

TEKNOLOGIFORSTÅELSE

9. KLASSE SOM FAG

1. forløb

KLASSENS APP

Udarbejdet af Malene Erkmann, Brian Ravnborg Christensen, Malte von Sehested og Mikkel Hjort*

*Materialet er udviklet af Københavns Professionshøjskole, Professionshøjskolen UCN, VIA University College samt læremiddel.dk for Børne- og Undervisningsministeriet under rammerne for Forsøg med teknologiforståelse i folkeskolens obligatoriske undervisning. Læs mere om forsøget på www.tekforsøget.dk og www.emu.dk.



KØBENHAVNS
PROFESSIONS
HØJSKOLE

KP

LÆRE
MIDDEL
DK



VIA University
College

UCN

RAMBOLL

INDHOLDSFORTEGNELSE

1.	FORLØBSBESKRIVELSE	3
1.1	BESKRIVELSE	4
1.2	RAMMER OG PRAKTISKE FORHOLD	4
2.	MÅL OG FAGLIGE BEGREBER	6
3.	FORLØBSNÆR DEL	11
3.1	INTROFASE: FORFORSTÅELSE OG KOMPETENCER	14
3.2	UDFORDRINGS- OG KONSTRUKTIONSFASE	15
3.3	OUTROFASE: NY FORSTÅELSE OG NYE KOMPETENCER	20
4.	PERSPEKTIVERING	21
4.1	EVALUERING	21
4.2	PROGRESSION	21
4.3	DIFFERENTIERINGSMULIGHEDER	21

Version 2

Dette er version 2 af forløbet. I revisionen af forløbene har vi arbejdet med at præcisere mål, rammer og aktiviteter. Der er ikke ændret fundamentalt ved forløbet, så materialer, som er udviklet til den konkrete undervisning på skolerne på baggrund af den første version af forløbet, vil stadig kunne anvendes.

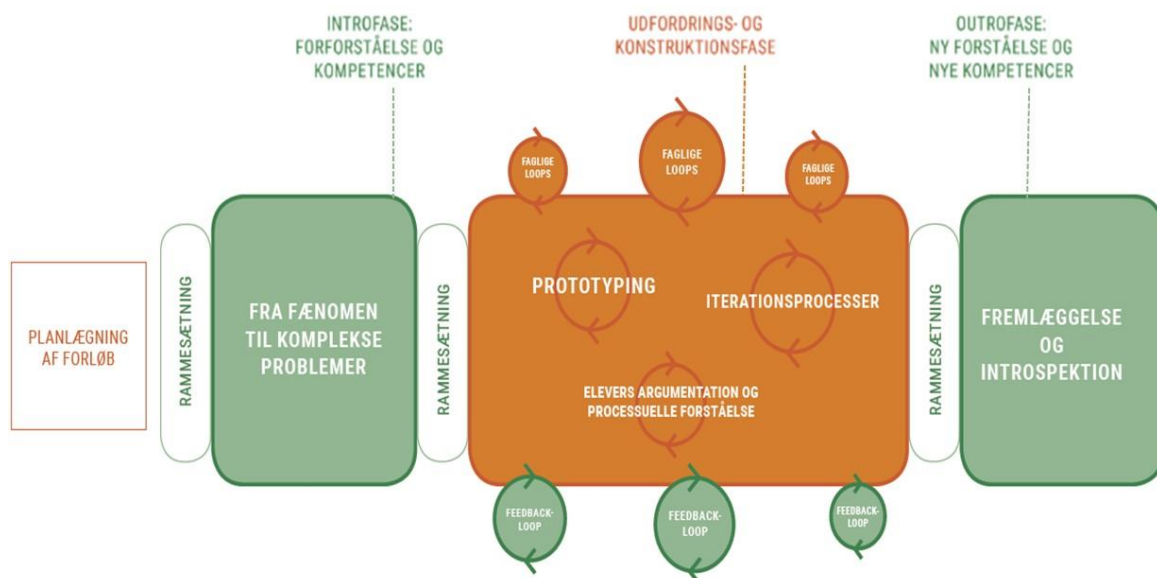
Vær opmærksom på at du altid selv skal sikre dig, at databeskyttelsesforordningen (GDPR) bliver overholdt i arbejdet med den konkrete teknologi eller internet-tjeneste i prototypen. Prototyperne er skabt med afsæt i et princip om, at eleverne ikke må dele personlig information med gratis teknologier. Det er dog i hvert tilfælde nødvendigt at tage konkret stilling til, hvordan teknologien eller tjenesten anvendes i tilrettelæggelsen af den konkrete undervisning. Undersøg altid om teknologien kan tilgås via unilogin eller anden sikker undervisningsadgang.

1. Forløbsbeskrivelse

Kameraapps, digitale notesbøger, online spil. De fleste af de apps, vi betjener os af på daglig basis, gemmer samtidig vores personlige data. Kameraappen gemmer fx oplysninger om, hvor vi var, da billedet blev taget. Den digitale notesbog registrerer, hvornår vi arbejder, og online spillet holder styr på vores point og placeringer. Overalt gemmes data om os og vores bevægelser på nettet og i den fysiske verden. Gennem dette forløb vil eleverne selv arbejde med at designe og producere en simpel app, der samtidig demonstrerer, hvordan digitale artefakter bygger på en tre-lags arkitektur, bestående af det grafiske lag, kodningen og den bagvedliggende database. Forløbet vil samtidig give eleverne mulighed for at stifte nærmere bekendtskab med tekstprogrammering og præsentere opgaver, der kræver modifikation af koderne bag blokprogrammeringen, der præsenteres i opgaveeksemplerne. Forløbet har en bred tidsramme, da det bygger på et forløb med app-produktion fra mellemtrinnet. Nogle elever vil derfor kunne bygge videre på deres erfaringer fra dette forløb, mens andre må starte forfra.

Forløbet er bygget op over det didaktiske format for prototyperne med en introducerende del (hvor eleverne præsenteres for forskellige apps, der bygger på data og databaser), en mere undersøgende/eksperimenterende del (hvor eleverne designer og programmerer en app) og en outro-del med fremlæggelser opsamlings og evalueringer, se figur 1. Der lægges i forløbet op til at arbejde med programmet App Lab, hvor eleverne kan designe og programmere en app, både i blokprogrammering og tekstprogrammering, og samtidig integrere en database.

Figur 1: Forløbsmodel for prototyperne



1.1 Beskrivelse

Forløbets eksempel bygger på, at eleverne skal arbejde med design og programmering i værktøjet Applap. Programmering finder sted i JavaScript, men kan i App Lab både ske gennem blokprogrammering og gennem tekstprogrammering. Der er, som beskrevet i indledningen i denne opgave, særligt fokus på at arbejde med databaser og samtidig kigge nærmere på den egentlige tekstprogrammering bag værktøjets blokprogrammering. Det er altså ikke tanken, at eleverne skal programmere hele appen direkte i JavaScript, men der vil være elementer i programmeringen, hvor eleverne har brug for at redigere eller tilføje koder direkte i JavaScript.

Der vil derfor i forløbet særligt være fokus på programmeringskompetencer og elementer som loops, modellering og navngivning af elementer, udover at forløbet lægger op til kritisk analyse og vurdering af apps i almindelighed, som samler og på forskellig vis behandler vores personlige data.

Eleverne vil i første del af forløbet blive præsenteret for forskellige digitale artefakter, der benytter databaser på forskellige måder. I denne fase supplerer eleverne med egne erfaringer, og deres forståelse for databasernes betydning i digitale artefakter udvides generelt.

Herefter gennemføres en designproces, hvor eleverne skitserer deres egen app, klassens app, og kommer med forslag til, hvilke data deres app med fordel kunne bygge på.

Endelig arbejdes med at producere et bud på klassens app. I materialet findes et eksempel, som eleverne kan bruge som inspiration eller vælge at redesigne.

Produkt

Eleverne producerer som afslutning på forløbet "klassens app". En kvalitet ved klassens app er, at den bygger på data, der hentes fra den integrerede database. Det kan fx være en database med loginoplysninger, en quiz eller en simpel *Simon-says* funktion, der integreres i design af appen.

Simon says: En børneleg, i stil med kongens efterfølgere, hvor en person er *Simon*, der – med ordene "Simon siger..." - fortæller resten af gruppen, hvad de skal gøre. Aktiviteterne er typisk fysiske aktiviteter, som fx at hoppe på et ben eller løbe tre gange rundt om et træ. Så længe man følger Simon, er man med i legen. Hvis man ikke længere kan følge med, er man ude. Legen findes i digitale versioner, hvor programmet er Simon og vælger tilfældige opgaver fra en database, som gruppen skal udføre.

1.2 Rammer og praktiske forhold

Der lægges i dette forløb op til, at der arbejdes med programmet App Lab. App Lab er et programmeringsværktøj, der gør det muligt at designe og programmere en simpel app ved hjælp af blokprogrammering eller direkte i JavaScript. Værktøjet stilles til rådighed af organisationen Code.org, der står bag en række non-profit initiativer med fokus på at styrke brugernes programmeringskompetencer.

1.2.1 Samlet varighed

Forløbet er anslået til at dække 20-40 lektioner – afhængigt af brugen af faglige loops samt valg af funktioner i app'en. Forløbet kan med andre ord skaleres både op og ned og på den måde tilpasses undervisningen i øvrigt.

1.2.2 Materialer

Konkrete elev- og lærerressourcer i ressourcebanken til forløbet findes på www.tekforsøget.dk/forlob.

Digitale teknologier:

- App Lab: <https://studio.code.org/home>

Elevehenvendte ressourcer:

- Eksempel på opbygning af database og kodning
- Oplæg til teknologianalyse og designproces.

Lærerhenvendte:

- Udkast til PowerPoint om *Personlige data – dine rettigheder*
- Udkast til PowerPoint om *Data i dine apps*
- Intro til App Lab: <https://studio.code.org/s/applab-intro/stage/1/puzzle/1>
- Eksempel på design af app

1.2.3 Lokaler

Ingen særlige krav.

1.2.4 Videnspersoner og andre eksterne aktører

Besøg af lokale aktører, der arbejder med produkter, der håndterer andres personlige data.

1.2.5 Tværfaglighed

Arbejdet med databaser kan med fordel indgå i et samarbejde med matematik eller samfundsfag.

2. Mål og faglige begreber

Fokus i dette forløb er på teknologisk handleevne, hvor der fx også arbejdes med, at eleverne skal læse og forstå programmer skrevet i et tekstbaseret programmeringssprog. Med deres viden om data og databaser skal eleverne producere en app, der retter sig mod dem selv som målgruppe, men til gengæld trækker på deres analyse af andre apps, de selv benytter. Eleverne arbejder med digital myndiggørelse, når de skal teste deres prototype ved hjælp af en tænk-højt-test og efterfølgende justere deres design.

Gennem dette forløb arbejdes med centrale og komplekse problemer omkring data og databehandling, og der lægges op til løbende kritisk refleksion over egen og andres håndtering af dine og mine data.

KOMPETENCE OMRÅDER	DIGITAL MYNDIGGØRELSE	DIGITAL DESIGN OG DESIGNPROCESSER	COMPUTATIONEL TANKEGANG	TEKNOLOGISK HANDLEEVNE
Kompetencemål (efter 9. klasstrin)	Eleven kan handle med dømmekraft i komplekse situationer, der vedrører digitale artefakters betydning for individ, fællesskab og samfund.	Eleven kan tilrettelægge og gennemføre iterative designprocesser og skabe digitale artefakter, der løser komplekse problemstillinger, relevante for individ, fællesskab og samfund.	Eleven kan reflektere over og anvende computationel tankegang på problemstillinger fra omverdenen.	Eleven kan vurdere, vælge og på kvalificeret vis anvende digitale teknologier i autentiske situationer.
Færdigheds- og vidensmål (efter 9. klasstrin)	<p>Brugsstudier</p> <ul style="list-style-type: none"> Eleven kan gennemføre enkle undersøgelser af brugeres perspektiver på og anvendelse af digitale artefakter Eleven har viden om undersøgelsesmetoder, der kan anvendes til at forstå brugeres perspektiver på og anvendelse af digitale artefakter <p>Redesign</p> <ul style="list-style-type: none"> Eleven kan, på baggrund af kritisk analyse og vurdering, udvikle konkrete forslag til redesign af digitale artefakter og de situationer, artefaktet indgår i Eleven har viden om egne handlemuligheder i forhold til digitale artefakters betydning i samfundet 	<p>Idegenerering</p> <ul style="list-style-type: none"> Eleven kan generere, udvælge og kvalificere ideer, der imødekommer en problemstilling Eleven har viden om metoder og teknikker til divergent og konvergent tænkning, idegenerering og eksternalisering af ideer 	<p>Strukturering</p> <ul style="list-style-type: none"> Eleven kan strukturere fænomener og begreber i et problemfelt og i computationelle modeller Eleven har viden om principper for abstraktion og strukturering af et problemfelt samt fundamentale teknikker til strukturering af data og processer 	<p>Programmering</p> <ul style="list-style-type: none"> Eleven kan læse og forstå programmer skrevet i et tekstbaseret programmeringssprog samt anvende et sådant til systematisk modifikation og konstruktion af programmer ud fra en problemspecifikation Eleven har viden om metoder til at analysere og forudsige programmers opførsel samt teknikker til systematisk og trinvis udvikling af programmer

Konkretiserede læringsmål

- Eleven kan gennem en designproces designe en app til en særlig målgruppe.
- Eleven kan brainstorme på en masse ideer og udvælge en brugbar ide.
- Eleven kan gennem arbejdet med en konkret app få indsigt i, hvordan en app ikke bare har til formål at løse brugerens behov, men fx også at den sælger data videre.
- Eleven kan oprette simple databaser samt importere og strukturere data.

Centrale (teknologi)faglige begreber

Fagligt begreb	beskrivelse
App	En app kan beskrives som et lille softwareprogram, som du kan downloade på din digitale enhed.
Argumentation	Argumentation betegner de begrundelser, der kan gives i forhold til valg og fravalg i en designproces. I designprocessen kan eleven argumentere for sine valg med henvisning til den viden, der er skabt i designprocessen gennem egne eller andres undersøgelser. Eleven kan som eksempel argumentere for et design gennem reference til elementer i deres egen undersøgelse af en eksisterende brugspraksis eller til anerkendte kriterier og konventioner for godt design.
Brugspraksis	Brugspraksis er, sammenlignet med brug, en mere generel beskrivelse af de aktiviteter, vaner, adfærd og handlinger, som fremkommer når et digitalt artefakt har været i anvendelse over en længere periode. Det er vigtigt at forstå brugspraksis for at kunne foretage konsekvensanalyser og vurdere, hvad et givet digitalt artefakt betyder for individet, fællesskabet eller samfundet. Viden om brugspraksis opnås som regel gennem brugsstudier.
Brugsstudier	Brugsstudier betegner de undersøgelser, hvor der tilvejebringes viden om den specifikke brug af digitale artefakter eller om den generelle brugspraksis. Brugerstudier vil omfatte indsamling og analyse af data, der belyser et digitalt artefakts betydning og brug. Dataindsamlingen vil ofte indbefatte observationer af, hvordan et digitalt design bruges, samt interviews, der søger at afdække oplevelsen af det digitale artefakt og dets betydning.
Data	Data er "enhver repræsentation af fakta eller ideer på en formaliseret måde, som kan kommunikeres eller manipuleres ved en eller anden proces" (Peter Naurs definition). Et banalt eksempel er en persons navn og alder, som simpelt kan repræsenteres i form af en tekststreng og et tal – i modsætning til udsagn om, at en person er venlig eller pæn. I takt med at man ønsker at repræsentere flere og flere aspekter af virkeligheden i digitale modeller, er man tvunget til at formalisere disse aspekter af virkeligheden som data. Det åbner for slagkraftige analyser (data science), men også for "overformalisering" hvor virkeligheden struktureres mere end godt er. Dette er et vigtigt aspekt af computationel modellering.

Database	Et database-system gør det muligt at gemme en større mængde af information på en form, så informationen er maskinel omkostningslet at tilgå og opdatere. I modsætning til data, der er gemt i en fil, kan flere computerprogrammer opdatere de samme data uden information tabes på grund af overskrivninger.
Digital artefakt	Digitale artefakter betegner en af mennesket tilvejebragt genstand, som indeholder et væsentligt element af digital teknologi. Til forskel fra betegnelsen digital teknologi, betoner betegnelsen digitalt artefakt de produktkvaliteter, der er blevet til gennem design og programmering, hvorved intentionalitet og formål er blevet indlejret i artefaktet. En app, en programmeret robot, en simulering af fotosyntese eller en programmeret micro:bit indlejret i et fysisk artefakt er eksempler på digitale artefakter.
Eksternalisering	Eksternalisering betegner den proces, hvor en idé eller en tanke, gives fysisk form eller på anden måde gøres tilgængelig og konkret for andre. I design vil eksternaliseringer ofte være modeller, skitser eller mock-ups af et digitalt artefakt, der udarbejdes i tilgængelige materialer, for eksempel papir, træ eller i digitale omgivelser som App Lab. Disse skabes med henblik på at kommunikere ideer til andre og modtage feedback på disse idéer.
Idegenerering	Idégenerering omhandler systematisk behandling af viden med henblik på at skabe løsningsforslag, der gennem eksternalisering gøres til genstand for kollektiv bearbejdning og vurdering. Med idégenerering giver eleverne specifikke svar på en problemstilling. Dette kan finde sted på mange tidspunkter i en designproces, men vil som oftest bygge på elevens undersøgelser. I idégenereringen kan eleverne eksempelvis skitsere designidéer, bygge dem i pap og papir eller skrive scenarier, der angiver en måde, hvorpå et fremtidigt digitalt artefakt kan bringes i anvendelse.
Inputteknologi	Inputteknologi er en samlet betegnelse for de dele af et digitalt artefakt, som lader brugeren interagere med artefaktet. Det kan eksempelvis være et tastatur, et kamera eller en sensor, som kan registrere et input fra en bruger og omsætte det til en handling i en computer. Det kan være vigtigt at forholde sig til inputteknologier i forhold til dels egne digitale artefakter og dels i forhold til analytisk at vurdere andres digitale artefakter.
Konstruktion	Konstruktion omhandler den aktivitet, hvor ideer finder udtryk i et konkret digitalt artefakt, som kan gøres til genstand for en efterprøvnig af form, funktion og interaktion. Konstruktion i digitale teknologier rummer aktiviteter såsom computationel tankegang, valg af programmeringssprog, programmering, design af grænseflade, konstruktion af en tidlig papirmodel eller diagram over det fremtidige artefakt, at bygge den fysiske udformning af et digitalt artefakt og iterative evalueringer af digitale artefakter under udvikling.

Konvergent tænkning	Konvergent tænkning betyder – modsat divergent tænkning – at indsnævre eller dedikere sine tankeprocesser i en særlig retning. I designprocessen omfatter konvergent tænkning hvor eleverne fravælger mulige løsninger eller frasorterer designidéer og dermed fokuserer arbejdet i en særlig retning.
Løkker eller loop	Løkker er et centralt begreb indenfor programmering. Grundlæggende er en løkke en sekvens af instruktioner der gentages indtil en betingelse er opfyldt. I tekstbaseret programmering er der normal 2 slags løkker; en for-løkke og en while-løkke
Modellering	Computational modellering af et problemfelt, der har til formål at frembringe abstrakte modeller af data og dataprocesser (for eksempel algoritmer samt data- og klassemodeller, der udarbejdes som mellemformer på vejen til realisering af et digitalt artefakt) eller konkrete computationelle modeller realiseret i et programmeringssprog – for eksempel AgentCubes eller NetLogo – hvor man simulerer et system ved at lave simple regler for de enkelte komponenters adfærd. Den form for computational modellering kaldes agentbaseret modellering.
Outputteknologi	Outputteknologi er en samlet betegnelse for de dele af et digitalt artefakt, som giver brugeren feedback på en interaktion med artefaktet. Det kan eksempelvis være en skærm (visuel feedback), en højttaler (auditiv feedback) eller en vibration (taktile feedback).

3. Forløbsnær del

Fase (jf. model)	Aktivitet	Læringsmål
Intro/rammesætning 6-8 lektioner		
Rammesætning 1 lektion	<ul style="list-style-type: none"> ■ Eleverne præsenteres for eksempler på app, der bygger på databaser. 	<ul style="list-style-type: none"> ■ Eleverne får viden om datas betydning for apps vi omgiver os med.
Fra fænomen til komplekse problemstillinger (Feltundersøgelse) 4 lektioner	<ul style="list-style-type: none"> ■ Elevernes viden om regneark og databaser eksternaliseres. ■ Eleverne bygger eksempel på brugerdatabase i Excel. ■ Eleverne supplerer selv med eksempler på apps, de selv bruger, der benytter databaser. ■ Eleverne gennemfører en teknologianalyse med fokus på databaser. 	<ul style="list-style-type: none"> ■ Elevernes får repeteret centrale begreber som kolonner, rækker og datasæt mm. ■ Eleverne får færdigheder til at genkende brugerflader, der henter eller sender data til en bagvedliggende database.
Idégenerering 1 lektion	<ul style="list-style-type: none"> ■ Med udgangspunkt i elevernes erfaring med brug af eksisterende apps, brainstormes nu på en klasseapp, der kan samle relevante funktioner fra andre apps og måske tilføjes helt nye. 	<ul style="list-style-type: none"> ■ Elevernes kan gennemføre en designproces i fællesskab.
Fagligt loop 2 lektioner	<ul style="list-style-type: none"> ■ Konstruktion af databaser 	
Udfordrings- og konstruktionsfase Idegenerering 17-29 lektioner		

Argumentation og introspektion: Idéudvælgelse 1 lektion	<ul style="list-style-type: none"> ■ Brainstorm i grupper (se ressource-ark) ■ Udvalgelse af ideer 	<ul style="list-style-type: none"> ■ Eleven kan udtrykke en idé med konstruktion af et digitalt produkt. ■ Eleven kan formgive et digitalt produkt ud fra egne undersøgelser. ■ Eleven får viden om fagbegreber for argumentation i en designproces. ■ Eleven får færdigheder i at vurdere en løsning af et komplekst designprodukt ud fra flere parametre.
Prototyping 2 lektioner	<ul style="list-style-type: none"> ■ Prototype på papir ■ Visualisering af processer 	<ul style="list-style-type: none"> ■ Eleven kan konstruere en 3D form efter fastsatte principper.
Testning af prototype 1 1 lektion	<ul style="list-style-type: none"> ■ Tænke-højt-test <p>Se ressourceark</p>	<ul style="list-style-type: none"> ■ Eleven får viden om iterative processer. ■ Eleven får færdigheder til at analysere en designløsning.
Design og produktionsfase 10 lektioner	<ul style="list-style-type: none"> ■ Design i App Lab ■ Programmering i App Lab 	<ul style="list-style-type: none"> ■ Eleven får viden om programmering og strukturere in af databaser i forbindelse med programmering.
Testning af prototype 2 1 lektion	<ul style="list-style-type: none"> ■ Test 	<ul style="list-style-type: none"> ■ Eleven øver kompetencer i at sætte sig i brugerens sted og vurdere egen produktion med andre øjne.
Tilpasning af prototype 2 lektioner	<ul style="list-style-type: none"> ■ Tilpas på baggrund af test 	<ul style="list-style-type: none"> ■ Eleven øver kompetencer i at arbejde i iterationer, hvor produktet hele tiden forbedres.
Faglige loop 12 lektioner	<ul style="list-style-type: none"> ■ Designfunktioner i App Lab ■ Blokprogrammering i App Lab ■ Databaser i App Lab ■ Visning af data fra databaser i app design ■ Miniforløb om håndtering af andres personlige data 	

Feedback-loop	<ul style="list-style-type: none"> ■ Hvordan tror I databasen bag jeres udvalg af app ser ud? ■ Hvor længe tror I brugernes personlige data må ligge i databasen? ■ Har I sikret jer, at alle brugere er unikke? Fx via e-mail? ■ Kan den samme fysiske bruger optræde flere gange som bruger i jeres database? 	<ul style="list-style-type: none"> ■ Eleverne får viden om databaser og viden om opbevaring og brug af personlige data i database.
Outrofase: Ny forståelse og nye kompetencer 3 lektion		
Præsentation og introspektion 3 lektioner	<ul style="list-style-type: none"> ■ App'ens begrænsninger og muligheder som eksemplarisk artefakt, der kan fortælle noget om data og databaser i teknologier, der omgiver os. ■ Fremlæggelse af produkt og diskussion i klassen, med udgangspunkt i prototyperne, om data i digitale teknologier i vores hverdag. 	<ul style="list-style-type: none"> ■ Eleven får viden om data og mulighederne for at opbevare og udnytte data i digitale produktioner. ■ Eleven får viden om kritiske aspekter ved udlevering og opbevaring af personlige data.

3.1 Introfase: Forforståelse og kompetencer

Introfasen rammesætter forløbet for eleverne. I dette afsnit gennemgås de forskellige faser:

- Eleverne præsenteres for eksempler på app, der bygger på databaser.
- Elevernes viden om regneark og databaser eksternaliseres.
- Eleverne bygger eksempel på brugerdatabase i Excel (tværfagligt samarbejde med matematik).
- Eleverne supplerer med eksempler på apps, de selv bruger, der benytter databaser.
- Eleverne gennemfører en teknologianalyse med fokus på databaser.
- Med udgangspunkt i elevernes erfaring med brug af eksisterende apps, brainstormes nu på en klasseapp, der kan samle relevante funktioner fra andre apps og måske tilføje helt nye.

3.1.1 Varighed

6-8 lektioner

3.1.2 Problemfelt

Helt centralt i digitale artefakter er brugen af data – ikke mindst personlige data. Vi har stadig kun set begyndelsen af, hvad brug og misbrug af personlige data kan få af betydning for personer, firmaer og demokratiske systemer. Men det er tydeligt, at der i den aktuelle håndtering af digitale data åbnes for helt nye muligheder for at bruge data intelligent og skabe mere sikre liv for os alle, men samtidig risiko for, at fremmede magter eller diktaturer kan misbruge disse digitale data, der som regel er tilgængelige online.

3.1.3 Problemstilling

Med udgangspunkt i problemfeltet kan problemstillinger i dette forløb indeholde spørgsmål på forskellige niveauer:

Beskrivende

- Hvad er data?
- Hvad kendetegner personlige data?

Forklarende/analyserende

- Hvorfor er det vigtigt, at vores personlige data beskyttes?
- Hvem kan have interesse i at få adgang til vores personlige data?
- Hvilke konsekvenser kan læk af vores personlige data have?

Vurderende

- Hvilken betydning har det for vores liv, at flere og flere programmer gemmer vores data?
- Hvilken betydning kan det have for vores liv, når flere og flere programmer benytter fx Facebooks database og på den måde skaber et fælles login til mange tjenester?

3.1.4 Iscenesættelse/scenarie

Forløbet starter med en intro om data og apps, der bruger data. Eleverne præsenteres for eksempler på apps og bidrager med egne eksempler.

Lærerens oplæg kan fx bygge på følgende eksempler:

- Aula
- Fortnite
- Facebook

Der kan i de tre eksempler skelnes mellem data, vi selv (eller vores forældre) giver systemet, og hvilke data programmet indsamler, mens vi bruger det. Tag fx udgangspunkt i PowerPointen *Data i dine apps*. Lad herefter eleverne supplere med egne eksempler. Herefter udvælger eleverne i grupper et eksempel på en app, som de analyserer nærmere med henblik på at kunne forklare:

- Hvilke data giver brugeren til app'en?
- Hvilke data indsamler app'en om brugeren?

Analysen gennemføres i tre trin:

1. Afprøv app'en.
2. Læs beskrivelsen af app'en og firmaets datapolitik.
3. Søg på nettet for at finde andres anmeldelse af app'en.

3.1.5 Faglige loops

■ Konstruktion af databaser

For at kunne forstå og arbejde med databaser, kunne et fagligt loop med udgangspunkt i Excel være relevant. Dette loop kunne med fordel laves i samarbejde med faget Matematik, fx indenfor kompetenceområdet *statistik og sandsynlighed*. Selvom Excel ikke er et egentligt databasesystem som fx SQL, så giver det eleverne en god fornemmelse af, hvordan databaser grundlæggende fungerer.

Der er inspiration at hente for læreren i følgende to film:

- Som tabeller i Excel: https://youtu.be/8GcLq9y__BA
- I Database management systems. What is Database & SQL?: <https://youtu.be/FR4QleZaPeM> (engelsksproget animation)

■ Opbevaring af personlige data. Tag fx udgangspunkt i ressourcen, som indeholder en PowerPoint: *Personlige data – dine rettigheder*, der lægger op til, at eleverne laver deres egen quiz om deres rettigheder i forhold til andres håndtering af deres personlige data.

3.2 Udfordrings- og konstruktionsfase

I udfordrings- og konstruktionsfasen skal eleverne producere deres egen app. Processen gennemløber følgende faser:

- Brainstorm i grupper (se ressource-ark)

- Udvælgelse af ideer
- Prototype på papir: <http://ungkom.dk/prototyper-en-begynderguide/> Visualisering af processer
- Tænke-højt-test (se ressource-ark)
- Design i App Lab se <https://studio.code.org/s/applab-intro/> samt under faglige loop), herunder programmering i App Lab, design i App Lab og strukturering af databaser i App Lab

3.2.1 Varighed

17-29 lektioner

3.2.2 Konkret(e) udfordring(er)

I dette forløb skal eleverne udvikle en app i programmet App Lab. App'en udvikles som klassens egen app, hvor kun medlemmer af klassen må være brugere. Eleverne skal selv finde et problem eller behov i deres hverdag, som app'en imødekommer. En opgaveformulering kunne f.eks. være:

I skal lave en app til jeres klasse. App'en skal tage udgangspunkt i noget, I allerede gør sammen, men som kan forbedres ved at bruge en app. Det kan være, at I allerede bruger en app, men at den ikke er optimal, eller det kan være, I ikke tidligere har tænkt på at bruge en app til denne funktion.

Det er en meget bredt formuleret opgave, og derfor skal eleverne trække på deres erfaring med at rammesætte sådanne opgaver. Det kan fx være, at eleverne vælger at lave en lektielæsningsapp, en app, der koordinerer online aktiviteter som fx spillesessioner i Fortnite, en bytte-app eller måske en app, der koordinerer offline aktiviteter som f.eks. træning eller dans, hvor brugerne kan se hinanden, mens de træner eller danser sammen i hver deres fysiske lokation.

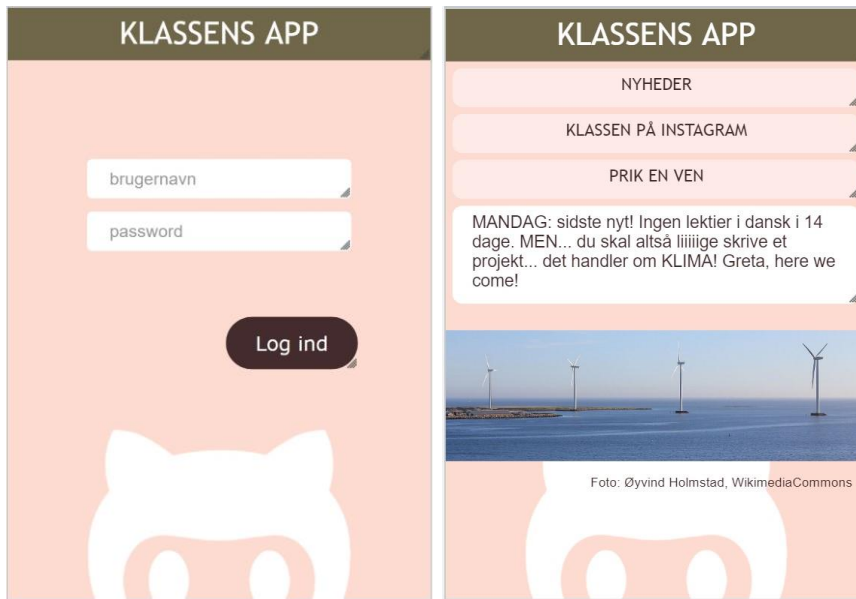
Der arbejdes i dette forløb med en app med brugerkartotek. Hermed kommer eleverne både til at arbejde med dataetiske problemstillinger og konkrete design-, strukturings- og programmeringsopgaver.

I et fagligt loop arbejdes med GDPR og personlige data. I andre loops arbejdes med modellering og elementer i App Lab.

3.2.2.1 Om klassens app

Herunder vises et eksempel på en simpel løsning af opgaven. I eksemplet består klassens app af to sider, en loginside og en indholdsside (præsentationslaget). Datalaget består af en simpel, struktureret database med brugernes brugernavn og deres password. Det logiske lag, koden, fremgår af den sidste illustration. Koden er i eksemplet primært lavet som blokprogrammering, med enkelte tilpasninger i selve JavaScript koden.

Klassens app består af minimum to sider (screens): en loginside og en forside:



Derudover består app'en af en database med brugere. Databasen opbevarer data i struktureret format som f.eks. en brugerdatabase. Når data opbevares i et struktureret format, kan det læses af forskellige computerprogrammer.

id	brugernavn	⚙	password	⚙	+	Actions
#	<input type="text" value="enter text"/>		<input type="text" value="enter text"/>			<button>Add Row</button>
1	"Smilla"		"SpiserOst"			<button>Edit</button> <button>Delete</button>
2	"Aya"		1234			<button>Edit</button> <button>Delete</button>
3	"Benjamin"		"footYESmine"			<button>Edit</button> <button>Delete</button>

Og endelig kode:

```

1  onEvent(▼"knapLogind", ▼"click", function() {
2  readRecords("brugere", {}, function(records) {
3  for (var i = 0; i < records.length; i++) {
4  if (getText(▼"text_brugernavn") == records[i].brugernavn && getText(▼"text_password") == records[i].password) {
5  setScreen(▼"forside");
6  }
7  }
8  });
9  });
10

```

Kritiske punkter:

- For-loop. Et for-loop eller en løkke kan være vanskelig at forstå. Lad fx eleverne arbejde med ressourcearket, hvor løkken er vist i JavaScript.
- Get record (records[i]).password. Denne del af koden indskrives direkte i felterne eller i JavaScript. Hvis koden ikke virker, kan det skyldes et ekstra eller manglende ")".
- Navngivning af tekstbokse, kolonner, variable. Navngivning er afgørende for, at de forskellige elementer kan hentes ind i koderne.
- Case Sensitive. I databasen er der forskel på store og små bogstaver.

3.2.2.2 Udvidelsesmuligheder

Eleverne kan i første omgang vælge at opbygge version 1 af deres app som eksemplet herover. Herefter kan de tilpasse design, database og endelig koden. Ideer til eleverne kunne fx være:

- Brugeren kunne få en meddelelse, hvis der ikke var tastet et brugernavn eller password ved login
- Brugeren kunne få en meddelelse, hvis der ikke var tastet korrekt brugernavn eller password ved login
- Appen kunne suppleres af et «orakel» (database, random svar) eller en anden tilfældighedsgenerator (se fx beskrivelse af *Simon Says* i afsnit 1.1)

3.2.3 Faglige loops

Alt efter elevernes erfaringer med programmet App Lab og arbejdet med databaser, kan undervisningen indeholde forskellige faglige loops.

- Opbevaring af personlige data: <https://www.datatilsynet.dk/generelt-om-databeskyttelse/hvad-er-dinerettigheder/> (se PowerPointen *Personlige data – dine rettigheder* til inspiration)
- Intro til App Lab: <https://studio.code.org/s/applab-intro/>
- For-loop/løkker. Lad fx eleverne se nærmere på eksemplet i ressourcearket.
- Strukturering af data (se PowerPointen *Strukturering af data* til inspiration)
- Databaser: What is Database & SQL?: <https://youtu.be/FR4QleZaPeM> (engelsksproget animation)

Yderligere inspiration til læreren kan findes her:

- Om at oprette brugere i App Lab: <https://youtu.be/nkj1cfWufoc>

- Modellering af log in proces. Bygger på denne: <https://youtu.be/hYxpRZxvUUY>
- Redesign med tjek af, om ny bruger er eksisterende bruger: <https://youtu.be/mcl0d8XrLOE>
- Oprette brugere i database: <https://youtu.be/nkj1cfWufoc>

3.2.4 Feedbackloops

- Analyse af apps
- Design af egen app
- Håndtering af personlige data

<p>Analyse af apps</p>	<p>I gruppernes fremlæggelse stilles to overordnede spørgsmål: Hvilke data giver brugeren til app'en? Hvilke data indsamler app'en om brugeren?</p> <p>Suppler fx med følgende spørgsmål:</p> <ul style="list-style-type: none"> • Kan man på forhånd se, hvilke data man skal afgive, for at bruge app'en? • Skal man måske afgive flere data undervejs? • Hvad betyder det for din oplevelse af app'en? • Er det tydeligt, hvilke data, app'en indsamler? • Er det tydeligt, hvordan app'en håndterer data? • Kan du forestille dig, hvordan disse data kunne misbruges?
<p>Design af egen app</p>	<p>I designprocessen arbejder eleverne i tre lag: præsentationslaget, databasen og logiklaget (koderne). I gruppernes præsentationer fokuseres især på datalaget.</p> <p>Spørg fx ind til:</p> <ul style="list-style-type: none"> • Hvilke andre oplysninger ville I kunne indsamle, hvis jeres database blev udbygget? • Ville flere data kunne give brugeren en bedre oplevelse? • Ville flere data kunne give jer flere personlige oplysninger om brugerne?
<p>Håndtering af personlige data</p>	<p>Hvis I har gennemgået loopet om retten til personlige data, kan præsentationen af elevernes app afsluttes med deres refleksioner i forhold til håndtering af data:</p> <p>Har I indsamlet personlige data om brugerne? Er data almindelige eller følsomme? Hvilke rettigheder har brugerne i forhold til de data, I indsamler om dem? Har I tænkt disse funktionaliteter ind i app'en?</p>

3.3 Outrofase: Ny forståelse og nye kompetencer

Forløbet afsluttes med fælles fremlæggelse af gruppernes bud på en version af *klassens app* (se også tidligere feedback-loop fra afsnit 3.2.4).

3.3.1 Varighed

3 lektioner

3.3.2 Fremlæggelse og introspektion

Eleverne præsenterer her deres koder, deres databaser og design af appens præsentationslag.

Med udgangspunkt i eksempler fra 3.2.2.1 lægges vægt på, hvordan gruppernes design adskiller sig fra det oprindelige. Eleverne kan fx:

- fokusere på særlige løsninger i det logiske lag, enten i form af forbedring af koden eller tilføjelser til den oprindelige kode.
- fokusere på udbygning af den strukturerede database.

Der lægges i elevernes præsentation vægt på deres evne til at abstrahere og fokusere deres fremlæggelse på relevante elementer.

Til sidst samles op på det samlede forløb. Eleverne samles i mindre grupper for at diskutere, om/hvordan arbejdet med design af klassens app har øget deres opmærksomhed på brugen af data i apps i almindelighed. Gruppens opsamling deles mundtligt eller samles i en kort video, hvor gruppen besvarer spørgsmålet: hvad kan klassens app fortælle om digitale artefakters indsamling af digitale data?

4. Perspektivering

4.1 Evaluering

Evaluering i forløbet foregår i forbindelse med de iterative processer og afprøvninger i form af feedforward på idéer til tilpasninger af klassens App. Her diskuteres, hvorvidt der er sammenhæng mellem problemstilling og løsning. Det er især feedbackloopsene der giver læreren mulighed for at sammenligne elevernes læring i forhold til de konkrete læringsmål.

Gennem vejledningen er der mulighed for indsigt i alle elevers proces og i gruppernes samarbejde. I den afsluttende evaluering lægges vægt på, om eleverne gennem arbejdet med at producere og analysere app, har fået større forståelse for, hvordan data indsamles, både når vi opretter os som brugere, og når vi bruger app'en i hverdagen.

Der kan efter gennemførelsen af forløbet sættes fokus på spørgsmål som:

- Hvilke observationer gjorde jeg mig om elevernes arbejde i den indledende proces med en særlig målgruppe?
- Hvad fungerede godt i gruppernes arbejde med iterative processer og brainstorms?
- Hvordan fungerede min egen rolle som vejleder undervejs i arbejdet?
- Hvordan støttede jeg om processen om at oprette simple databaser samt importere og strukturere data?
- Kan erfaringer fra dette forløb bruges i fremadrettet arbejde med fokus på teknologiskabelse?

4.2 Progression

Forløbet bygger videre på nogle af de arbejdsformer og tilgange, som eleverne har lært og udviklet igennem faget. Forløbet har et fokus på Digital myndiggørelse og teknologisk handleevne, men bygger på fagets sammenhæng mellem fagområderne. Derudover udvides elevernes arbejde nu med dataser og tekstprogrammering.

4.3 Differentieringsmuligheder

- Forskellige apps: differentiering kan ske på opgaveniveau, hvor eleverne kan arbejde med klassens app, eller andre typer apps, fx en sundhedsapp, hvor eleverne indsamler data om, hvor mange gange de drikker vand om dagen, eller spiser en grøntsag.
- Re-design eller nyt design: differentiering kan ske på ressourceniveau, hvor eleverne enten kan re-designe det viste eksempel, eller arbejde fra bunden med nyt design
- Komplexitet i programmering: eleverne kan arbejde primært med blokprogrammering eller tekstprogrammering.