

## Afleveringsopgave

# 07.01.01: Statisk site

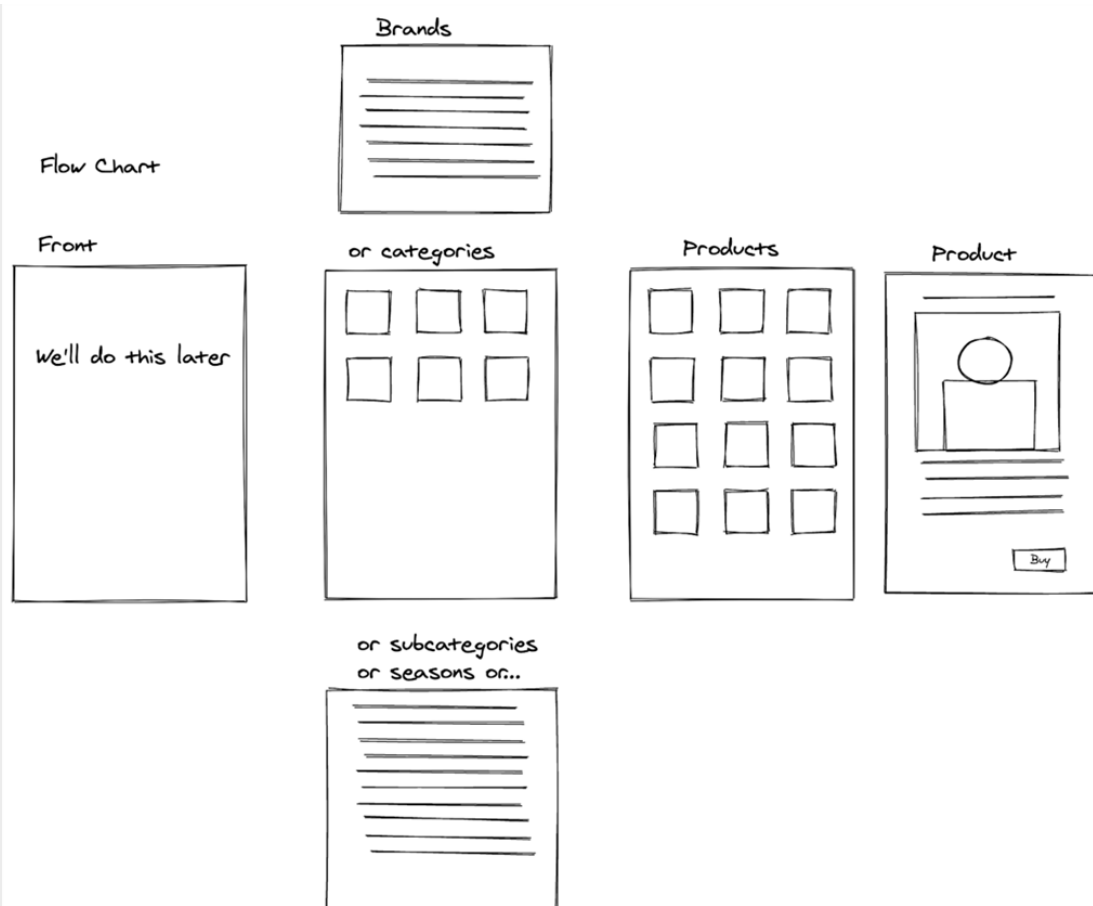
Denne opgave er at bygge en statisk hjemmeside til vores tøjbutik. I denne sammenhæng betyder statisk, at indhold, billeder osv. ikke ændres, og der er ingen integration med en database.

I løbet af de næste par uger vil vi gøre det dynamisk, så alle produkter hentes fra en database, hvilket betyder, at siden opdateres automatisk, når data ændres. Men i første omgang skal du lave fire statiske HTML-sider:

- produkt.html til visning af detaljerne for et enkelt produkt.
- produktliste.html til at vise en liste over produkter.
- kategori.html til at vise en liste med kategorier (fx brands, sæsoner, underkategorier eller lignende - [se evt. mulighederne her](#))
- index.html med links til de øvrige sider.

Så flowet bliver noget i stil med:

- Brugeren er på kategori-siden og klikker på en kategori.
- Brugeren klikker på en kategori og ser en liste over produkter på produktliste-siden
- Når brugeren klikker på et produkt, sendes de til produkt-siden der viser flere detaljer om det valgte produkt.



I denne omgang vil der som sagt kun være en statisk udgave af hver af siderne, så uanset hvilken kategori eller hvilket produkt der klikkes på, vises den samme side (senere, når du har lært at hente data fra en ekstern database dynamisk, vil det komme til at virke som beskrevet i flowet).

## produkt.html

Eksempeldata: <https://kea-alt-del.dk/t7/api/products/1163>

Flere forskellige billeder i to forskellige formater og opløsninger:

<https://kea-alt-del.dk/t7/images/webp/640/1573.webp> (Brug dette format)

<https://kea-alt-del.dk/t7/images/webp/1000/1533.webp>

<https://kea-alt-del.dk/t7/images/jpg/640/1552.jpg>

<https://kea-alt-del.dk/t7/images/jpg/1000/1580.jpg>

Husk, at dette kun er et udsnit af dataene, så du har en idé. Denne enkelte webside vil (snart) kunne vise alle de 44.000 produkter, der er i databasen!

Du får **adgang til andre billeder ved at skifte produktets id ud** i eksemplerne ovenfor (find andre id'er her <https://kea-alt-del.dk/t7/api/products?limit=50&start=10>) fx <https://kea-alt-del.dk/t7/api/products/1163>

[del.dk/t7/images/webp/640/1533.webp](https://del.dk/t7/images/webp/640/1533.webp) eller <https://kea-alt-del.dk/t7/images/webp/640/1535.webp>

## Udfordringer

Husk at tage højde for (og præsentere visuelt), at produktet kan

- være udsolgt
- har en rabatpris

## produktliste.html

Eksempeldata: <https://kea-alt-del.dk/t7/api/products>

Denne side skal vise en liste over produkter. Prøv at være konsekvent. Opret layoutet af et enkelt "kort", kopier det og skift dataene. Stop, når du har 3-4 produkter, og kopier alt et par gange, så du ved, hvordan dit websted opfører sig med masser af produkter.

Hvert kort skal linke til product.html

## Udfordringer

Husk at tage højde for (og præsentere visuelt), at produkterne kan

- være udsolgt
- have en rabatpris

## kategori.html

Eksempeldata, vælg en af følgende kategorier:

Kategorier: <https://kea-alt-del.dk/t7/api/categories>

Underkategorier: <https://kea-alt-del.dk/t7/api/subcategories>

Mærker: <https://kea-alt-del.dk/t7/api/brands>

Årstider: <https://kea-alt-del.dk/t7/api/seasons>

I en perfekt verden ville brugeren være i stand til at filtrere og navigere på vores side ved hjælp af alt det ovenstående, men jeg tror, vi kan blive enige om, at verden ikke ligefrem er perfekt i disse dage. Så vælg en og opbyg en slags liste, når brugeren kan vælge noget at filtrere efter.

- Alle links herfra skal pege på productlist.html

## Udfordringer

- Nogle af disse lister er ekstremt lange. Brug ikke alle data nu, copy/paste i stedet, når du har din struktur.
- Hvordan kan du præsentere disse simple data på en interessant måde?
- Skab dit design, så alle disse forskelle kan vises.
- Tip: Lav en klasse for hver mulighed, som du tilføjer til produktet.

## UI-krav

Brug CSS grid og flexbox til at layoute webstedet og listerne.

Husk, du kan have grid inde i grid, grid inde i flex, flex i flex osv. At have flere simple grids er ofte nemmere end et stort.

## Tips og gode ideer

- Husk at gøre dit design klar til den dynamiske del, gør produkterne så ens som muligt og noter forskellene mentalt.
- Brug så mange relevante klasser som muligt, en udsolgt klasse for produkter, der er udsolgt, en nedsat klasse for produkter, der er på rabat osv.
- Tilføj kommentarer til dig selv, for eksempel hvis der er et element, der bør fjernes under nogle omstændigheder.

## Levering

Denne iteration er ikke en aflevering, men noget du vil arbejde videre med efter afleveringen.

**Hav den klar torsdag morgen, hvor vi skal til at gøre den dynamisk!**

## Feedback

Vi vil se et par stykker, tilfældigt udvalgte, i klassen.

Du skal bedømme dit eget svar, når du har indsendt det.