

TEKNOLOGIFORSTÅELSE

SOM FAG 6. KLASSE
3. FORLØB

Computationelle strukturer i interaktive fortællinger

Udarbejdet af Eva Petropouleas Christensen og Ole Caprani i samarbejde med Bjarke Lindsø Andersen, Tina Hejsel og Louis Kørsen*

*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



LÆRE
MIDDEL
DK



VIA University
College



INDHOLDSFORTEGNELSE

1. Forløbsbeskrivelse	3
1.1 Beskrivelse	3
1.2 Rammer og praktiske forhold	4
2. Mål og faglige begreber	6
3. Forløbsnær del	10
3.1 Introfase: Forforståelse og kompetencer	10
3.2 Udfordrings- og konstruktionsfase	12
3.3 Outrofase: Ny forståelse og nye kompetencer	17
4. Perspektivering	19
4.1 Evaluering	19
4.2 Progression	19
4.3 Differentieringsmuligheder	20
4.4 Særlige opmærksomhedspunkter	20

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.

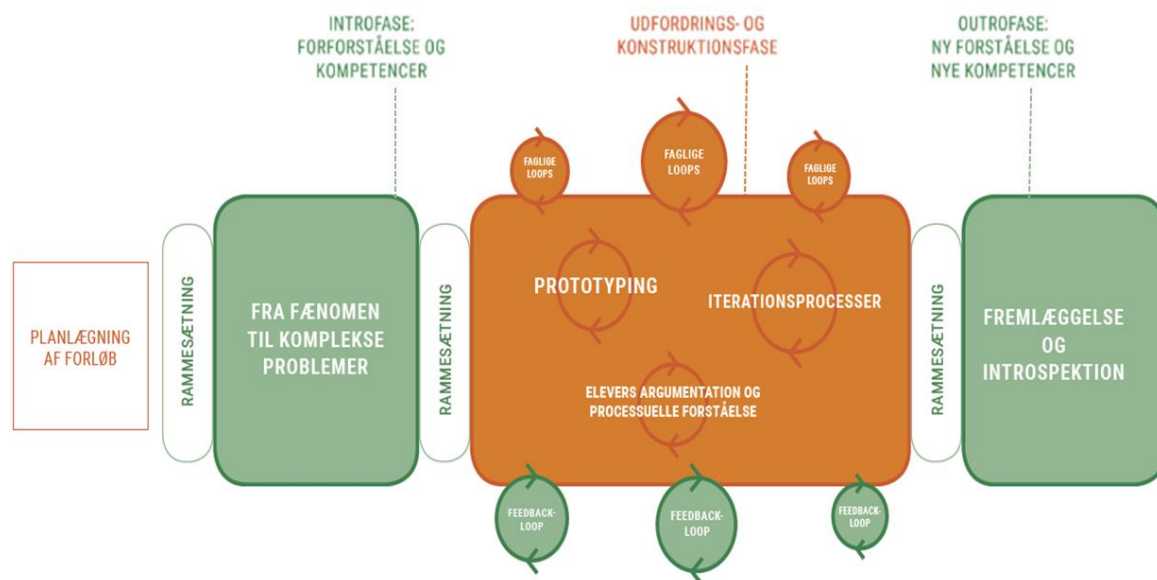
1. Forløbsbeskrivelse

Prototypen sigter mod at bibringe eleverne forståelser, som er essentielle, når de skal omsætte egne idéer til digitale artefakter.

I forløbet trænes eleverne i at strukturere deres idéer computationelt gennem skitsearbejde som en del af processen fra idé til færdigt artefakt og i at forstå, hvorledes samme computationelle struktur kan bruges i forskellige sammenhænge. Undervejs beskæftiger eleverne sig desuden med særlige muligheder, når der anvendes programmering i forbindelse med interaktive fortællinger.

Forløbet er bygget op over det didaktiske format for prototyperne, se figur 1.

Figur 1: Forløbsmodel for prototyperne



1.1 Beskrivelse

I introfasen introduceres eleverne gennem eksempler til begrebet interaktion og interaktive fortællinger som genre, og de får et indblik i forskellige strukturer. Desuden arbejder de med at lave en skitse over et eksisterende eksempels struktur.

I udfordrings- og konstruktionsfasen skal eleverne selv konstruere interaktive fortællinger gennem to rammesatte udfordringer.

I outro-fasen reflekterer eleverne over elementerne i forløbet og over, hvordan den viden om computationelle strukturer, som de har opnået gennem forløbet, kan anvendes i forbindelse med andre typer digitale artefakter. Afslutningsvis afprøver eleverne hinandens produkter.

Produkt

Eleverne producerer skitser og digitale interaktive fortællinger gennem forløbet. Der er udarbejdet lærer- og elevressourcer, som understøtter de forskellige elementer i forløbet. Disse findes i ressourcebanken til forløbet.

1.1.1 Problemfelt

Bag ethvert digitalt artefakt ligger en idé, som er omsat til en computationel struktur. En forståelse for computationelle strukturer er derfor essentiel i teknologiforståelsesfaget, både når eleverne forsøger at forstå eksisterende artefakters opbygning, og når de selv skal konstruere digitale artefakter og derfor skal kunne strukturere deres egne idéer computationelt.

Problemfeltet er således aktuelt for alle slags digitale artefakter, men rammesættes i dette forløb gennem genkravet "interaktive fortællinger". Dette skyldes, at:

- interaktive fortællinger er en genre, som i stigende grad vinder frem på grund af nye digitale muligheder, og derfor også en genre, som eleverne måske kender fra andre sammenhænge
- det er en nem genre for eleverne at arbejde med, når de skal udtrykke computationelle strukturer
- genren giver gode muligheder for at arbejde med centrale forhold vedr. programmering, herunder forskelle på opbygning af fortællinger med og uden programmering
- de computationelle strukturer, som eleverne skal arbejde med, lader sig overføre til andre typer artefakter, fx chatbots, websites, spil, apps – den opnåede forståelse gennem denne prototype kan altså anvendes videre frem i mange andre sammenhænge.

1.1.2 Problemstilling

På grund af ovennævnte rammesættelse lyder den konkrete problemstilling, som eleverne arbejder med, således:

"Hvordan kan jeg anvende viden om computationelle strukturer, når jeg skal omsætte mine idéer til interaktive fortællinger?"

Det er dog væsentligt, at eleverne i sidste del af prototypen også støttes i at overføre deres opnåede forståelse for computationelle strukturer i interaktive fortællinger til andre typer artefakter.

1.2 Rammer og praktiske forhold

1.2.1 Samlet varighed

Forløbet er estimeret til 15 lektioner.

Prototypen lægger op til, at de 15 lektioner fordeles således:

- Introfase ca 3 lektioner.
- Konstruktionsfase ca 10 lektioner.
- Outrofase ca 2 lektioner.

1.2.2 Materialer

Du finder elev- og lærerressourcer til forløbet i ressourcebanken til forløbet på www.tekforsøget.dk/forlob.

Analoge teknologier/materialer

- Papir (gerne A3 eller større), post-its, farveblyanter/tusser til at udforme skitser
- Abesnot eller tegnstifter/tape til at hænge skitser op

Digitale teknologier

- Computere

Oversigt over ressourcer i ressourcebanken

- Baggrundsmateriale til dig som lærer. I materialet kan du læse om genren "interaktive fortællinger, og om forskelle mellem hypertekster og programmerede tekster. Undervejs henvises til en række eksempler, som du kan bruge i din egen forberedelse eller vælge at dele med eleverne, hvis du synes. Baggrundsmaterialet indeholder desuden en grundig guide til Twine, som er det program, eleverne skal arbejde i.
- PP-præsentation, som følger forløbet, som det beskrives i prototypen. Du kan redigere præsentationen efter behov, så den passer til din undervisning.
- Kopiark til eleverne til øvelsen "tavs forhandling" i udfordrings- og konstruktionsfasen.
- Mappe med HTML-filer til alle de små eksempler på interaktive fortællinger, der anvendes undervejs. Vær opmærksom på, at du skal downloade dem, før du kan åbne dem op som fortællinger i din browser. Du kan også importere alle fortællingerne i Twine, så du får adgang til at ændre i dem.
- Video, som forklarer eksempel 3, der afprøves i introfasen.

Links

- Introducerende Twine-materiale: <https://sites.google.com/view/interaktivefortaellinger/startside>
Siden indeholder en række videoer, som introducerer Twine og guider eleverne gennem en udviklingsproces med interaktive fortællinger. **Udvalgte videoer til prototypen er indlejret i PP-præsentationen, som findes under ressourcer i ressourcebanken.**

1.2.3 Lokaler

Der arbejdes i almindeligt klasselokale.

1.2.4 Videnspersoner og andre eksterne aktører

Forløbet er ikke baseret på inddragelse af videnspersoner.

1.2.5 Tværfaglighed

Forløbet kan evt. samtænkes med dansk.

I dette forløb, som retter sig mod teknologiforståelse *som fag* fokuseres udelukkende på de computationelle mønstre.

Hvis forløbet samtænkes med dansk, vil det selvfølgelig være nødvendigt også at arbejde med elementer fra dette fag, fx det narrative indhold, den narrative opbygning, sprogton/sproglige greb og multimodale virkemidler.

2. Mål og faglige begreber

(*) Twine anvender meget simpel **tekstprogrammering**. Derfor er målene under programmering markeret med (*), da de omtaler **blokbaserede** programmeringssprog. At arbejde med tekstprogrammering rækker således ud over trinmålene. I udskolingen skal eleverne dog arbejde med tekstprogrammering. Denne prototype kan altså bidrage til at give eleverne en først og lettilgængelig erfaring med dette.

Twine er valgt som værktøj, da det er meget velegnet til denne prototype, og der findes et kvalificeret dansk introduktionsmateriale, som sagtens kan forstås af eleverne i 6. klasse,

KOMPETENCE-OMRÅDER	DIGITAL DESIGN OG DESIGNPROCESSER	COMPUTATIONEL TANKEGANG	TEKNOLOGISK HANDLEEVNE
Kompetencemål (efter 6. klassesettrin)	Eleven kan skabe digitale artefakter med digitale teknologier og gennemføre iterative designprocesser, der løser komplekse problemstillinger, relevante for individ og fællesskab	Eleven kan følge og anvende computationel tankegang i arbejdet med konkrete problemstillinger	Eleven kan, med udgangspunkt i viden om digitale teknologiers sprog og principper, handle med overblik med digitale teknologier i konkrete situationer
Færdigheds- og vidensmål (efter 6. klassesettrin)	Idégenerering <ul style="list-style-type: none"> Eleven kan anvende og argumentere for idégenererings- og eksternaliseringsteknikker for en konkret problemstilling Eleven har viden om forholdet mellem idégenererings- og eksternaliseringsteknikker for en konkret problemstilling 	Algoritmer <ul style="list-style-type: none"> Eleven kan genkende og tilrette algoritmer i forskellige sammenhænge og redegøre for deres funktion Eleven har viden om kendetegn ved algoritmer og deres 	Programmering (*) <ul style="list-style-type: none"> Eleven kan beskrive, tilrette og konstruere programmer i blokbaserede programmeringssprog samt foretage systematisk afprøvning og fejlretning af egne og andres programmer Eleven har viden om konstruktioner i

KOMPETENCE-OMRÅDER	DIGITAL DESIGN OG DESIGNPROCESSE	COMPUTATIONEL TANKEGANG	TEKNOLOGISK HANDLEVNE
		opbygning, samt hvordan de anvendes i forskellige sammenhænge	blokbaserede programmeringssprog og teknikker til systematisk konstruktion, fejlfinding og fejlretning af programmer
	Konstruktion <ul style="list-style-type: none"> Eleven kan med digitale teknologier konstruere artefakter, som udtrykker en idé, og kan reflektere over artefaktets anvendelse Eleven har viden om konstruktion af artefakter og om digitale teknologiers anvendelsesmuligheder 	Strukturering <ul style="list-style-type: none"> Eleven kan anvende mønstre i strukturering af data og dataproceser med udgangspunkt i konkrete problemstillinger Eleven har viden om mønstre i strukturering af data og dataproceser 	
	Argumentation og introspektion <ul style="list-style-type: none"> Eleven kan argumentere for sammenhænge mellem rammesættelse, idegenerering og konstruktion og kan forholde sig til egen designkompetence Eleven har viden om fagtermer for argumentation om designprocesser og for egen designkompetence 		

Konkretiserede læringsmål

- Eleven kan anvende skitser som et tænke- og designværktøj og til at udtrykke sine idéer for andre.
- Eleven kan redegøre for sin proces fra idé til færdigt produkt
- Eleven kan konstruere sin idé ved hjælp af digitale værktøjer
- Eleven kan se mønstre i og perspektivere sin løsning til andre typer artefakter

Centrale (teknologi)faglige begreber

Til Teknologiforståelse knytter sig et særligt ordforråd og bestemte måder at bruge sproget på.

Det faglige ordforråd, der er centralt for alle elevers læring i dette forløb, og som er vigtigt at arbejde

grundigt med i undervisningen, er oplistet nedenfor. Derudover indeholder listen særlige begreber for netop dette forløb

FAGLIGT BEGREB	BESKRIVELSE
Interaktion	Her tales om HCI, altså menneske-maskine-interaktion (human/computer-interaction). Ofte interagerer programdele jo også med hinanden, men i denne prototype handler det om de muligheder, en læser har, for at foretage valg i de interaktive fortællinger.
Idégenerering	Idégenerering omhandler systematisk behandling af viden med henblik på at skabe løsningsforslag, der gennem <i>eksternalisering</i> 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 <i>digitalt artefakt</i> kan bringes i anvendelse.
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 <i>digitalt artefakt</i> , der udarbejdes i tilgængelige materialer, for eksempel papir, træ eller i digitale omgivelser som App Lab. Disse skabes med henblik på at kommunikere idéer til andre og modtage feedback på disse idéer.
Iterativ proces	Man arbejder i en iterativ proces, når man ved hjælp af teknologi skaber et produkt, som man tester og herefter retter fejl, udfører forbedringer, tester igen, fejlretter og videreudvikler.
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 <i>brugspraksis</i> eller til anerkendte kriterier og konventioner for godt design.
Computational tankegang	Computational tankegang betegner elevens evne til at kunne omsætte rammesatte problemstillinger på en måde, så de kan eksekveres af en computer Computational tankegang omhandler analyse, modellering og strukturering af data og dataprocesser; at kunne afkode fænomener og processer (fra hverdagen, fra faglige sammenhænge og i digitale artefakter) og beskrive disse i form af algoritmer og modeller.

FAGLIGT BEGREB	BESKRIVELSE
Konstruktion	Konstruktion omhandler den aktivitet, hvor idéer finder udtryk i et konkret <i>digitalt artefakt</i> , som kan gøres til genstand for en efterprøvning af form, funktion og interaktion. Konstruktion i <i>digitale teknologier</i> rummer aktiviteter såsom <i>computational tankegang</i> , valg af <i>programmeringssprog</i> , <i>programmering</i> , design af grænseflade, konstruktion af en tidlig papirmodel eller diagram over det fremtidige artefakt, at bygge den fysiske udformning af et <i>digitalt artefakt</i> og iterative evalueringer af <i>digitale artefakter</i> under udvikling.
Peer feedback	At få feedback fra peer handler om at få feedback fra ligestillede – i dette eksempel fra andre elever.
Divergent tænkning	Når eleverne arbejder <i>divergent</i> , arbejder de med at udfolde problemfeltet eller udvikle mange ideer samtidig. Når elever arbejder i designprocesser, benyttes både divergent og konvergent tænkning.
Konvergent tænkning	Når eleverne arbejder <i>konvergent</i> , arbejder de med indsnævring af problemet eller idéløsning og kvalificering. Når elever arbejder i designprocesser, benyttes både divergent og konvergent tænkning.
Digitale artefakter	Digitale artefakter kan betegnes som applikationer eller 'devices', som er produceret med henblik på at opfylde en bestemt funktion og tjene et bestemt formål.
Knudepunkter/knuder /flaskehalse	I introduktionsmaterialet til Twine anvendes dette begreb. Knuder/knudepunkter/flaskehalse er de steder i den interaktive fortælling, hvor læseren stilles overfor et valg
Blindgyder	I introduktionsmaterialet til Twine anvendes dette begreb. Blindgyder er de steder i den interaktive fortælling, hvor læseren dør eller historien må starte forfra af anden grund. Det er altså blindgyder.
Statisk fortælling	Her er de valg, som læseren møder, forudbestemte og ens hver gang. Man kan sige, at fortællingen er genereret, før læseren går i gang.
Dynamisk fortælling	Her er de valg, som læseren møder, bestemt af forudgående handlinger og valg. For at kunne lave dynamiske fortællinger, er eleverne nødt til at anvende programmering. Deres valg skal kunne lagres som variable, og programmet skal kunne udføre beregninger på disse variable. Man kan sige, at fortællingen bliver genereret, mens læseren er i gang.

I forbindelse med arbejdet med elevernes forståelse af faglige begreber kan læreren stilladsere elevernes sproglige udvikling ved at anvende begreberne selv, sikre at eleverne forstår betydningen og støtte dem i at anvende de faglige begreber selv i deres egne sproglige ytringer.

Se ordliste på EMU over udvalgte faglige begreber i teknologiforståelse:
<https://emu.dk/grundskole/teknologiforstaelse/ordliste?b=t5-t34>

3. Forløbsnær del

3.1 Introfase: Forforståelse og kompetencer

Introfasen består af følgende tre elementer:

- Fagligt loop:
- Afprøvning af eksempel:
- Undersøgelse af eksempel og fælles udformning af skitse over struktur

3.1.1 Varighed

Estimeret varighed: 3 lektioner á 45 minutter

3.1.2 Problemfelt

Hele forløbet adresserer et overordnet problemfelt, som er beskrevet indledningsvis.

3.1.3 Problemstilling i introfasen

Problemfeltet er på forhånd afgrænset til at omhandle en problemstilling vedrørende computationelle strukturer i interaktive fortællinger

I introfasen introduceres eleverne til den afgrænsede problemstilling gennem følgende spørgsmål:

- Hvad betyder begrebet interaktion og hvad er interaktive fortællinger?
- Hvordan bygges interaktive fortællinger op?

Isenesættelse/scenarie:

■ Fagligt loop:

Det faglige loop er stilladsret gennem PP-præsentationen til prototypen, men skitseres her i kort form.

Indledende klassedrøftelse. Her repeteres begrebet "interaktion", som eleverne har arbejdet med i tidligere prototyper. Du kan fx tage udgangspunkt i:

- Prototype 6: [Interaktion med Scratch og micro:bit](#) fra 4. årgang, hvor eleverne arbejdede med en oversigt over input/output og forskellige input-/output-enheder, som bruges til at interagere med

en computer/et program. Vis evt. denne oversigt og tal om den igen.

- Elevernes erfaringer med micro:bitten – hvordan interagerer man med den? Husk fx tilbage på afstemninger eller spionlegen.
 - Det gennemgående tema ”mennesker, maskiner, magt og samarbejde” fra 5. årgang. Måske kan eleverne selv formulere forskellige måder at interagere med noget på ud fra konkrete eksempler fra dette års arbejde?
- **Afprøv eksemplet** ”Spørgeshvisker i fællesskab.
 - **Se første video** i Twine-materialet om interaktive fortællinger. Videoen findes her <https://youtu.be/sjDTKnJkWXI> men er også indlejret i PP-præsentationen. Hav derefter en fælles samtale om interaktive fortællinger. Måske kender eleverne andre eksempler, end dem, som videoen nævner?
 - **Vis slide** fra powerpoint præsentationen med forskellige fortællestrukturer og lad klassen kigge på eksemplerne et af gangen og prøve at forestille sig, hvilke typer fortællinger, der kan laves ud fra hver struktur. Brug noterne fra slide´t og introducéer begreberne ”knudepunkter”/”knuder”/ ”flaskehalse” og ”blindgyder”, hvor det giver mening (eleverne møder ellers først begreberne i den tredje video til Twine under udfordrings- og konstruktionsfasen, men det gør ikke noget, at de har hørt dem før, så videoen dermed repeterer dem frem for at introducere dem).
 - **Afprøvning af eksemplet** ”Et mærkeligt hus” og sammenligning med grafen – link og graf findes i præsentationen.
 - **Afprøvning af eksemplet** ”Eventyr mashup”: Del HTML-filen med eleverne: Der afsættes 10 minutter til afprøvning.
 - **Undersøgelse af eksempel 3** og fælles udformning af skitse over struktur: Lad eleverne undersøge, hvordan historien egentlig er bygget op. Du kan enten lade klassen lave en fælles skitse over opbygningen, fx på tavle, eller arbejde med opgaven i store grupper.
 - **Fagligt loop:**
Vis eleverne video, som gennemgår, hvordan eksemplet kan laves. Videoen findes i ressourcebanken og gennemgår både eksemplet uden og med programmering.

3.1.4 Faglige loops

Formålet med det første faglige loop er at etablere en forforståelse og relevant ordforråd, som eleverne skal anvende, både i den efterfølgende lille øvelse med afprøvning og undersøgelse af eksempel, og senere i forløbet, når de skal udforme deres egne interaktive fortællinger. Det første faglige loop består af en række mindre aktiviteter, som er skitseret ovenfor og i den powerpoint præsentation, som er udarbejdet til læreren.

Det andet faglige loop bruges som afslutning på elevernes egne undersøgelser.

3.2 Udfordrings- og konstruktionsfase

Eleverne arbejder både individuelt, i makkerpar og i større grupper i denne fase.

Undervejs anvendes udvalgte videoer fra et større introduktionsmateriale til Twine, hvor eleverne allerede har set første video i introfasen. Hele materialet findes her:

<https://sites.google.com/view/interaktivefortaellinger/startside>

Udfordrings- og konstruktionsfasen er bygget op over en længere række aktiviteter, som beskrives nærmere under punkt 3.2.3. i skematisk form. Her peges på, hvilke ressourcer eleverne skal arbejde med hvornår. Dette fremgår også af den powerpoint præsentation, som læreren kan bruge til sin undervisning, og hvor de udvalgte videoer er indlejrede. Vær opmærksom på, at der inddrages flere computationelle strukturer i PP-præsentationen, end Twine-videoerne lægger op til.

Der er angivet estimeret tid ud for hver enkelt aktivitet. Nogle elever vil måske have behov for længere tid til de enkelte dele. Se punkt 4.3 for forslag til, hvorledes der kan differentieres.

Det er en særlig opgave at sikre, at eleverne arbejder *teknologifagligt*, som det er beskrevet i fælles mål, herunder at de programmerer, når de konstruerer digitale artefakter. Det er altså ikke nok blot at lave fortællingerne som statiske hypertekster. Med programmering kan eleverne gøre deres fortællinger dynamiske. Baggrundsmaterialet til læreren, som findes i ressourcebanken, gennemgår forskellen mellem statiske og dynamiske fortællinger grundigt.

I den første udfordring skal eleverne blot lære det grundlæggende i programmet Twine, og derfor indgår der ikke programmering i denne. Dette er til gengæld hovedfokusset i den anden udfordring, hvor eleverne arbejder med at modificere og udvide en eksisterende (programmeret) fortælling.

3.2.1 Problemstilling i udfordrings- og konstruktionsfasen

I udfordrings- og konstruktionsfasen arbejdes med den afgrænsede problemstilling gennem følgende spørgsmål

- Hvordan kan jeg organisere og udtrykke mine idéer?
- Hvordan kan jeg sammen med andre udvikle fælles idéer?
- Hvordan kan jeg bruge strukturer til at støtte mig i arbejdet med at konstruere en interaktiv fortælling?
- Hvilke nye muligheder giver programmering, når jeg laver interaktive fortællinger?

3.2.2 Varighed

Estimeret varighed: 10 lektioner á 45 minutter

3.2.3 Konkret(e) udfordring(er)

UDFORDRING 1: LÆR TWINE AT KENDE, MENS DU LAVER DIN EGEN EVENTYRFORTÆLLING

Tavs forhandling (ca 30 minutter)

Gennem aktiviteten får deltagerne eksternaliseret egne idéer først, så de alle kan indgå kollaborativt med lige deltagelsesmuligheder. Aktiviteten giver mulighed for på en legende, anderledes og sjov måde at samskabe om en fælles idé og eleverne anvender undervejs både divergent og konverget tænkning (se ordlisten).

Aktivitetens struktur gør det nemt at iterere, indtil alle er enige. Slutresultatet er, at alle deltagere gennem aktiviteten ender med en række byggesten, der kan arbejdes videre med.

1. Læreren rammesætter aktiviteten ved at skrive følgende kategorier på tavlen: "Subjekt, objekt, hjælpere, modstandere, prøvelser". Eleverne bør kende kategorierne fra arbejde med folkeeventyr i dansk, men evt. kan læreren kort repetere dem. Læreren udleverer derefter post-its og instruerer eleverne i, at der fra nu af **ikke må tales** overhovedet.
2. Læreren sætter alle i gang med at idégenerere tavst ud fra kategorierne. De skal individuelt skrive så mange muligheder under hver kategori, som de kan komme i tanke om. En post-it pr. idé. Efter ca 5 minutter fortsættes til punkt 3.
3. Læreren instruerer eleverne i at gå en tur rundt og se de andres post-its og kopiere, hvis de opdager nye idéer, de gerne vil have med i deres eget "idé-dæk". Ca. 5 minutter.
4. Læreren instruerer eleverne i at gå sammen i større grupper (eksempelvis 6-8 deltagere). Alle medbringer deres eget "idé-dæk" og hver gruppe får nu et fællesark med begrænsede pladser til idéer under hver kategori.
5. Læreren instruerer eleverne i, hvordan de skal kollaborere om at udfylde fællesarket, og det understreges, at der stadig **ikke må tales**:
 - A. Man skal lægge post-its fra eget "idé-dæk" på, hvor man synes.
 - B. Det behøver ikke gå efter tur.
 - C. Når alle felter er fyldte, må man fjerne en post-it for at få plads til en anden.

UDFORDRING 1: LÆR TWINE AT KENDE, MENS DU LAVER DIN EGEN EVENTYRFORTÆLLING

	<p>D. Man må også gerne bare fjerne en, hvis man vil.</p> <p>E. Man må også gerne tage fra de andres "idé-dæk", hvis man pludselig ser noget, man gerne vil have med på fællesarket. I instruktionen lægges vægt på, at man både skal være åben overfor og prøve at arbejde med på andres idéer, men også vedholdende, hvis man mener, ens eget element er meget vigtigt. Her er det særlig vigtigt, at der ikke tales sammen, men forhandles gennem fjernelse og placering af post-its</p> <p>6. Læreren iagttager og indgår evt. som flydende deltager, hvis grupperne har svært ved at komme i gang.</p> <p>7. Læreren sikrer, at grupperne faktisk er enige, før aktiviteten stoppes. Nogle gange kan der godt være "stilstand" et stykke tid, før der pludselig er en, der alligevel ændrer noget, og gruppen dermed kommer i gang igen.</p>
<p>Arbejde med at lave mindmap og skitse: (ca 60 minutter)</p>	<p>Eleverne arbejder videre i makkerpar med de byggesten, de har fået gennem den tavse forhandling. De skal lave mindmaps og skitser. For at stilladsere elevernes arbejde med dette, skal de først se følgende videoer fra Twine-materialet:</p> <ul style="list-style-type: none"> ■ Brainstorms og mindmaps ■ Timeglasset og fodboldbanen: <p>Læreren viser igen eleverne slide med de mange andre strukturer, som man kan arbejde ud fra.</p> <p>Alle makkerpar har et A3 ark (og gerne større, hvis muligt, men ellers kan flere ark tapes sammen). Skitsen udarbejdes på papir-arket med små post-it eller, der skrives direkte på papir-arket. Det er frivilligt, men det er nemmere at fejlrette, hvis der bruges post-it's.</p> <p>Det er vigtigt, at eleverne fastholdes lidt i at gennemtænke fortællingen for at undgå alt for mange problemer i selve konstruktionen, fordi der ikke er overblik over den fulde proces. Læreren understøtter dette ved at besøge hvert makkerpar og sammen tale om deres idéer.</p> <p>Læreren kan også anbefale elever, som har svært ved opgaven, at besøge andre makkerpar og se, hvordan de arbejder, for på den måde at blive inspireret til mulige løsninger.</p>

UDFORDRING 1: LÆR TWINE AT KENDE, MENS DU LAVER DIN EGEN EVENTYRFORTÆLLING

<p>Fagligt loop: Introduktion til twine + konstruktion af eventyrfortælling undervejs (Ca 90 minutter)</p>	<p>Klassen arbejder med følgende videoer i Twine-materialet, som introducerer dem til de basale funktioner. (De udvalgte videoer er indlejrede i PP-præsentationen, som findes i ressourcebanken)</p> <ul style="list-style-type: none"> ■ Lær Twine at kende ■ Lav din første side i Twine <p><i>Eleverne laver et skelet af forbundne passager i Twine og skriver noter til, hvad der skal ske</i></p> <ul style="list-style-type: none"> ■ Formatering i Twine ■ Indsæt billeder og video i twine <p><i>Eleverne gør deres lille eventyrfortælling færdig</i></p>
<p>Feedback loop: Peer feedback (ca. 30 minutter)</p>	<p>Makkerparrene går sammen med et andet makkerpar og afprøver hinandens små fortællinger. Derefter skal de hver især lave en skitse ud fra det andet pars lille fortælling. Til sidst sammenligner de to par skitserne.</p> <p>Eleverne giver hinanden feedback ud fra hvor godt skitserne ligner de originale skitser og det færdige artefakt. Hvis der er store forskelle på de oprindelige skitser og det færdige artefakt, skal eleverne kunne redegøre for, hvorfor de har ændret deres idé undervejs. Hvis der er store forskelle på de oprindelige skitser og de skitser, som de afprøvende elever har lavet, må de sammen undersøge, hvorfor det mon er sådan.</p>

UDFORDRING 2: ARBEJD MED PROGRAMMERING I TWINE GENNEM MODIFIKATION/UDVIDELSE AF FORTÆLLINGEN "I SKOLE TIL TIDEN"

<p>Undersøgelse af grundfortællingen (ca. 30 minutter)</p>	<p>Eleverne skal nu arbejde videre i makkerpar. De skal åbne fortællingen "I skole til tiden" ved at importere HTML-filen i Twine. Det er beskrevet i baggrundsmaterialet til læreren, hvordan dette gøres.</p> <p>Eleverne afprøver fortællingen og forsøger at forstå, hvad programmeringen gør.</p> <p>Afslutningsvist forklarer læreren, hvordan programmeringen virker. Dette er også beskrevet i baggrundsmaterialet til læreren.</p>
<p>Fagligt loop (ca. 15 minutter)</p>	<p>Læreren forklarer, hvordan programmeringen virker, så eleverne forstår koderne. Dette er også beskrevet i baggrundsmaterialet til læreren.</p>
<p>Idégenerering (ca. 25 minutter)</p>	<p>Eleverne skal nu modificere og evt. udvide fortællingen med nye valgmuligheder/forhindringer. Makkerparrene skriver så mange idéer ned på post its, som de kan komme på. Hvad kan der ellers ske, som kunne skabe forsinkelser eller hjælpe en, inden man kommer i skole. Det kan både være realistiske idéer og helt skøre påfund, fx at der lander en UFO i baghaven, som man tager med, eller at man opfinder en tidsmaskine, som kan give ekstra minutter.</p> <p>Læreren sørger for, at eleverne ved, at de bagefter skal konkurrere med et andet makkerpar om at have flest idéer.</p>
<p>Feedback loop: (ca. 20 minutter)</p>	<p>Hvert makkerpar går sammen med et andet makkerpar. Efter tur "spiller" makkerparrene en post it, dvs. de lægger en post it på bordet og forklarer kort idéen. Hvis idéen allerede er beskrevet af det andet makkerpar, kan den ikke længere spilles.</p> <p>Det makkerpar, som først løber tør for post its, har tabt.</p>
<p>Idéfokusering: (ca 45 minutter)</p>	<p>Makkerparrene har nu fået inspiration til endnu flere muligheder. De skal vælge i alt 3 idéer, som de vil indarbejde i grundprogrammet. De laver en hurtig skitse over grundprogrammet, og hvor de tre idéer skal kobles ind.</p> <p>De kan enten ændre i eksisterende knuder eller de kan tilføje helt nye knuder.</p>
<p>Konstruktion af første modifikation/udvidelse</p>	<p>Eleverne konstruerer deres første modifikation/udvidelse. Enten ændrer de teksterne i eksisterende knuder eller også laver de helt nye knuder og</p>

UDFORDRING 2: ARBEJD MED PROGRAMMERING I TWINE GENNEM MODIFIKATION/UDVIDELSE AF FORTÆLLINGEN "I SKOLE TIL TIDEN"

(ca 45 minutter)	trækker nye linjer i mellem. De nye knuder skal laves som de eksisterende, så læserens valg gemmes som en variabel.
Konstruktion af de øvrige modifikationer/udvidelser (ca 90 minutter)	Eleverne arbejder videre med at implementere de resterende to idéer som ovenfor.

3.3 Outrofase: Ny forståelse og nye kompetencer

I outrofasen skal eleverne redegøre for og reflektere over deres proces ud fra de konkretiserede læringsmål, som lød:

- Eleven kan anvende skitser som et tænke- og designværktøj og til at udtrykke sine idéer for andre.
- Eleven kan redegøre for sin proces fra idé til færdigt produkt
- Eleven kan konstruere sin idé ved hjælp af digitale værktøjer
- Eleven kan se mønstre i og perspektivere sin løsning til andre typer artefakter

Læreren faciliterer dette ved hjælp af co-operative learning strukturen "Dobbeltcirkler":

3.3.1 Varighed

Estimeret varighed: 2 lektioner á 45 minutter

3.3.2 Fremlæggelse og introspektion

- Der anvendes dobbeltcirkler til en del af fremlæggelsen og introspektionen for at sikre, at alle får talt og lyttet. Strukturen er beskrevet nedenfor

DOBBELTCIRKLER

Trin 1: Halvdelen af eleverne former en cirkel og vender ryggen mod centrum. Den anden halvdel af eleverne danner en cirkel udenom, således at hver elev stiller sig ansigt til ansigt med en fra indercirklen.

Trin 2: Læreren stiller et spørgsmål, evt. flere – eleverne skiftes til at svare. Læreren vælger, om det er eleverne i yder- eller indercirklen, der skal svare først.

Trin 3: Når læreren siger "skift", rykker ydercirklen en plads til højre, hvorefter trin 2 gentages

DOBBELTCIRKLER

Læreren kan fx stille spørgsmål á la nedenstående:

Tænk på strukturer i interaktive fortællinger	<ul style="list-style-type: none"> ■ Hvad er knuder? ■ Hvad er flaskehalse? ■ Forklar en struktur, du kan huske ■ Forklar en anden struktur, du kan huske ■ (...)
Tænk på mindmaps og skitser:	<ul style="list-style-type: none"> ■ Lavede du dem rigtigt første gang, eller lavede du om på noget en eller måske flere gange, før du var færdig? ■ Opdagede du noget undervejs, som gjorde at du ændrede din idé? ■ Brugte du skitserne, da du skulle lave dit artefakt? ■ (...)
Tænk på din færdige eventyrfortælling	<ul style="list-style-type: none"> ■ Lykkedes det at lave det, som du og din makker gerne ville? ■ Hvis du havde længere tid, hvad ville du så gerne lave? ■ Hvad var svært/let? ■ (...)
Tænk på den tavse forhandling	<ul style="list-style-type: none"> ■ Gav metoden dig nye idéer, du ikke havde tænkt på inden? ■ Hvad kunne du godt lide eller ikke lide ved metoden? ■ Lykkedes det jer i gruppen at blive enige om, hvad der skulle være på fællesarket? ■ (...)
Tænk på udfordring 2, hvor du og din makker skulle modificere den programmerede fortælling: "I skole til tiden"	<ul style="list-style-type: none"> ■ Forklar med dine egne ord, hvad programmering giver af nye muligheder? ■ Forklar med dine egne ord, hvad forskellen er på statiske og dynamiske fortællinger ■ (...)

- Derefter drøfter klassen fælles, hvordan viden om strukturer kan bruges i forbindelse med andre artefakter. Læreren faciliterer dette ved at spørge eleverne om, hvordan knuder og flaskehalse fx bruges i følgende eksempler:
 - Quiz (ved fejl dør man)

- Chatbot (man kan få valgmuligheder undervejs og forskellige svar kan føre tilbage til samme spørgsmål)
- App (hjem-knap kan fx lede tilbage til udgangspunkt, man kan få forskellige valgmuligheder undervejs)
- Spil (man kan dø, man kan måske blive sendt tilbage til start, måske får man valgmuligheder undervejs)
- (...)

- Til sidst afprøver eleverne hinandens artefakter

4. Perspektivering

En del af formålet med forløbet er, at eleverne bliver bevidste om, at de computationelle mønstre, som de har arbejdet med, går igen på tværs af typer af digitale artefakter. Den afsluttende klassesdrøftelse i outrofasen er således helt central for forløbet, og det er væsentligt, at den opnåede forståelse for dette anvendes fremadrettet.

Således vil det være hensigtsmæssigt at gribe tilbage til erfaringerne fra denne prototype og begrebet "computationelle mønstre", når eleverne fremover skal konstruere et digitalt artefakt. Du kan som lærer helt konkret bede eleverne om at formulere deres idéer som skitser som en fast bestandel, enten i den indledende idégenerering eller som del af idéfokuseringen eller måske begge dele, når de selv skal konstruere et artefakt, og du kan i forbindelse med analyse af forskellige eksisterende artefakter eller som del af feedbackloops på deres egne artefakter bede dem om at reflektere over, hvilket computationelt mønster, der ligger bag.

4.1 Evaluering

En stor del af evalueringen foregår ved, at læreren er aktivt lyttende. Det gælder både i peer feedback aktiviteterne og i dobbelt-cirkel-øvelsen. Derudover evaluerer læreren løbende, mens eleverne arbejder med de konkrete udfordringer gennem små dialoger med eleverne om, hvor de er i deres proces, og hvad de forestiller sig, der nu skal ske.

Elevernes produkter undervejs (mindmaps, skitser og digitale artefakter) er også evalueringspunkter for læreren, ligesom den afsluttende klassesdrøftelse kan afdække, om eleverne har fået en bredere forståelse af computationelle strukturer gennem forløbet.

4.2 Progression

Forløbets formål peger frem mod dybdekompetencer i forhold til computationel tankegang og digitale designprocesser, i det det tilstræber, at eleverne kan overføre viden fra dette forløb til andre sammenhænge.

I takt med, at eleverne skal kunne løse mere komplekse problemstillinger med en større grad af selvstændighed, er følgende elementer væsentlige:

- Metoder til at eksternalisere og forhandle om idéer (tavs forhandling, mindmaps, skitser)

- At kunne give og modtage feedback
- At kunne opnå viden gennem (video)tutorials og anvende den i egne projekter
- At kunne reflektere over og argumentere for egen designproces og -kompetence.

4.3 Differentieringsmuligheder

I udfordrings- og konstruktionsfasen kan differentieres på følgende måder:

- I det første projekt skal eleverne se seks små videoer, som introducerer til brainstorm, mindmapping og Twine (disse er indlejret i PP-præsentationen). Hvis læreren vurderer, at det tidsmæssigt er svært at nå, kan de to sidste (formatering i Twine + Indsæt billeder og video i Twine) udelades, så eleverne kun arbejder med at skrive uformaterede tekster og lave links i mellem dem.
- I det andet projekt lægges der op til, at eleverne skal finde på tre modifikationer/udvidelser, som skal indarbejdes i grundprogrammet. Læreren kan begrænse dette til en idé og kan desuden lade eleverne modificere eksisterende knuder i fortællingen frem for at udvide med nye knuder.

4.4 Særlige opmærksomhedspunkter

I forløbet skal eleverne arbejde selvstændigt (i makkerpar) det meste af tiden. Det stiller krav til læreren om at være tydelig i sin instruktion og særligt opmærksom på måder, hvorpå han/hun kan understøtte eleverne undervejs i deres processer.

Der er ikke indlagt tid til "fælles statusser", så læreren skal sørge for at komme omkring alle grupper/makkerpar hver gang.

Det er altid en udfordring med projekter over tid, når elever er fraværende. Læreren skal på forhånd tage stilling til, hvordan det kan løses, fx ved at den tilstedeværende elev arbejder videre på egen hånd, men tilknyttes et andet makkerpar, så det er muligt at spørge om hjælp.

Hvis begge elever i et makkerpar er fraværende og dermed mangler noget væsentligt for at kunne gå videre, kan læreren evt. dele dem ud på to andre makkerpar, så de kan fortsætte i nye konstellationer, når de igen er tilstede.