

Projekt Rapport



IKEA Family
erbjudanden

Upp till
25%



Studerande: Mana Pecar

Kurs: EHBSNH22 HTML5 och CSS3, EC Utbildning

Innehållsförteckning

1. Sammafattning	3
2. Inledning	3
3. Presentation av uppgiften.....	4
3.1. Index.html.....	4
3.2. Product-details.html	6
3.3. Checkout.html	8
4. Slutsats	9

1. Sammanfattning

Syftet med denna uppgift var att implementera det vi lärt oss under kursens gång och använda våra kunskaper för att fördjupa oss ytterligare i HTML5 och CSS3. Genom att slutföra ett större projekt så kan man testa sig själv att bygga upp en hemsida från grunden trots att vi hade ett manus på hur det ska se ut.

I sajten har jag skapat tre olika sidor med HTML och dess externa stylesheet med CSS. För att uppnå målet använde jag olika CSS metoder samt semantiska element i html i största möjliga utsträckning. Att använda semantiska element visade sig vara en bra lärdom för mig när man skulle navigera i djungeln av kod och jag kunde enkelt hitta dit jag ville eftersom taggarna var självförklarande. Dessutom har jag lagt några breakpoint för att sajten ska vara responsiv. I processen lärde jag mig också hur man använder ett versionskontrollsystem som git för att registrera och spåra ändringshistoriken för programmets källkod.

2. Inledning

Målet med denna uppgift är att skapa en kopia av en existerande sajt baserad på kunskaper av HTML5 och CSS3. Att använda de olika CSS attributen så som Grid och Flexbox samt positionering i praktiken för att vidare stärka sina kunskaper och få en större helhetsbild på hur HTML och CSS är uppbyggt och hör ihop. Planen är att börja med html och sedan bit för bit styla de olika elementen. För att försöka på mig de olika tillvägagångssätten valde jag att börja med PC first istället för mobile first för att kunna få en bild på vad som är enklast.

3. Presentation av uppgiften

3.1 Index.html

Jag började med att skapa alla element som skulle vara med genom att använda mig utav devtools samt själv försöka lista ut vad som behövdes på sidan genom att titta på den. Efter att ha radat upp alla element så började jag jobba på CSS:en.

I headern använde jag en UL lista och gjorde själva sökrutan "sticky" på positioneringen, det vill säga när man scrollar så ser man en del av headern följa med, precis som på IKEA:s hemsida. Testade dessutom att använda "fixed" vilket också blev rätt så snyggt om jag gjorde headers bakgrund vit, men var inte så IKEA hade gjort och ville få hemsidan att se ut som IKEA:s i största möjliga utsträckning.

Den lurigaste delen av förstasidan var att få griden att fungera så som IKEA har byggt upp deras hemsida. Med hjälp av devtools och övningsuppgifter som vi haft kunde jag få en idé hur jag kunde bygga upp griden så att den ser likadan ut.

För att bygga upp griden använde jag mig utav grid-template-area. Se nedan kodblock.

```
.image-grid{
  display: grid;
  grid-template-columns: 50% 1fr 1fr;
  grid-template-rows: 40% 1fr 1fr;
  grid-gap:25px;
  grid-template-areas:
    "img-1 img-2 img-3"
    "img-1 img-4 img-3"
    "img-1 img-4 img-5"
}
```

Genom att ange vilken position de olika bilderna skulle vara på kunde jag konstruera önskad gridform. Grid är ett kraftfullt verktyg som gör det möjligt för utvecklare att kunna uppnå en struktur där man kan placera olika element på olika positioner enkelt på hemsidan som hade varit väldigt svårt och krävande om man skulle använda exempelvis flexbox.



IKEA Family
erbjudanden
Upp till
25%



En annan knepig sektion var ”karusellen”. Genom att använda flexbox och för knappen position ”absolute” så fick jag önskad effekt. För att göra detta möjligt var jag tvungen att göra boxen relativ för att kunna positionera knappen där jag ville den skulle vara. När allt var fixat så var det bara att kopiera hela elementet och på så sätt skapa två till och bara ändra bild. Det var då jag insåg att alla bilder var olika storlek. Det löste jag med nedan kodrad som också gjorde bilderna responsiva.

```
carousel .flex-item img {
  width: 100%;
  height: 300px;
  max-width: 100%;
  margin-bottom: 25px;
```



Inspireras av höstens nyheter

Lorem ipsum dolor sit amet,
consectetur adipisicing elit.
Suscipit et odit laudantium
quia neque iste sequi
reprehenderit omnis
dolorum? Dicta.



Inspireras av höstens nyheter

Lorem ipsum dolor sit amet,
consectetur adipisicing elit.
Suscipit et odit laudantium
quia neque iste sequi
reprehenderit omnis
dolorum? Dicta.



Inspireras av höstens nyheter

Lorem ipsum dolor sit amet,
consectetur adipisicing elit.
Suscipit et odit laudantium
quia neque iste sequi
reprehenderit omnis
dolorum? Dicta.



På index.html sidan har jag använt 2 breakpoints (900px och 1200px). Jag tyckte att det var mycket enklare att göra mediaqueries med PC first metoden eftersom det gjorde allt smidigare att fixa med hjälp av olika flex-direction, margins och paddings. När det är mindre än 900px så skulle vissa saker försvinna med hjälp av display: none och två andra element dyka upp. Griden fick också en annan look och utseendet på footern mobilanpassades också.

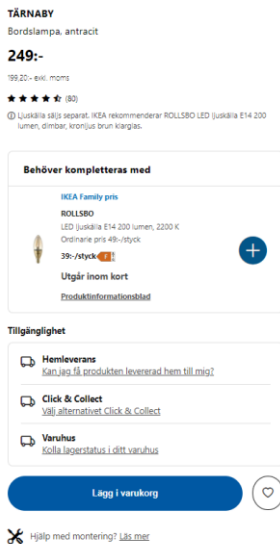
3.2 Product-details.html

Jag använde mig utav samma teknik som jag gjorde i index.html genom att skriva all html innan jag började med css. Här var det lite enklare då jag kunde återanvända både headern och footern samt kunna använda deras mediaqueries. Det som jag var tvungen att göra var att göra en grid till bilderna och produkt detaljerna.

```
.hero .grid-container{
  display: grid;
  grid-template-columns:repeat(3, 1fr);
  grid-template-rows:repeat(4, 570px);
  grid-gap:25px;
  margin:0 0 40px;
  grid-template-areas:
    "img-1 img-2 add-to-cart"
    "img-3 img-4 add-to-cart"
    "img-5 img-6 add-to-cart"
    "img-7 img-8 add-to-cart";
}
```

Jag använde mig utav liknande teknik som jag gjorde i index.html hero:n men med en lite annorlunda tvist. Som man kan se ovan så la jag hela ".add-to-cart" klassen på höger sida för att få till designen på samma sätt som IKEA har.

I delen som man kan se på nedan bild var det klurigt att få allt att se samma ut som på ikea. För att styra hur man skulle vilja ha det var jag tvungen att styla alla komponenter individuellt med hjälp av :first-child och :nth-child(). Man skulle nog kunna göra det på ett annat sätt men för att följa strukturen som jag planerat att göra med att använda så mycket selektorer som möjligt så utmanade jag mig själv genom att försöka nå de olika elementen med hjälp av selektorer.



Knappen är ”animerad” om man hoverar över den så byter den färg. Det gjorde jag genom att använda :hover och kopiera in samma attribut som den vanliga stylingen.

```
transition-property: opacity,transform;
```

```
transition-duration: 250ms;
```

```
transition-timing-function: cubic-bezier(0.4, 0, 0.4, 1);
```

Jag använde mig utav ovanstående transition för att få önskat resultat. Dessa värden kunde jag hämta från IKEA:s hemsida för att göra knappen exakt samma så som IKEA:s utvecklare har gjort.

När det kommer till mediaqueries så räckte det med en breakpoint för jag hade redan gjort elementen någorlunda responsiva.

Själva hero:n ändrade jag både grid kolumnerna, raderna och hur de skulle vara positionerade.

Jag kunde ha gömt alla bilder för att bara använda en så som ikea gjorde (de hade en karusell) där man swipar eller trycker höger eller vänster knapp som fanns på bilden för att se de andra bilderna.

```
.hero .grid-container{
  display: grid;
  grid-template-columns: 1fr;
  grid-template-rows: repeat(9,1fr);
  gap: 15px;
  grid-template-areas:
    "img-1 "
    "img-2 "
    "img-3 "
    "img-4 "
    "img-5 "
    "img-6 "
    "img-7 "
    "img-8 " ;
}
```

Att kunna styra de olika elementen när en viss breakpoint nås var mycket intressant då man kunde manipulera koden till att bli responsiv. Att använda mobile first skulle ha medfört lite mer tid på att få allt att se rätt ut och det hade varit mycket kod i de olika mediaqueries. Det blev inte så mycket mediaqueries då all kod var redan skriven hur de olika elementen skulle se ut. Det som var kvar var att manipulera de till önskad mobilanpassad version.

3.3 Checkout.html

Samma som de två föregående html filerna skrev jag först html:n och sedan började jag styla elementen. Denna sida var inte så krävande som de andra men det enda som var det stora problemet var att få till griden. Efter att ha läst officiella dokumentationen kunde jag komma fram till en slutsats där jag använde nedanstående kodblock för att få till önskad layout. Att kunna använda punkter är att lämna raden tom.

```
.layout-grid{
  margin-top: 40px;
  margin-left: 60px;
  display:grid;
  grid-template-columns: repeat(8,minmax(0,1fr));
  grid-template-rows: 300px 50px auto;

  grid-template-areas:
    "bi-1-circle      first-step  first-step first-step      . order order order"
    ".                .          .          .                  . order order order"
    "bi-2-circle-fill second-step  second-step second-step .   .   .   . ";
}
```

Anledningen till att jag delade upp hela sidan i ovanstående grid var att jag ville få formuläret att hamna på exakt samma position så som IKEA hade gjort och kunde få till det att se fint ut. Allt annat var inte så krävande då jag redan hade footern samt återanvände själva headern men tog bort allt förutom loggan.

Jag använde mig utav table elementet på själva beställnings översikten och kunde få till hela layouten smidigt utan större bekymmer. Jag hittade ett intressant attribut (table:cell) som gjorde i princip hela arbetet åt mig.

För att besökare ska kunna ange sin information skapade jag ett formulär med olika element. Exempelvis använder jag ett input-element för att ange namn och adress. I input-elementet kan man lägga olika typ attribut. De typ attributen gör att input-element kan renderas på flera olika sätt. Jag valde text attribut för namn och adress och checkbox attribut för checkboxen. Om man bara vill välja ett objekt från flera alternativ kan man antingen skapa en alternativknapp eller en rullgardinsmeny. Den här gången skapade jag en rullgardinsmeny (dropdownmeny) med ett select-element för att ange land och telefonnummer eftersom det finns många alternativ och dessa behöver inte synas hela tiden. På varje menyval skapas med ett option element och i elementet lägger man value attributet där skriver man in ett alternativ.

```
.order {  
  position: sticky;  
  z-index: 80;  
  top: 1.5rem;  
  border: none;  
  margin: 0px;  
}
```

Jag använde mig utav sticky positioneringen för att beställnings översikten skulle följa med när man scollar. Dessa värden kunde jag ta genom att titta hur IKEA hade gjort med hjälp av devtools.

4. Slutsats

Jag använde inte någon av de olika ramverken för att förenkla arbetet då jag ville utmana mig själv och lära mig hur man kan använda de olika CSS attributen. Det är enkelt att bara ta en kodrad och sedan fixa och trixa men enligt min åsikt så faller mycket kunskap bort på detta vis. I framtiden tänker jag använda mig av exempelvis bootstrap, tailwind o.s.v. eftersom jag nu fått en lite mer övergripande kunskap för jag själv har behövt skapa alla sektioner. Jag var lite kluven om jag skulle göra alla tre sidor då det kändes väldigt överväldigande men nu i efterhand är jag glad att jag fortsatte skapa de två resterande sidorna då jag känner att jag fått jättemycket kunskap och sett själv min utveckling jämfört med när jag började på projektet. Ju mer jag skrev desto snabbare blev jag på att styla de olika komponenterna. Detta projekt har varit väldigt givande och lärorikt