

Oppgave 1

Feil nr 1: Linje 6 til Linje 12. `<style>` tag er ikke lukket. Dette vil føre til at alt innholdet på siden ikke blir vist, da det regnes som CSS og ikke som HTML.

Feil nr 2: Linje 25, 26, 32, 47, 56, 66 og 70. Bruk av `
` tag for å posisjonere/skape avstand mellom elementer. Posisjonering

Feil nr 3: Linje 4 til linje 14. `<head>` tag er ikke lukket.

Feil nr 4: Linje 42, 43. `` tag er ikke lukket.

Feil nr 5: Linje 50 `` tagen havner utenfor slutt på `<p>`. ``

Feil nr 6: Linje 15 til 120: `<body>` tag er ikke lukket.

Feil nr 7: Linje 104 til 110: Det mangler en `<td>` innenfor `<tr>`. Dette gjør at tabellen mangler én tom rute.

Feil nr 8: Linje 37 (med kontekst fra `<h2>` linje 34). Her hoppes det fra h2 til h4. I stedet for `<h4>` burde det vært brukt `<h3>` her. Om man vil så kan man eventuelt endre størrelsen på overskriften i CSSen. Alle påfølgende `<h4>`-er burde altså også være `<h3>`.

Feil nr 9: Det mangler `<!DOCTYPE html>`.

Feil nr 10: Det mangler `<meta>` tag med charset i `<head>`.

Oppgave 2:

Alle tomme <td> fjernes (Linje 93, 94, 101, 102, 109, 116 og 117)

Fra linje 89 blir koden endret til:

```
<tr>
  <td>Systemutvikling</td>
  <td>Praktisk prosjektarbeid</td>
  <td colspan="2">Applikasjonsutvikling for web</td>
  <td rowspan="4">Selvvalgte emner</td>
  <td rowspan="4">Bacheloroppgave og selvvalgte emner</td>
</tr>
```

Endringen er 2 nye <td> med rowspan="4".

Resten av koden er som før, men unntak av at tomme <td> har blitt fjernet, som spesifisert tidligere.

Oppgave 3:

Dokumentet burde starte med `<!DOCTYPE html>`, og ha en `<meta>` tag med charset innenfor `<head>`

Det burde brukes en `<header>` tag som inneholder alt fra linje 16 til 26.

Det burde brukes en `<nav>` tag for meny lenkene fra linje 18 til linje 25.

Fra linje 27 til linje 120 kan det brukes en `<main>` tag hvis man vil gjøre noe med innholdet som går utenom hovedoverskrift og meny-lenker.

Hver del burde deles opp i `<article>`-tagger. Eksempel på dette er innholdet fra linje 27 til 32. Hvis dette innholdet, og innholdet som har samme funksjon som denne har en `<article>`-tag rundt seg så vil man kunne posisjonere denne delen uten bruk av `
` tag slik som det er nå. Det ser ut som `
` taggen er brukt for å skape vertikal avstand mellom de ulike delene - Dette burde gjøres ved bruk av `<article>` tag hvor man styrer f.eks margin i CSSen.

Linje 38, 50, 59: I stedet for å ha en `<p>Tekst</p>` for å lage underoverskrifter så burde man brukt neste nivå av h1-6-taggen, eller bare ha en vanlig `<p>` uten font-weight bold. Man kunne også brukt `<p class="viktig">` (eller et annet class-navn) hvis man må bruke `<p>`-taggen for en eller annen grunn, da dette gjentar seg flere steder i html-koden.

Oppgave 4

Basert på utvidelse i oppgave 2, fjerning av
 og andre feil, samt bruk av semantiske tagger skal følgende kode være lik den på bildet i oppgaveteksten oppgave 4:

```
<!DOCTYPE html>
<html>

<head>
<meta charset="UTF-8">

<style>

body    {background-color:lightblue;
        }
main    {background-color:lightyellow;
        margin-bottom:3em;}
article {margin-bottom:2em;}
nav a   {margin-right:0.2em;}
h1,h2,h3 {color:blue;
          font-family:Verdana, sans-serif;}

table   {border: 3px solid black;
        border-collapse: collapse;
        background-color:lightgreen;
        }
th, td  {border: 1px solid black;
        }
@media (min-width: 1501px) {
    main {column-count: 2;}
    #del6 {break-before:column;}
}
</style>

<title>Studieplan IT og informasjonssystemer</title>
</head>

<body>

<header>
<h1> Studieplan for bachelor i IT og Informasjonssystemer</h1>

<nav>
<h2>Innholdsfortegnelse</h2>
<a href="#del1">Om studieprogrammet</a>
<a href="#del2">Læringsutbytte</a>
<a href="#del3">Kunnskaper</a>
<a href="#del4">Ferdigheter</a>
<a href="#del5">Generell kompetanse</a>
<a href="#del6">Læringsaktiviteter</a>
<a href="#del7">Studiemodell</a>
</nav>

</header>

<main>

<article>
```

<h2 id="del1">Kort om studieprogrammet</h2>

<p>Studiet i IT og informasjonssystemer er et treårig, fulltids studium som gir 180 studiepoeng og graden Bachelor i IT og informasjonssystemer.</p>

<p>Næringslivet og offentlig sektor har stort behov for medarbeidere med kompetanse innenfor program- og systemutvikling, webbaserte løsninger og IT-forvaltning generelt. Studiet i IT og informasjonssystemer imøtekommer denne etterspørselen. Studiet gir et godt fundament innenfor teknologi, med vekt på programmering og systemutvikling. Dette kombineres med kunnskap og praktiske ferdigheter om hvordan teknologien brukes i bedrifter og virksomheter. Denne tilnærmingen gir studenter som er ettertraktet i arbeidsmarkedet og godt rustet for videre studier på masternivå</p>

<p>Bachelor i IT og informasjonssystemer gir et godt teknisk grunnlag og en forståelse av samspillet mellom teknologi og organisasjon, gjennom fokus på bruk av teknologi i næringsliv og offentlig sektor. Studiet gir god kompetanse i utvikling, innføring og anvendelse av IT-systemer i offentlig og privat sektor.</p>

</article>

<article>

<h2 id="del2">Læringsutbytte</h2>

<p>En kandidat med fullført bachelorgrad i IT og informasjonssystemer skal ha følgende totale læringsutbytte:</p>

</article>

<article>

<h3 id="del3">Kunnskap</h4>

<p>Kandidaten har kunnskap om</p>

informasjonsteknologien i dens fulle bredde og kompleksitet

virksomheters mest grunnleggende behov for informasjonssystemer

Systemutviklingsteorier, modeller og verktøy

organisering av data og databaser

prosjektledelse og prosjekt som arbeidsform

et utvalg av programmeringsspråk og verktøy som anvendes for utvikling av applikasjoner

</article>

<article>

<h3 id="del4">Ferdigheter</h3>

<p>Kandidaten kan</p>

utføre drifts-, utviklings, opplærings- og IT-lederoppgaver i en virksomhet

lede og delta i ulike roller i utviklingsprosjekter

programmere applikasjoner og web-løsninger

</article>

<h3 id="del5">Generell kompetanse</h4>

<p>Kandidaten kan</p>

planlegge og gjennomføre prosjekter sammen med andre

analysere relevante IT-faglige problemstillinger

anvende sine IT-faglige kunnskaper og ferdigheter i utvikling av nye virksomhetsprosesser

bidra til nytenking og i innovasjonsprosesser i forbindelse med bruk av IT i virksomheter

<article>

<h2 id="del6">Læringsaktiviteter</h2>

Sentrale læringsaktiviteter er forelesninger og fagseminarer, skrive- og presentasjonstrening, praktiske øvelser og caseoppgaver, samt bacheloroppgave. Læringsaktivitetene vil vektlegge en prosessuell tilnærming på individ og gruppenivå. Det vil si at studenten må samarbeide med andre studenter og komme fram til et felles resultat for å nå læringsutbyttene. Det forutsettes at studentene arbeider selvstendig og i grupper med forerledelser og etterarbeid knyttet til læringsaktivitetene, samt arbeid med pensumlitteratur.

Bachelor i IT og Informasjonssystemer					
Obligatorisk emner 1. og 2. studieår - 120 stp					
2020 Høst	2021 Vår	2021 Høst	2022 Vår	2022 Høst	2023 Vår
Systemutvikling	Praktisk prosjektarbeid	Applikasjonsutvikling for web		Selvvalgte emner	Bacheloroppgave og selvvalgte emner
Informasjonssystemer	Database 1	Database 2	Objektorientert programmering 2		
Grunnleggende programmering 1	Grunnleggende programmering 2	Objektorientert programmering 1	Objektorientert analyse og design		
Webutvikling og HCI	Organisering og ledelse	Etikk og samfunnsansvar	Samfunnsvitenskapelig metode		

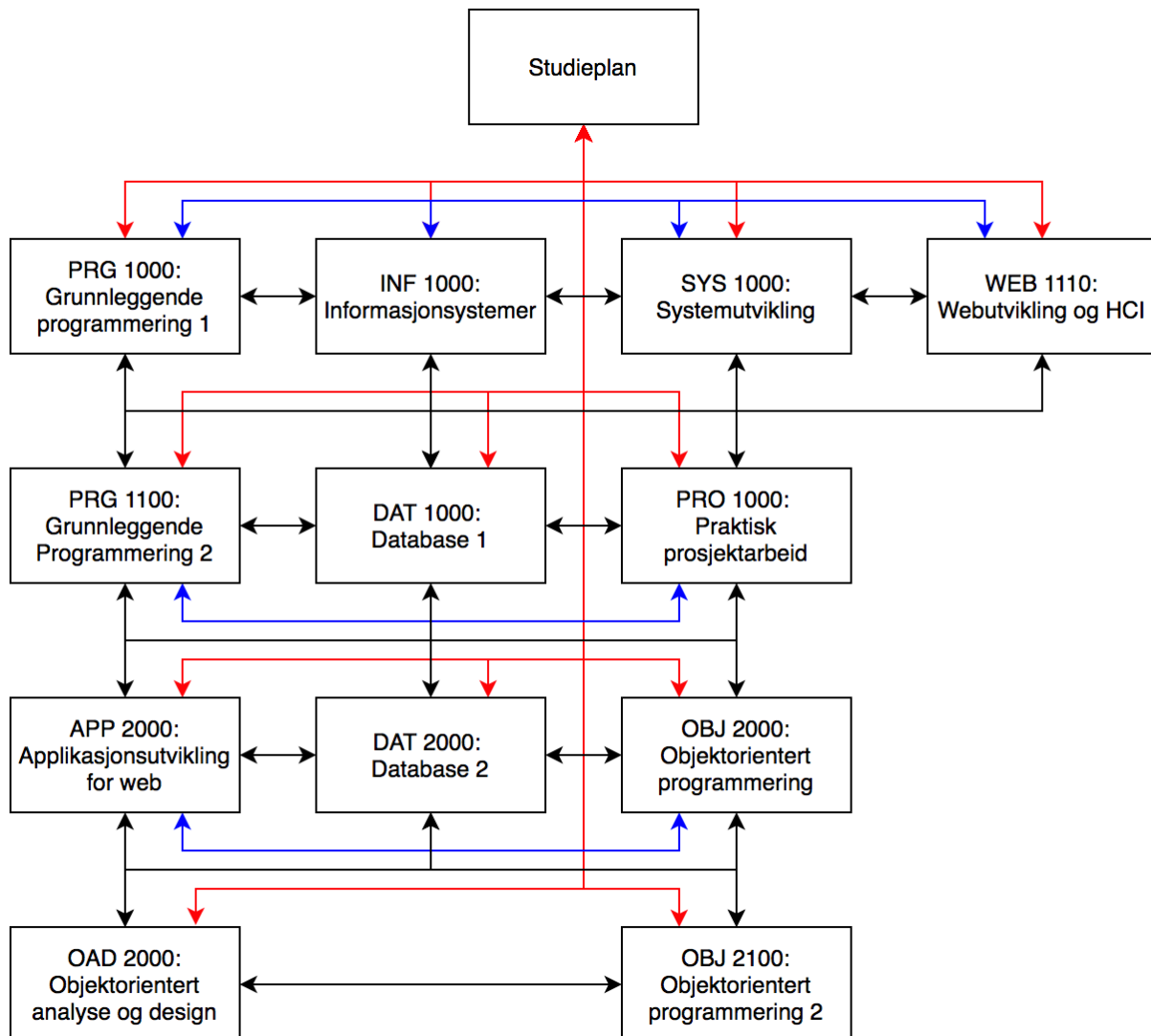
Oppgave 5

Hovedillustrasjon hypertekst-struktur for presentasjon av emner på bachelorstudiet i IT og informasjonssystemer:

Svart linje er lenking mellom enkeltsider.

Rød linje er lenking fra studieplan til andre sider

Blå linje er lenking mellom sider på samme nivå



Alle emnene kan nåes fra studieplan.

Alle emner per semester kan nåes fra andre emner i samme semester.

Emene fra semester kan nåes fra emnene fra semester 2, og motsatt.

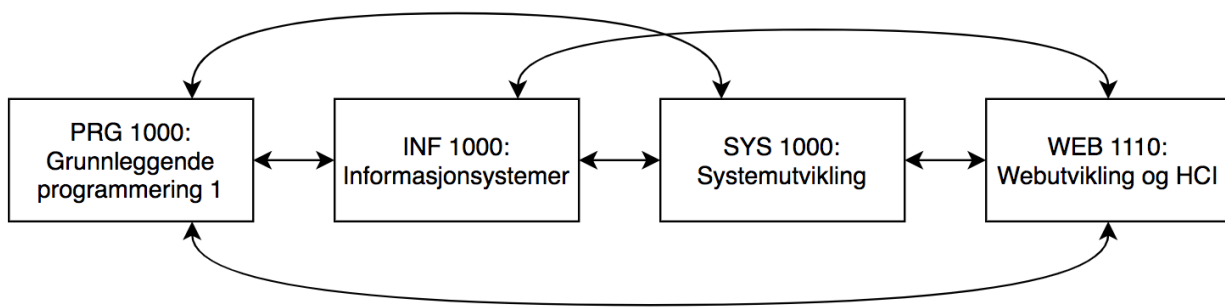
Det er også satt opp at emner som bygger på hverandre lenker til hverandre: Se PRG 1000 & PRG 1100 for eksempel. Et annet eksempel er DAT 1000 og DAT 2000.

Grunnen til at jeg ville satt opp hypertekst-strukturen sånn er at det er en naturlig måte å komme seg rundt til ulike emner på. Fra "Studieplan" kan man se og gå til alle emnene på de ulike

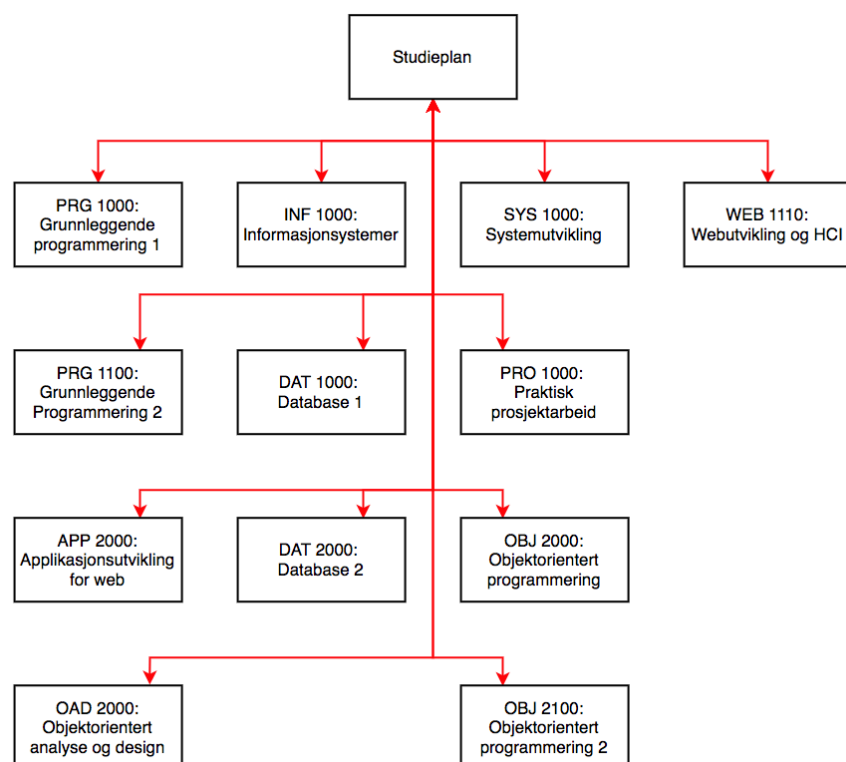
semesterene. Fra en hvilket som helst emneside i et gitt semester kan man også gå til og finne informasjon om de andre emnene man skal ha det året.

Fra temaer som går over flere semestre, men er delt opp i 2 emner er det også naturlig å ville finne informasjon om hva man lærer de ulike forskjellene, og hva det forventes å ha kunnskap om etter endt emne 1 (For eksempel Grunnleggende programmering 1 og Grunnleggende programmering 2.)

Del-illustrasjon 1: Lenker mellom alle emnene i ett semester isolert



Del-illustrasjon 2: Lenking fra studieplan til alle emnesider isolert



Oppgave 6

En god konseptuell modell er en modell som treffer med brukerens mentale modell. Det brukes gode metaforer og bruk av systemet er enkel og forståelig, kontra vanskelig men funksjonsrik. Den konseptuelle modellen burde være tydelig og samsvare med brukers mentale modell. Den burde også tilpasses brukergruppen -Hvis brukergruppen er homogen kan systemet være avansert og i noen grad lite forklarende, mens hvis brukergruppen er heterogen og systemet skal brukes av allmennheten så må systemet være enkelt forståelig for personer med liten erfaring ved bruk av systemer.

Metaforer er en av de beste teknikkene for å få bruker til å forstå hvordan systemet fungerer. Ikonene kan bygge på generelle erfaringer som de fleste har - for eksempel at hvis man skal ringe så er det et symbol for et telefonrør. Dette er noe de fleste er kjent med, selv om nyere generasjoner aldri har brukt et slikt "gammeldags" telefonrør. Symboler som går igjen i andre kjente systemer som bruker kanskje har forholdt seg til før er greie å bruke. Man kan også eliminere brukerfeil ved bruk av metaforer, som for eksempel når det kommer til innskriving av datoer i et skjema eller i en registreringsboks. I stedet for å la bruker skrive inn datoen som tall (30/11/2020) så kan bruker for eksempel velge dato i en virtual kalender. Da unngår man problemet med ugyldig input fra bruker - For eksempel at datoen er skrevet inn som dd/mm/yy i stedet for mm/dd/yy eller dd/mm/year. Man unngår også problemstillingen med at datoen skulle skrives inn med punktum i stedet for skråstrek.

Det er også viktig å være bevist på hvordan bruker oppfatter og leser informasjon. "Gestalts principles of perception" er viktig å være oppmerksom på. Gruppering av informasjon må oppfattes som logisk, det må være tydelig skille mellom bakgrunn og forgrunn og man må unngå blandede signaler. Hvis det ikke er samsvar mellom noen av disse punktene blir systemet forvirrende for brukeren og man bommer med den konseptuelle modellen.

For å treffe med sin konseptuelle modell må man være oppmerksom på at man utvikler systemet for brukere, ikke for seg selv som utvikler. De mentale modellene til brukere av systemet er vidt forskjellige, ufullstendige og er basert på brukers tidligere erfaringer og data-ferdigheter. For å verifisere at den konseptuelle modellen fungerer og stemmer overens med brukers forventninger så kan man observere brukere som tester systemet. Det er viktig at man ikke kun spør bruker om deres mening om hvordan systemet fungerer, men faktisk får se en bruker som prøver å bruke systemet. Det kan være stor forskjell på hva en bruker sier og hva en bruker faktisk gjør (Sandnes kapittel 9, side 234). Brukertesting blir altså en sentral del i verifiseringen av at den konseptuelle modellen fungerer. Gjennom observasjon, samtaler med bruker og andre teknikker for brukertesting vil man finne ut hva som fungerer og hva som ikke stemmer overens med en brukers mentale modell samt brukers forventning om hvordan systemet skal fungere.