødvendige kompetansen for bruk i undervisningen.

Nettbaserte ressurser er enklere å få tilgang til, da det fins flere verktøy som tilbyr støttemateriell og opplegg gratis, for eksempel Scratch, Code.org, og MakeCode (Micro:bit). Dette reflekteres også i svarene fra respondentene der nesten 70% rapporterer å ha tilfredsstillende eller bedre tilgang på nettbaserte ressurser (Tabell 8). Dette studiet viser derimot at om lag 65% av respondentene mener at de ikke har tilstrekkelige opplæring og kompetanse for programmering. Det vil da være nyttig dersom lærere får opplæring i hvordan de kan benytte seg av disse ressursene, da uten den tilstrekkelige erfaring og kompetanse hos lærerne vil likevel ikke disse ressursene komme til god nytte for skolene. (Falkner & Vivan, 2015, s.396-398).

Tilgangen på arbeidsrom er derimot mer mangelfull, og rundt 65% opplever å ha mindre enn tilfredsstillende tilgang (Tabell 8). Dette kan både gjøre det vanskeligere å benytte seg av fysisk utstyr, for eksempel for hinderløyper for roboter, og kan gjøre det vanskeligere å legge til rette for skaperverksteder. Å ha tilgang på en arena med tilstrekkelig utstyr for læring kan være gunstig både for undervisningen, og for lærernes egen eksperimentering, noe som igjen kan ha positive effekter på undervisningen (Rich, 2017, s.13).

Utviklingsarbeid

Når det gjelder utviklingsarbeid svarer nesten 70% at de opplever at det har vært gjennomført lite eller nesten ingenting i form av utviklingsarbeid knyttet til ny læreplan (Figur 11). Ser man på utviklingsarbeid som helhet har respondentene fått tatt del i litt flere typer opplegg, men allikevel er det 4% som opplever å ikke ha tatt del i noen type opplegg, og 30% som opplever å ha deltatt i kun en type opplegg (Figur 13). Det i seg selv er problematisk da i likhet med situasjonen ved innføringen i Japan rapporterer en stor gruppe av lærerne selv et lavt nivå av spesielt kompetanse og selvtillit for undervisning, noe som fremhever et behov (Ohashi, 2017, s.133-136; Tabell 12; Tabell 15). Dersom vi ser på respondentenes svar basert på skoletrinn kommer det frem at lærere ved høyere trinn er litt mindre fornøyd og opplever at det i mindre grad har blitt gjennomført relevant utviklingsarbeidet (Tabell 9). En mulig forklaring kan være i hvordan de faglige kompetansemålene for de forskjellige trinnene er svært forskjellige, så lærere vil ha forskjellig nivå av nødvendig kompetanse basert på hvem de skal undervise. Det kan være slik at i det utviklingsarbeidet som har blitt gjennomført ved skolene har man måtte gjort prioriteringer og valg som har resultert i et opplegg som skulle være dekkende for alle lærerne. I en slik situasjon må noe ha blitt prioritert, og man kan ha endt opp med et veldig grunnleggende opplegg som ikke har vært dekkende for de høyere skoletrinnene (Dağ, 2019, s.304-308; Vinnervik, 2022, s.21-22; Changpetch et al., 2022, s.9-16). Det kan være litt av forklaringen for forskjellen i hvorvidt lærerne opplever å ha deltatt i utviklingsarbeid, men tallene er allikevel lave både for hele gruppen, og for de tre skoletrinnene individuelt.

Gjennomføringen og prioriteringen av utviklingsarbeid slik at skolen kan operere i tråd med læreplanen er ledelsen og skoleeiers ansvar. Det innebærer både det å ta en lederrolle i arbeidet, men også å danne grunnlaget i form av å sette av tid og ressurser til arbeidet (Dons, 2010, s.85; Irgens, 2010, s.141-142). Ledelsen har et ansvar for å igangsette utvikling ved skolen i samarbeid med lærerne. Som leder har de et spesielt ansvar for å tydeliggjøre målene, og å legge til rette for å nå disse målene. I svarene kom det frem at flere av lærerne opplevde at det ble satt av lite tid til arbeidet, og for noen ble programmering heller ikke prioritert i arbeidet som ble gjort. Mangel på tid er en utfordring som kan dukke opp når det gjelder lærere og deres timeplaner, og er ofte heller ikke en utfordring ledelsen ved skolene ikke er klar over (Lyngnes & Rismark, 2014, s.193-194; Mausestagen & Helstad, 2019, s.168; Irgens, 2010, s.128-130). I fritekstsvarene er det også noen av lærerne som påpeker at programmering i liten grad har vært tema i utviklingstiden/fellestiden ved skolen, og hos noen har det utviklingen kun vært aktuell for matematikklærere. Det er problematisk da matematikklærere i liten grad underviser i de

andre fagene programmering skal inkluderes i, og man vil kunne gå glipp av effekten av algoritmisk tenkning i andre fag, dersom det blir sett på som relevant kun for matematikk (Sands et al., 2018, s.161-163; Tabell 3). Det er også uheldig dersom lærernes inntrykk om at det i liten grad har vært et tema i fellestiden skyldes at denne tiden fylles med annet arbeid som ikke er givende for lærerne, da rundt 45% av respondentene har svart at det felles utviklingsarbeidet de har deltatt i har vært mindre enn tilfredsstillende nyttig (Tabell 11). Det er mulig at dette har rot i enten at oppleggene ikke er tilpasset de individuelle lærernes behov, og at det derfor ikke treffer hele gruppen, eller at oppleggene er planlagt for i større grad å “fylle tiden” enn i samarbeid med lærerne for deres utvikling (Eftedal & Damsgaard, 2014, s.72-73). Dersom arbeidet tar for seg blokkprogrammering vil det være relevant for de lavere trinnene der man bør arbeide mer med tankegangen bak programmering, som er en viktig del av programmeringen (Crick, 2017, s.13). Det vil da være noe mindre relevant for lærerne ved de høyere trinnene, der Python kunne vært mer relevant både i overgangen til tekstbasert, eller i videre programmering (Kluge, 2021, s.143; Haraldsrud et al., 2020, s.102-103). I et forsøk på å hjelpe med planlegging og organisering av slikt arbeid utarbeidet myndighetene de digitale kompetansepakkene som et forsøk på støtte til arbeidet (Dolonen et al., 2019, s.21). Ifølge respondentene de til en viss grad blitt tatt i bruk i skolene, men arbeidet som er blitt gjort i sammenheng med pakkene har blitt opplevd i stor grad like nyttig som annet felles utviklingsarbeid, med cirka 40% som oppgir at det har vært mindre enn tilfredsstillende nyttig (Tabell 11). Kompetansepakker kunne ha vært et godt verktøy for skolene for å utvikle lærernes kompetanse. Men to forutsetninger for at kompetansepakkene kan være et nyttig verktøy blir likevel at kompetansepakkene utvikles på en måte som gjør at lærerne opplever dem som nyttige for egen undervisningspraksis, og at arbeidet legges opp på en god måte. En måte å sikre dette på er å involvere lærerne i planleggingen av arbeidet. Å involvere lærerne i planleggingen av utviklingsarbeidet kan også ha en positiv effekt på lærernes opplevelse av utviklingsarbeidet, og kan bidra til bedre resultater for alle typer arbeid (Nilsen, 2010, s.114-115; Imsen, 2016, s.542-548).

I Stortingsmelding 31 (2007-2008) blir det fremhevet at formell kompetanse gir bedre og mer varige resultater over uformell kompetanse (Kunnskapsdepartementet, 2008, s.40). I formelle kurs går utdanningen også gjerne over lengre tid i motsetning til et uformelt kurs, noe som også viser større positive endringer (Menekse, 2015, s.22). Med utgangspunkt i respondentene er det under 10% som har formell utdanning innenfor programmering, og kun 20% som har fått deltatt i formelle kurs ved universiteter eller høyskoler (Tabell 10). En av respondentene oppga i et

fritekst-svar å ha bedt om videreutdanning, men ikke fått muligheten. Da det ikke var et eksplisitt spørsmål, blir det vanskelig å si om det gjelder flere av respondentene. Om vi derimot ser for oss at flere ønsker formell kompetanse er det 80% av respondentene som ikke har deltatt i formelle kurs. Likevel fremkommer det fra aktuell forskning og teori at kurs og utdanning er viktig for at lærere skal opparbeide seg kompetanse for drive med programmering i undervisningen (Dağ, 2019, s.304-308; Rouhani et al., 2020, s.54-56). Her fremkommer det et tydelig behov for endringer i skolen, for at lærerne skal kunne drive variert undervisning om programmering på en måte som er i tråd med kompetansemålene for faget (Falkner & Vivan, 2015, s.396-398; Menekse, 2015, s.22; Erümit, 2020, s.1032-1034). Dersom man skal legge til rette for at alle får muligheten til å nå delta i formell utvikling krever dette både mye tid fra lærerne, og ressurser fra utdanningsinstitusjoner som i mange tilfeller ikke vil strekke til, og kan trenge støtte fra private kompetansmiljøer (Kjällander et al., 2018, s.25-28). Blant respondentene er det flere som har deltatt i opplegg i regi av vitensenter eller newtonsenter, og opplever at dette har vært nyttig. Dette antas å være arbeid likt det som vil foregå i skaperverksted. Der det er mulig kan denne typen opplegg oppleves som praksisnære dersom oppleggene har en tydelig sammenheng med undervisning, og ha positive resultater for deltakerne (Kong & Lao, 2019, s.981; Kjällander et al., 2018, s.25-28). Samarbeidet mellom grunnskoler og høyere utdanningsinstitusjoner vil derimot allikevel være avgjørende i arbeidet fremover da læreres og lærerstudenters programmeringskompetanse må forbedres, og formell utdanning er et viktig verktøy i den prosessen (Menekse, 2015, s.22). Det er også et samarbeid som produserer gode resultater både for læreres og lærerstudenters programmeringskompetanse, problemløsningskompetanse, og selvtillit for undervisning, noe som alle er viktige elementer dersom skolene skal kunne legge til rette for god undervisning og god dybdelæring i programmering (Bati, 2021, s.13; Sentance & Csizmadia, 2017, s.489).

5.2.2 Positiv utvikling, for større skoler

Det andre forskningsspørsmålet som er “Hvordan har matematikk-læreres situasjon utviklet seg etter fagfornyelsen?”. Dette spørsmålet gjelder utviklingen i målingene av lærernes programmeringskompetanse, selvtillit for undervisning, og motivasjon for undervisning.

Kompetanse, selvtillit, og motivasjon

Resultatene for de tre aspektene som ble spurt etter i denne oppgaven har alle tre hatt en positiv utvikling. For kompetanse har de som har deltatt på universitetskurs og uformelle kurs vært de

som har rapportert sterkest utvikling, men respondentene som gruppe har hatt positiv utvikling i programmeringskompetanse uansett hvilket opplegg de har deltatt i (Tabell 12; Tabell 13). Det kan være et tegn på at et av de viktigste faktorene i utvikling er at det blir gjennomført utviklingsarbeid for lærerne. De forskjellige typene arbeid vil ha forskjellig virkningsgrad, men alle viser positiv utvikling innenfor programmeringskompetanse.

For respondentenes selvtilit er resultatene stort sett like, i den form av at det er de som har deltatt på universitetskurs som rapporterer om sterkest utvikling, men respondentene som helhet har hatt en positiv utvikling også her. Gruppen respondenter som har deltatt på uformelle kurs viser derimot ikke en signifikant forskjell i før- og etter-verdiene (Tabell 15; Tabell 16). I dette tilfellet er det altså ikke stor nok forskjell i målingene til å si at det har vært en endring, selv om målingene selv har flyttet seg noe i positiv retning. Dette kan skyldes at kursene har vært dårlig planlagt og dermed ikke hatt en større effekt, eller har hatt en avskrekkende effekt for emnet (Woodrow, 1992, s.216-217), eller det kan skyldes unøyaktighet eller feil i målingene. Årsaken blir vanskelig å forklare uten mer informasjon, så her kan det ikke trekkes noen slutninger om hvorfor det ikke er en signifikant forskjell i resultatene.

På spørsmålene om motivasjonen for undervisning i programmering rapporterer respondentene de høyeste verdiene blant de tre aspektene. Allerede før de deltok i arbeid knyttet til programmering var over halvparten motiverte for å undervise i programmering. Etter arbeidet har det økt til over 70%, noe som bygger på påstanden om at lærerne er motiverte for å undervise i programmering (Tabell 18; Tabell 19).

Den positive utviklingen kan tyde på at arbeidet som har blitt gjort har hatt en effekt, selv om effekten kan argumenteres til å ikke ha vært stor nok. Formelle opplegg gjennomført for praktiserende lærere har en sterk positiv effekt på kompetanse, motivasjon, og selvtilit, og det samme har emner i lærerutdanningen på lærerstudenter (Rouhani et al.,2020, s.54-56; Dağ, 2019, s.304-308), og er konkrete tiltak myndighetene og universiteter kan og har gjort en innsats for å forbedre.

Skolestørrelse

Skolestørrelse er det som har utgjort størst forskjell for resultatene av utviklingen. I forhold til de tre aspektene oppgaven måler, kan vi ikke observere signifikante forskjeller for noen av dem blant respondentene fra skoler med mindre enn 200 elever (Tabell 14; Tabell 17; Tabell 20).

Dette kan henge sammen med et mindre profesjonsfelleskap, noe som byr på ekstra utfordringer i forhold til muligheter for kompetanseutveksling innad i kollegiet. Da mindre skoler ofte ligger mer landlig, byr det også på problemer knyttet til deltakelse både på formelle og uformelle kurs, da reisen krever mer tid og ressurser for å ha muligheten til å delta (Abalde, 2014, s.15). Her kan støtte ovenfra både i form av kompetansepakker og andre opplegg, og direkte støtte til å involvere eksterne kompetansemiljø være ekstra viktig for å sikre god utvikling ved de skolene som har et større behov (Mausethagen & Helstad, 2019, s.167-181).

6.0 Avslutning

I dette kapittelet vil det bli gjort avsluttende refleksjoner rundt problemstillingen basert på diskusjonen i forrige kapittel. Avsluttende vil det også bli presentert konsekvenser for profesjonen, og mulige temaer for videre forskning.

6.1 Problemstillingen

Målet for denne studien har vært å besvare følgende problemstilling: “Hvordan har matematikklæreres utvikling i relasjon til programmering foregått etter fagfornyelsen, og hvilken effekt har det hatt?”. For å besvare denne problemstillingen ble det benyttet et spørreskjema for å kunne samle data fra en større gruppe lærere. Målet med undersøkelsen og datamaterialet var å se etter større trender eller mønster om utviklingen som er skjedd i skolene og som også var statistisk signifikante.

Funnene indikerer at det er noe variasjon i Nordland i forhold til graden av gjennomføring av utviklingsarbeid, men at det jevnt over ikke har vært satset stort på. En stor andel av respondentene rapporterer å mangle tilgang på viktige ressurser som skolebøker, fysisk, og nettbaserte ressurser for bruk i programmeringsundervisningen, noe som blant annet gjør lærernes egen undervisning mer krevende.

Majoriteten av respondentene opplever at det har blitt gjennomført lite, eller nesten ingen form for utviklingsarbeid knyttet til spesifikk til den nye læreplanen, noe som er problematisk når det i læreplanen har blitt introdusert et tema som er helt nytt for mange skoler og lærere. Det utviklingsarbeidet som har blitt gjennomført er respondentene noe delt i deres oppfattelse av hvor nyttig de forskjellige oppleggene er. Mens det ikke er signifikante forskjeller mellom lærernes opplevelse av nytthet, er det formelle og uformelle kurs som scorer høyest. Her mener 72,8% at formelle kurs er nyttig eller svært nyttig, og 57,9% mener uformelle kurs er nyttig eller svært nyttig (Tabell 11). Denne oppfatningen stemmer også overens med tidligere forskning som viser at formelle kurs eller emner i lærerutdanningen som foregår over en lengre periode har varige positive effekter både for kompetanse, motivasjon, og selvtillit (Dağ, 2019, s.304-308; Rouhani et al., 2020, s.54-56; Menekse, 2015, s.22). Fra respondentenes målinger over egen kompetanse, selvtillit, og motivasjon kan vi på tross av lærernes delte oppfatninger se en signifikant positiv endring for deltakerne i alle gruppene fra før- til etter-målingene, for de større skolene. At skolene med mindre enn 200 elever ikke viser en signifikant endring taler

sterkt for at mindre skoler trenger ekstra støtte fra øvre organer, enten fra kommunen eller myndighetene, på lik linje med hva som ble gjort etter LK06, der det var en forståelse av at skolene ikke hadde den nødvendige kompetansen selv.

6.2 Studiens betydning og videre forskning

Betydningen av denne studien er en tydeliggjøring av behovet for en større satsning på kompetanseheving for lærere som skal undervise i programmering. Dersom skolene skal være i stand til å legge undervisningen på et slikt nivå at elevene kan oppnå kompetansemålene, må også læreren ha den nødvendige pedagogiske, teknologiske, og faglige kompetansen. Med en så stor andel lærere som ikke er fornøyde eller kjenner at utviklingsarbeidet ikke er nyttig kan det være nødvendig for skolene å gjøre en større innsats for å involvere lærerne i planleggingen. Det kan forbedre både motivasjonen for, og resultatene av arbeidet. Da lærerne selv er motiverte og interessert i å tilegne seg kompetansen som trengs for å undervise er det viktig at det legges til rette for denne utviklingen så tidlig og omfattende som mulig.

Fra denne forskningen er det flere muligheter til steg videre, noen av dem ble nevnt tidlig i diskusjonskapittelet. Spørsmålet om hvilke typer utviklingsarbeid som egner seg best til hvilke aspekter ved utviklingen vil være spennende og nyttig for det arbeidet som skal skje videre i skolene. En aksjonsstudie der man arrangerer opplegg i samme stil, men for forskjellige emner ved flere skoler vil gi et interessant bilde av hvordan man bør gjøre utviklingsarbeid for programmering videre. Felles utviklingsarbeid kan for eksempel være det beste verktøyet for læreres utvikling av undervisningskompetanse for programmering, mens det kan ha lite effekt på utviklingen av læreres evne til å kode eller læreres forståelse av algoritmisk tenkning. Det kan også være interessant å undersøke lærere og rektors opplevelse av arbeidet så langt, via en intervjustudie ved flere skoler. Hvorvidt lærere og rektorer har lik oppfatning av hvordan kompetanseutviklingen har gått, og hva som er nødvendig videre for å sikre matematikklærernes utvikling kunne også gi god innsikt i hva skolene bør være observant på i planleggingen av videre arbeid.

7.0 Litteraturliste

- Abalde, M. A. (2014, november 03). School Size Policies: A Literature Review. *OECD Education Working Papers*, 106. <https://doi.org/10.1787/5jxt472ddkjl-en>
- Angeli, C. (2021, juni). The effects of scaffolded programming scripts on pre-service teachers' computational thinking: Developing algorithmic thinking through programming robots. *International Journal of Child-Computer Interaction*. <https://doi.org/10.1016/j.ijcci.2021.100329>
- Bati, K. (2021). Integration of Python into Science Teacher Education, Developing Computational Problem Solving and Using Information and Communication Technologies Competencies of Pre-service Science Teachers. *Informatics in Education*. DOI:10.15388/infedu.2022.12
- Berggren, S., & Jom, P. (2019, november 21). *Fagartikkel: Lærerne er positive til programmering - men mangler kunnskap*. Utdanningsnytt. Hentet mai 15, 2022, fra <https://www.utdanningsnytt.no/fagartikkel/fagartikkel-laererne-er-positive-til-programmering---men-mangler-kunnskap/220753>
- Bjørkeng, P. K. (2020, september 9). Nesten ingen lærere kan faget de nå må lære bort. *Aftenposten*. <https://www.aftenposten.no/kultur/i/OpLb33/nesten-ingen-av-laererne-kan-faget-de-naa-skal-laere-bort>
- Bocconi, S., Chiocciariello, A., & Earp, J. (2018). *The Nordic approach to introducing Computational Thinking and programming in compulsory education* [Report prepared for the Nordic@BETT2018 Steering Group]. <https://doi.org/10.17471/54007>
- Brinkmann, S., & Kvale, S. (2021). *Det kvalitative forskningsintervju* (T. M. Anderssen & J. Rygge, Trans.; 3rd ed.). Gyldendal akademisk.
- Bueie, H. (2019). *Programmering for matematikklærere*. Universitetsforlaget.
- Changpetch, C., Panjaburee, P., & Srisawasdi, N. (2022). A comparison of pre-service teachers' variable misconceptions in various computer-programming preferences: findings to teacher education course. *Journal of Computers in Education*, 9, 149-172. <https://doi.org/10.1007/s40692-021-00200-0>
- Creswell, J. W., & Creswell, J. D. (2018). *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (5th ed.). SAGE Publications.
- Crick, T. (2017). *Computing Education: An Overview of Research in the Field*. Royal Society. <https://cronfa.swan.ac.uk/Record/cronfa43589>

- Dağ, F. (2019). Prepare pre-service teachers to teach computer programming skills at K-12 level: experiences in a course. *Journal of Computers in Education*, 6, 277-313. <https://doi.org/10.1007/s40692-019-00137-5>
- Dalland, O. (2012). *Metode og oppgaveskriving*. Gyldendal Norsk Forlag.
- Dolonen, J. A., Kluge, A., Litherland, K., & Mørch, A. (2019, november 8). *Litteraturgjennomgang av programmering i skolen*. UiO - DUO Vitenarkiv. Hentet mai 15, 2022, fra <https://www.duo.uio.no/handle/10852/76290?show=full>
- Dons, C. F. (2010). Styringsstruktur, ledelse og demokratisk dannelse i skolen. In R.-A. Andreassen, E. J. Irgens, & E. M. Skaalvik (Eds.), *Kompetent skoleledelse* (s.79-90). Tapir akademisk forlag.
- Eftedal, C. I., & Damsgaard, H. L. (2014). *-men hvordan gjør vi det? tilpasset opplæring i grunnskolen*. Cappelen Damm akademisk.
- Erümit, A. K. (2020). Effects of different teaching approaches on programming skills. *Education and Information Technologies*, 25, 1013-1037. <https://doi.org/10.1007/s10639-019-10010-8>
- Falkner, K., & Vivian, R. (2015). A review of Computer Science resources for learning and teaching with K-12 computing curricula: an Australian case study. *Computer Science Education*, 25(4), 390-429. <https://doi.org/10.1080/08993408.2016.1140410>
- Gjøvik, Ø., & Torkildsen, H. A. (2019). Algoritmisk tenkning. *Tangenten: Tidsskrift for matematikkundervisning*, 30(3), 31-37. <http://www.caspar.no/tangenten/2019/Tangenten%203%202019%20Gj%C3%B8vik%20Torkildsen.pdf>
- Haraldsrud, A. D., Sveinsson, H. A., & Løvold, H. H. (2020). *Programmering i skolen*. Universitetsforlaget.
- Heintz, F., Mannila, L., Nordén, L., Parnes, P., & Regnell, B. (2017). Introducing Programming and Digital Competence in Swedish K-9 Education. In A. Hellas & V. Dagienė (Eds.), *Informatics in Schools: Focus on Learning Programming: 10th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2017, Helsinki, Finland, November 13-15, 2017, Proceedings* (s. 117-128). Springer International Publishing. <https://ebookcentral-proquest-com.ezproxy.nord.no/lib/nord/detail.action?docID=6304970>
- IBM. (n.d.). *IBM SPSS Statistics*. IBM. Hentet mai 16, 2022, fra <https://www.ibm.com/products/spss-statistics>

- Imsen, G. (2016). *Lærerens verden: innføring i generell didaktikk* (5th ed.). Universitetsforlaget.
- Informatics Europe, ACM Europe, & ACM Europe and Informatics Europe liaison. (2013, April). *Informatics education: Europe cannot afford to miss the boat* [Report of the joint Informatics Europe & ACM Europe Working Group on Informatics Education]. Hentet mai 15, 2022, fra <https://www.informatics-europe.org/images/documents/informatics-education-acm-ie.pdf>
- Irgens, E. J. (2010). Rom for arbeid: Lederen som konstruktør av den gode skole. In R.-A. Andreassen, E. J. Irgens, & E. M. Skaalvik (Eds.), *Kompetent skoleledelse* (s.125-145). Tapir akademisk forlag.
- Johnson, D. D., & VanderStoep, S. W. (2009). *Research Methods for Everyday Life: Blending Qualitative and Quantitative Approaches*. John Wiley & Son.
- Justis- og beredskapsdepartementet. (2015, november 30). *Digital sårbarhet – sikkert samfunn*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/nou-2015-13/id2464370/>
- Kjällander, S., Åkerfeldt, A., Mannila, L., & Parnes, P. (2018). Makerspaces Across Settings: Didactic Design for Programming in Formal and Informal Teacher Education in the Nordic Countries. *Journal of Research on Technology in Education*, 34(1), 18-30. <https://doi.org/10.1080/21532974.2017.1387831>
- Kluge, A. (2021). *Læring med digital teknologi. Teorier og utviklingstrekk*. Cappelen Damm.
- Koehler, M. J., Mishra, P., & Cain, W. (2013). What is Technological Pedagogical Content Knowledge (TPACK)? *Journal of Education*, 193(3), 13-19. DOI:10.1177/002205741319300303
- Kommunal- og distriktsdepartementet. (2013, januar 7). *NOU Norges offentlige utredninger 2013: 2 - Hindre for digital verdiskaping*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/nou-2013-2/id711002/>
- Kong, S.-C., Lai, M., & Sun, D. (2020, juli). Teacher development in computational thinking: Design and learning outcomes of programming concepts, practices and pedagogy. *Computers & Education*, 151. <https://doi.org/10.1016/j.compedu.2020.103872>
- Kong, S.-C., & Lao, A. C.-C. (2019, februar). Assessing In-service Teachers' Development of Computational Thinking Practices in Teacher Development Courses. *SIGCSE '19: Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 976-982. <https://doi.org/10.1145/3287324.3287470>

- Kunnskapsdepartementet. (2004). *St.meld. nr. 030 (2003-2004)*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/stmeld-nr-030-2003-2004-/id404433/>
- Kunnskapsdepartementet. (2008). *St.meld. nr. 31*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/stmeld-nr-31-2007-2008-/id516853/>
- Kunnskapsdepartementet. (2013). *Meld. St. 20 (2012–2013)*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/meld-st-20-20122013/id717308/>
- Kunnskapsdepartementet. (2014). *NOU 2014: 7*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/NOU-2014-7/id766593/>
- Kunnskapsdepartementet. (2016, august 15). *Om lærarrollen. Et kunnskapsgrunnlag - Rapport fra ekspertgruppe*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/om-lararrolla.-eit-kunnskapsgrunnlag/id2555498/>
- Kunnskapsdepartementet. (2017a). *Framtid, fornyelse og digitalisering. Digitaliseringsstrategi for grunnsopplæringen 2017–2021*. Regjeringen.no. Hentet mai 15, 2022, fra <https://www.regjeringen.no/no/dokumenter/framtid-fornyelse-og-digitalisering/id2568347/>
- Kunnskapsdepartementet. (2017b). *Overordnet del – verdier og prinsipper for grunnsopplæringen*. Fastsatt som forskrift ved kongelig resolusjon. Læreplanverket for Kunnskapsløftet 2020.
- Kunnskapsdepartementet. (2019a). *Læreplan i kunst og håndverk (KHV01-02)*. Læreplan i kunst og håndverk (KHV01-02).
- Kunnskapsdepartementet. (2019b). *Læreplan i matematikk 1.–10. trinn (MAT01-05)*. Fastsatt som forskrift. Læreplanverket for Kunnskapsløftet 2020.
- Kunnskapsdepartementet. (2019c). *Læreplan i musikk (MUS01-02)*. Fastsatt som forskrift. Læreplanverket for Kunnskapsløftet 2020.
- Kunnskapsdepartementet. (2019d). *Læreplan i naturfag (NAT01-04)*. Fastsatt som forskrift. Læreplanverket for Kunnskapsløftet 2020.
- Leavy, P. (2017). *Research Design: Quantitative, Qualitative, Mixed Methods, Arts-Based, and Community-Based Participatory Research Approaches*. Guilford Publications.
- Lyngsnes, K., & Rismark, M. (2014). *Didaktisk arbeid* (3rd ed.). Gyldendal akademisk.

- Mausethagen, S., & Helstad, K. (Eds.). (2019). *Nye lærer- og lederroller i skolen*. Universitetsforlaget.
- Menekse, M. (2015). Computer science teacher professional development in the United States: a review of studies published between 2004 and 2014. *Computer Science Education*, 25(4), 325-350. <https://doi.org/10.1080/08993408.2015.1111645>
- Mishra, P., & Koehler, M. J. (2006, juni). Technological Pedagogical Content Knowledge: A Framework for Teacher Knowledge. *Teachers College Record*, 108(6), 1017-1054. DOI:10.1111/j.1467-9620.2006.00684.x
- Monk, S. (2017). *Programming the BBC Micro:bit: Getting Started with MicroPython*. McGraw-Hill Education.
- Moreau, H. N. (2021, desember 30). Lærere trenger hjelp til å knekke koden på koding. *NRK*. <https://www.nrk.no/innlandet/laerere-trenger-hjelp-til-a-knekke-koden-pa-koding-1.15781343>
- Neuman, W. L. (2014). *Social Research Methods: Qualitative and Quantitative Approaches* (7th ed.). Pearson Education Limited.
- Nilsen, N. O. (2010). Rektors ledelseskompetanse - hva skal til for å utvikle en digitalt kompetent skole? In R.-A. Andreassen, E. J. Irgens, & E. M. Skaalvik (Eds.), *Kompetent skoleledelse* (pp. 107-122). Tapir akademisk forlag.
- NordNordWest. (2012, oktober 2). *Norway Nordland - Herøy* [English: Location of Herøy in Nordland County in Norway.] [Bilde]. Wikimedia Commons. Hentet mai 05, 2022, fra https://commons.wikimedia.org/wiki/File:Norway_Nordland_-_Her%C3%B8y.svg
- Ohashi, Y. (2017). Preparedness of Japan's Elementary School Teachers for the Introduction of Computer Programming Education. In A. Hellas & V. Dagiene (Eds.), *Informatics in Schools: Focus on Learning Programming: 10th International Conference on Informatics in Schools: Situation, Evolution, and Perspectives, ISSEP 2017, Helsinki, Finland, November 13-15, 2017, Proceedings* (s.129-140). Springer International Publishing. <https://ebookcentral-proquest-com.ezproxy.nord.no/lib/nord/detail.action?docID=6304970>
- Oldendick, R. W. (2012). Survey Research Ethics. In L. Gideon (Ed.), *Handbook of Survey Methodology for the Social Sciences* (pp. 23-35). Springer New York. 10.1007/978-1-4614-3876-2_3
- Opplæringslova. (1998). *Lov om grunnskolen og den videregående opplæringa (LOV-1998-07-17-6)*. <https://lovdata.no/dokument/NL/lov/1998-07-17-61>

- Piedade, J., Dorotea, N., Pedro, A., & Matos, J. F. (2020, august 20). On Teaching Programming Fundamentals and Computational Thinking with Educational Robotics: A Didactic Experience with Pre-Service Teachers. *Education Sciences*, 10(9). <https://doi.org/10.3390/educsci10090214>
- Postholm, M. B. (2011). *Kvalitativ metode: en innføring med fokus på fenomenologi, etnografi og kasusstudier* (2nd ed.). Universitetsforlaget.
- Rich, P. (2017, januar). Computing and Engineering in Elementary School: The Effect of Year-long Training on Elementary Teacher Self-efficacy and Beliefs About Teaching Computing and Engineering. *International Journal of Computer Science Education in Schools*, 1(1). DOI:10.21585/ijcses.v1i1.6
- Rouhani, M., Farshchain, V., & Divitini, M. (2020). Teaching Programming in Secondary Schools: Stepping and Stumbling Stones. *Interaction Design and Architecture(s) Journal - IxD&A*, 47, 48-68. <https://doi.org/10.55612/s-5002-047-003>
- Sands, P., Yadav, A., & Good, J. (2018). Computational Thinking in K-12: In-service Teacher Perceptions of Computational Thinking. In M. S. Khine (Ed.), *Computational Thinking in the STEM Disciplines: Foundations and Research Highlights* (s.151-164). Springer International Publishing. https://doi.org/10.1007/978-3-319-93566-9_8
- Sanne, A., Berge, O., Bungum, B., Jørgensen, E. C., Kluge, A., Kristensen, T. E., Mørken, K. M., Svorkmo, A.-G., & Voll, L. O. (2016). *Teknologi og programmering for alle*. Utdanningsdirektoratet. Hentet mai 15, 2022, fra <https://www.udir.no/tall-og-forskning/finn-forskning/rapporter/teknologi-og-programmering-for-alle/>
- Sentance, S., & Csizmadia, A. (2017). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies*, 22, 469-495. <https://doi.org/10.1007/s10639-016-9482-0>
- Siraj, A. E. (2013, februar 09). *A quest for depth and breadth of insight through combination of Positivism and Phenomenology*. SlideShare. Hentet januar 02, 2022, fra <https://www.slideshare.net/AwaisSiraj/a-quest-for-depth-and-breadth-of-insight-through-combination-of-positivism-and-phenomenology>
- Sisman, B., & Kucuk, S. (2018, oktober). Pre-Service Teachers' Experiences in Learning Robotics Design and Programming. *Informatics in Education*, 17(2), 301-320. DOI:10.15388/infedu.2018.16
- Statistisk sentralbyrå. (2021, juni 22). *Ansatte i elevrettet arbeid i grunnskolen, etter kjønn, statistikkvariabel og år*. Statistisk sentralbyrå. Hentet mai 15, 2022, fra <https://www.ssb.no/statbank/table/12707/>

- Stenlund, E. (2021, mai). *Programmering og Fagfornyelsen* [Masteroppgave]. Universitetet i Oslo. Hentet mai 15, 2022, fra https://www.duo.uio.no/bitstream/handle/10852/87187/1/Masteroppgave_Erlend_Stenlund.pdf
- Utdanningsdirektoratet. (2019). *Algoritmisk tenkning* [Bilde]. Utdanningsdirektoratet. <https://www.udir.no/kvalitet-og-kompetanse/profesjonsfaglig-digital-kompetanse/algoritmisk-tenkning/>
- Utdanningsdirektoratet. (2021). *Slik ble læreplanene utviklet*. Utdanningsdirektoratet. Hentet mai 15, 2022, fra <https://www.udir.no/laring-og-trivsel/lareplanverket/fagfornyelsen/slik-ble-lareplanene-utviklet/>
- Utdanningsdirektoratet. (2022). *Studietilbud - Videreutdanning*. Utdanningsdirektoratet. Hentet mai 15, 2022, fra <https://www.udir.no/kvalitet-og-kompetanse/etter-og-videreutdanning/studietilbud/>
- Vinnervik, P. (2022). Implementing programming in school mathematics and technology: teachers' intrinsic and extrinsic challenges. *International Journal of Technology and Design Education*, 32, 213-242. <https://doi.org/10.1007/s10798-020-09602-0>
- Woodrow, J. E. J. (1992). The Influence of Programming Training on the Computer Literacy and Attitudes of Preservice Teachers. *Journal of Research on Computing in Education*, 25(2), 200-213. <https://doi.org/10.1080/08886504.1992.10782044>
- Zukanovic, H. (2021). *Hva kjennetegner matematikklæreres holdninger rundt undervisning av programmering i teoretisk matematikk?* [Masteroppgave]. Norges miljø- og biovitenskapelige universitet. Hentet mai 15, 2022, fra https://nmbu.brage.unit.no/nmbu-xmlui/bitstream/handle/11250/2786405/Masteroppgaven_signert.pdf?sequence=1