

Webbteknik för Ingenjörer

Html & CSS: G-nivå

VT 2021

Kursansvarig

Keni Ren: keni.run@umu.se

Handledare

Stefan Berglund: stefan.berglund@umu.se

Siyu Luan: siyu.luan@umu.se

Introduktion

HTML (Hyper Text Markup Language) är ett språk för att grafiskt skapa webbsidor på internet med hypertext. HTML berättar för datorn hur sidan ska se ut och vilket innehåll som ska visas.

CSS (Cascading Style Sheet) är ett språk som berättar hur html elementen ska stolas med färg, typsnitt, font osv. Dessa två språk tillsammans med Javascript utgör det mesta av det vi ser på våra skärmar när vi surfar på webben.

I denna rapport beskrivs fortsättningen från min förra uppgift, hur jag tog min Lo-Fi prototyp gjord i Adobe XD och skapade den som en faktiskt klickbar webbsida på internet med hjälp av HTML och CSS.

Problembeskrivning

Uppgiften går ut på att utifrån min Lo-Fi prototyp skapa en webbsida motsvarande minst 5 sidor med HTML och sen stila elementen med hjälp av stilmallar i CSS med rätt standard samt följa de etablerade webbriktlinjerna.

Webbsidan ska vara responsiv och fungera på flera olika enheter samt inte tappa strukturen ifall man drar eller ändrar storleken på fönstret (Iphone 8 valdes som standard skärmstorlek för mobil-variationen). Hemsidan ska sedan laddas upp på en tilldelad webbserver och valideras via *W3C - Markup Validation Service*

Implementationen gjordes i utvecklingsmiljön Visual Studio Code och för att visa sidan användes främst webbläsaren Google Chrome.

Implementation

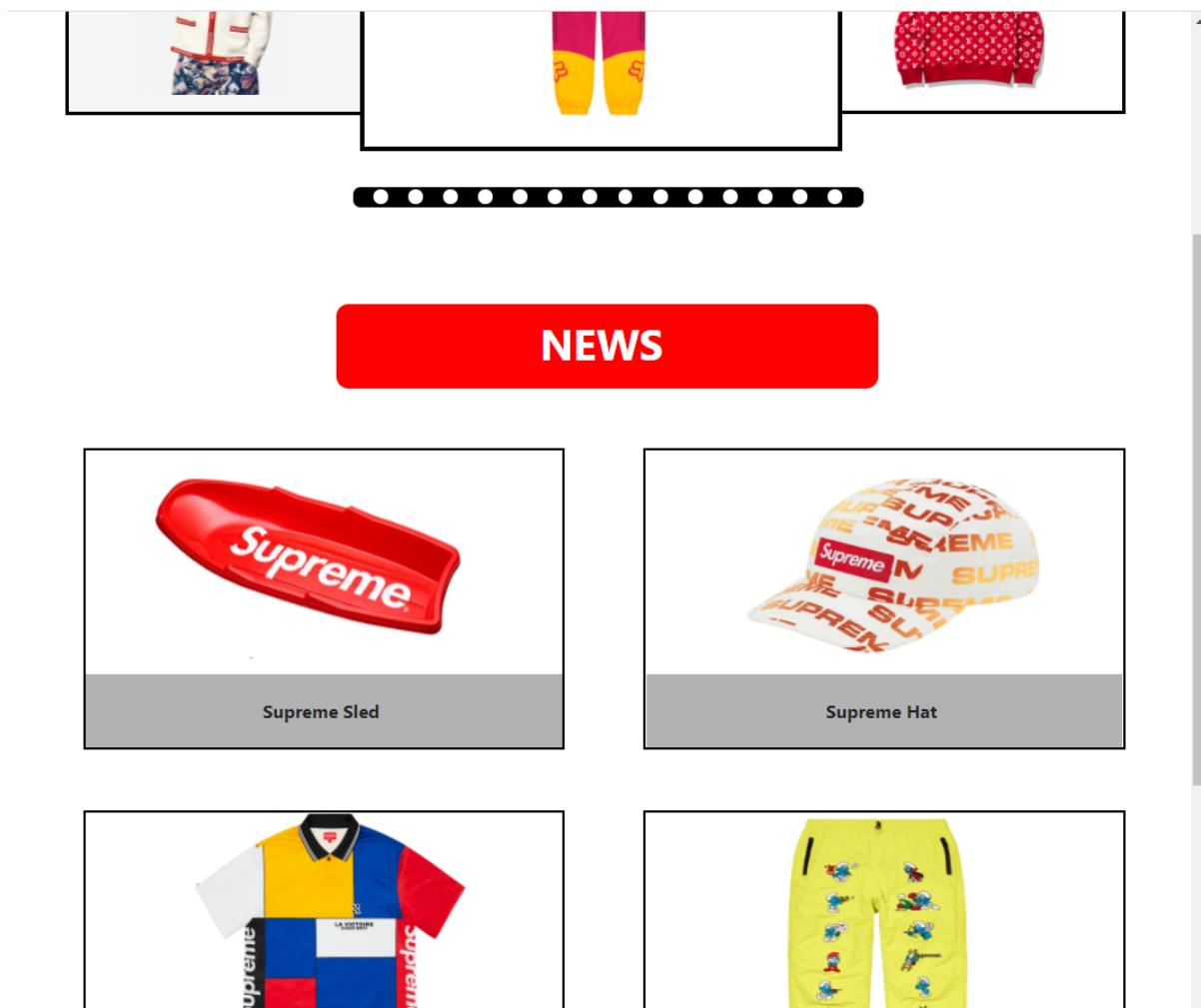
Det första jag gjorde var att göra min Index-sida. Jag använde min Lo-Fi prototyp som mall och började sätta upp mitt HTML dokument med META taggar och body. I bodyn började jag lägga in de "Headers" och "Footers" samt sökruta som fanns med i min Lo-Fi prototyp.



Figur 1: Lo-Fi prototyp mall

Jag använde mig av HTML och CSS parallellt och skapade stilens i mitt style.css samtidigt som jag skapade de olika sektionerna för att lättare kunna se hur allt ska flyta ihop. Det kluriga här var att sätta in bilderna och förstå hur jag skulle kunna ha en bild över en annan och dessutom vara inramad. Det tog lite tid att förstå men efter ett tag fick jag till det.

Jag visste att det skulle finnas mer på förstasidan än vad som visades i prototypen. Jag hade tänkt att det ska finnas populära/erbjudande/nya produkter i början som ska vara lätt att komma åt så jag la in några sådana som syns i figuren under.



Figur 2: Startsida News sektionen

När index-sidan var klar var det dags att börja på nästa. Eftersom Footern och Headern ska vara med på alla sidor kopierade jag helt enkelt dessa för att snabbt lägga in på alla andra sidor. Däremot la jag till på varje sida att stället man är på är ifyllt så att man vet vart på hemsidan man befinner sig.



Figur 3: Clothes är ifyllt eftersom användaren befinner sig där.

Den andra sidan jag gjorde var kläddes sektionen vilket var relativt lätt. Det svåra var att få in bilderna på rätt ställe och eftersom alla föremål har olika proportioner krävdes det mycket ändringar av varje individuellt element för att ge rätt storlek.

Vad jag valde att göra då var att istället för att använda mitt CSS-dokument skriva stilen direkt i bilden med style = " ". Jag gjorde detta för annars skulle det behövts en individuell klass för varje bild och eftersom jag har ungefär 60 olika bilder i hela webbsidan hade det behövts 60 olika klasser som alla är ungefär samma förutom att de har olika värden.

Detta betyder alltså att min webbsida både har CSS direkt i html-filerna samt i min Style.css mall, se nedan.

```
<div class = "row1">
<div class ="itemsl" id="indexsledimg">
    
    <div class ="itemtext">
        <p> Supreme Sled</p>
    </div>
</div>

<div class ="itemsl" id="indexhatimg">
    
    <div class ="itemtext">
        <p>Supreme Hat</p>
    </div>
</div>

iv>

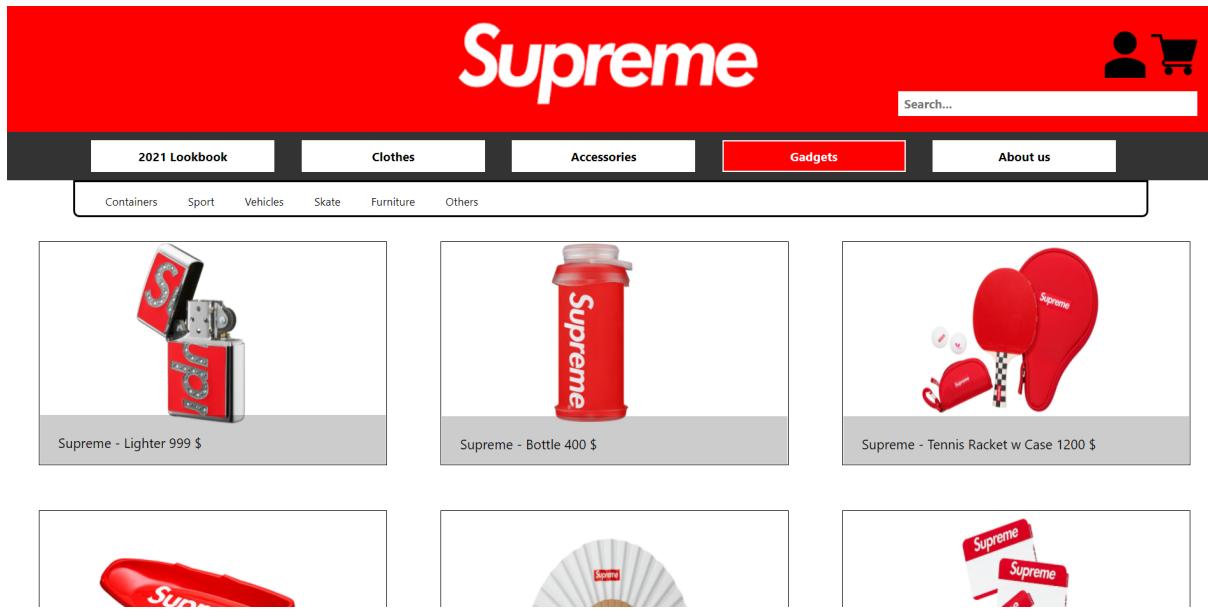
v class = "row2">
<div class ="itemsl" id="indextshirtimg">
    
    <div class ="itemtext" id="indextshirt">
        <p> Supreme T-shirt</p>
    </div>
</div>

<div class ="itemsl" id="indexsmurfpantsimg">
    
    <div class ="itemtext" id="indexsmurfpants">
        <p> Supreme Smurf Trousers</p>
    </div>
</div>
</div>

<div class = "row2" id="indexbeanieimg">
<div class ="itemsl">
    
    <div class ="itemtext" id="indexbeanie">
```

Figur 4: Skärmdump av html-koden för klädesplaggen, som man ser är css skrivet direkt i html filen med hjälp av <style>

I min Lo-Fi prototyp hade jag endast med en av produktsidorna vilket var clothes. Men eftersom Accessories och Gadgets har samma tema fast bara andra produkter valde jag att implementera dem också. Detta gjorde att jag kodade 7 sidor istället för 5.

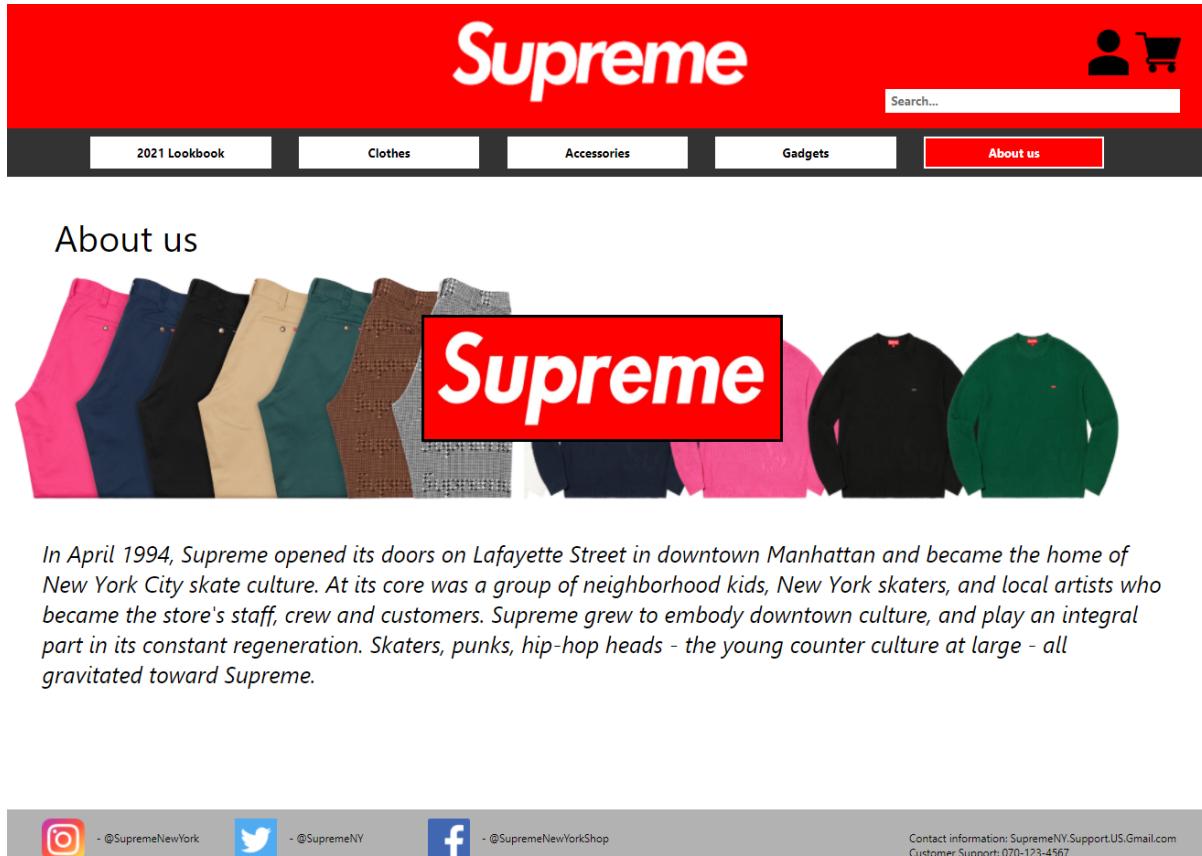


Figur 5: Skärmdump av gadgets-sektionen av webbsidan.

```
> Images  
<> 2021 Lookbook.html  
<> About us.html  
<> Accessories.html  
<> Clothes.html  
<> Gadgets.html  
<> Index.html  
<> Penguin Hoodie.html  
# phone-layout.css  
# print.css  
<> Shopping Cart.html  
# style.css
```

Figur 6: Skärmdump av alla filerna till webbsidan

Efter att jag hade gjort alla produktsidor gjorde jag “About us” sidan. Denna var väldigt lätt eftersom den enbart innehöll text och några bilder. Det svåra här var än en gång att sätta bilderna ovanför varandra och för detta använde jag z-index för att lägga en bild över en annan.

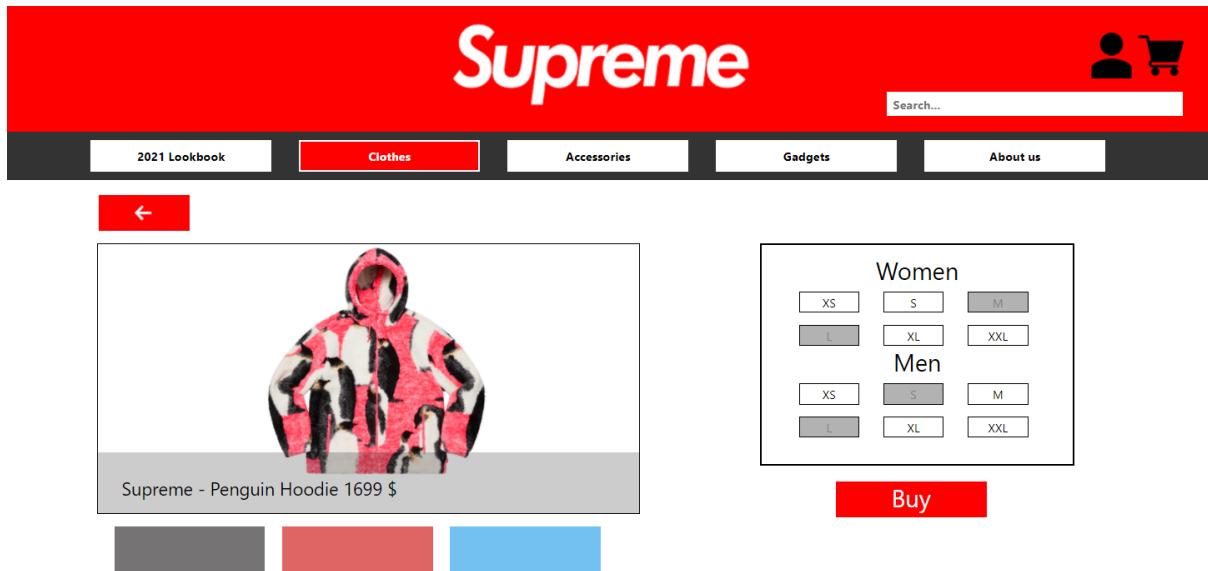


Figur 7: Skärmdump av About us sidan

```
<div class = "AboutUsPictures">
    <div class = "row13">
        
        
        
    </div>
</div>
```

Figur 8: Html-kod för Bilderna på About us sidan med z-index

När jag hade gjort klar detta var det endast ”inspektions-fliken” när man klickar på en produkt och kundvagnen kvar. Jag valde att göra pingvin-hoodien klickbar som tar en till en annan sida där man kan välja storlek, färg och köpa den produkten precis som jag hade gjort på min Lo-Fi prototyp.



Figur 9: Flik för att köpa/inspektera en produkt.

Vad jag valde att göra här däremot var att lägga till lite Javascript. Tanken är att ifall man klickar på en av färgerna ska hoodien byta färg till den färgen man klickade på. Detta hade gått att lösa genom att endast göra 2 nya identiska html sidor fast med en annan färg på hoodien som man kan bläddra mellan. Men ett mycket mera praktiskt och realistiskt sätt att fixa det på hade helt enkelt att använda Javascript. Jag lät till 3 funktioner i denna sida som ändrar bilden beroende på vilken knapp man klickar på. Se figurerna nedan

```
<script>
    function pictureChangeGray()
    {
        document.getElementById("redImage").src="Images/supreme22.PNG";
    }

    function pictureChangeRed()
    {
        document.getElementById("redImage").src="Images/supreme8.PNG";
    }

    function pictureChangeBlue()
    {
        document.getElementById("redImage").src="Images/Supreme23.PNG";
    }
</script>
```

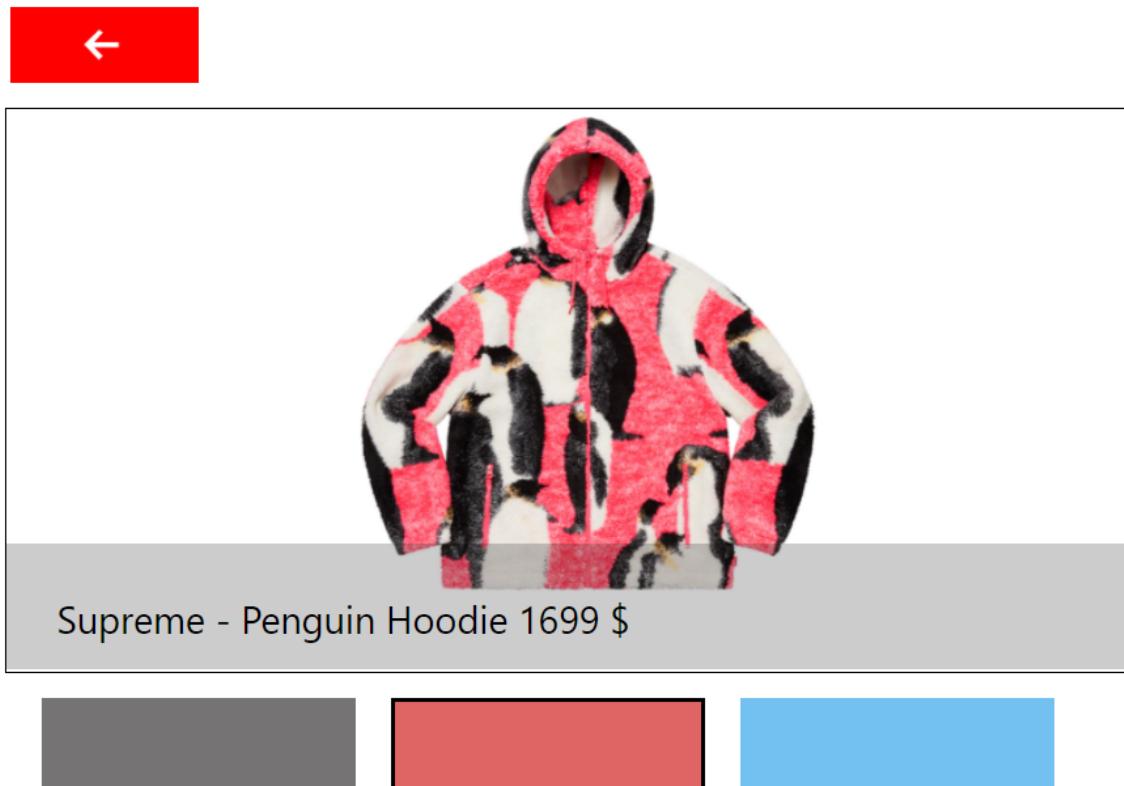
Figur 10: kod på scriptet som körs



Figur 11: Blå hoodie visas när den blåa knappen är tryckt.

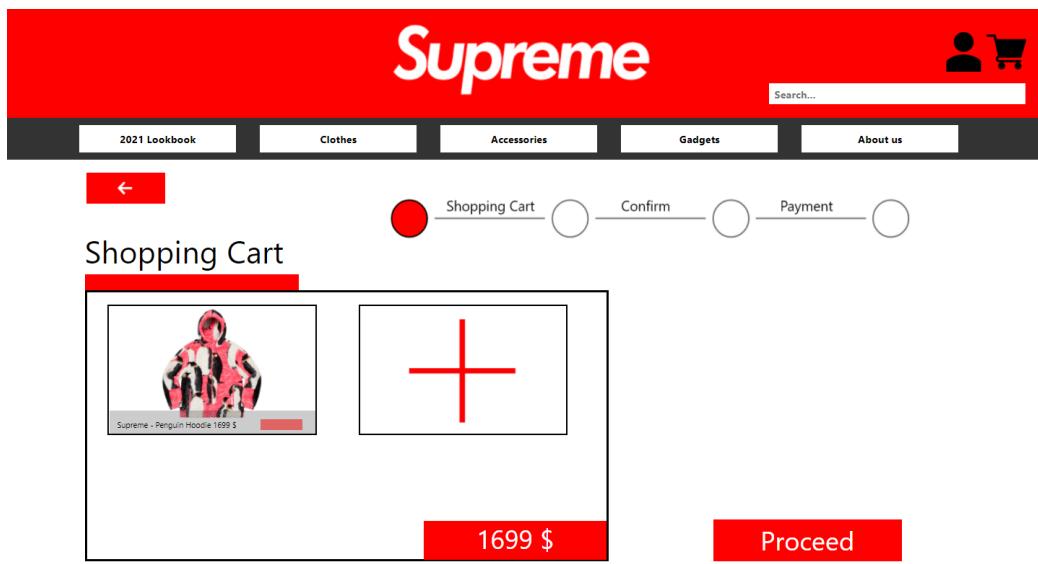


Figur 12: Grå hoodie visas när den gråa knappen är tryckt



Figur 13: Röd hoodie visas när den röda knappen är tryckt

Den sista sidan jag gjorde var kundvagnen som man kommer till om man antingen klickar på "Buy" knappen på den tidigare sidan eller om man i något av sidorna klickar på kundvagns-ikonen uppe i det högra hörnet. Denna sida var ganska lätt att göra då det för det mesta bara var en blandning av saker som jag hade gjort tidigare.



Figur 14: Skärmdump av kundvagnen

Nu var alla sidor klara, men det är mycket annat runt omkring som jag inte har nämnt ännu. Jag har aldrig jobbat med html tidigare och förstod inte att det inte är praktiskt att använda pixelvärdet. Jag visste att hemsidan skulle vara responsiv men jag såg det som ett problem jag kan lösa i framtiden. Ungefär 80% in i implementationsfasen fattade jag att jag måste göra om hela sidan för att den ska kunna vara responsiv. Jag blev tvungen att byta ut alla pixelvärdet mot vw som ändrar storleken förhållande till skärmen. Detta var extremt jobbigt eftersom jag fick gå in i varenda bild, ram och text och översätta alla värden till vw så att det skulle se likadant ut. Jag hade också använt en massa "position: absolute" och "position: relative" som gjorde att det endast såg bra ut i helskärm. Men ifall man drog i skärmen eller zoomade in så flög allting omkring.

Hade jag vetat mera om HTML och CSS innan hade jag förstås inte gjort dessa misstag och kodat rätt redan från början. När sidan var klar och responsiv var det dags att ändra utseendet för mobilen.

Många av flikarna och texterna går inte att se något bra med en mobil och alla knappar var mer eller mindre för små. Jag hade gjort en Lo-Fi prototyp för Mobilgränsnittet sen innan och det var den jag utgick från när jag skapade min mobilvariant.



Figur 15: Lo-Fi gränsnitt för mobil.

För att kunna ändra utseendet använde jag mig av @media screen med en max-width så att ifall fönstret är tillräckligt litet kan jag ändra utseendet på klasserna.

```
@media screen and (max-width: 601px) {  
    .topnav.responsive {position: relative;}  
    .circleborder button{margin-left: 1.7vw; margin-bottom: 0vw;}  
    .topnav {height: 30vw;}  
    .row1{margin-top: 5vw;}  
    .searchbar{position: absolute; height: 8vw; width: 70vw; left: 10vw; top: 15vw;}  
    .searchbar input[type=text]{height: 6vw; width: 65vw; margin-left: 2vw; margin-top: 1vw;}  
    .itemtext2{margin-top: -0.5vw; width: 80vw; height: 15vw;}  
    .itemtext2 p {font-size: 4vw; margin-top: 4vw;}  
    .circleborder{width: 95vw; height: 10vw; margin-left: 2.5vw; padding-left: 1vw;}  
    .circleborder button {width: 4vw; height: 4vw; margin-top: 3vw; margin-left: 2.5vw;}  
    .news{width: 95vw; height: 20vw; margin-left: 2.5vw;}  
    .news p{font-size: 17.5vw;}  
    .row1{display: block;}  
    .items1{width: 90vw; height: 45vw; margin-left: 5vw;}  
    .items1 img{width: 60% !important; height: 95% !important; margin-left: 7vw;}  
    .itemtext {width: 90vw; margin-top: -8vw;}  
    .itemtext p{font-size:5vw; margin-top: 6vw;}  
    .row2{display: block;}  
  
    .filter{height: 6vw; border-width: 0.5vw;}  
    .filter p{font-size: 2.5vw; margin-left: 4.8vw; margin-top: 1.5vw;}  
    .inspectItemText{margin-top: -1.25vw;}  
    .inspectItemText p{font-size: 2.5vw; margin-left: 1.5vw; margin-top: 1.5vw;}  
    .price{margin-top: 7vw;}  
    .footer{margin-top: 30vw;}  
    .row3{display:block; margin-left: 6.5vw;}  
    .row4{display:block; margin-left: 6.5vw;}  
    .clothItems{width: 80vw; height: 50vw;}  
    #supremelogo {height: 100% !important; width: 70% !important; margin-left: 20vw !important}  
}
```

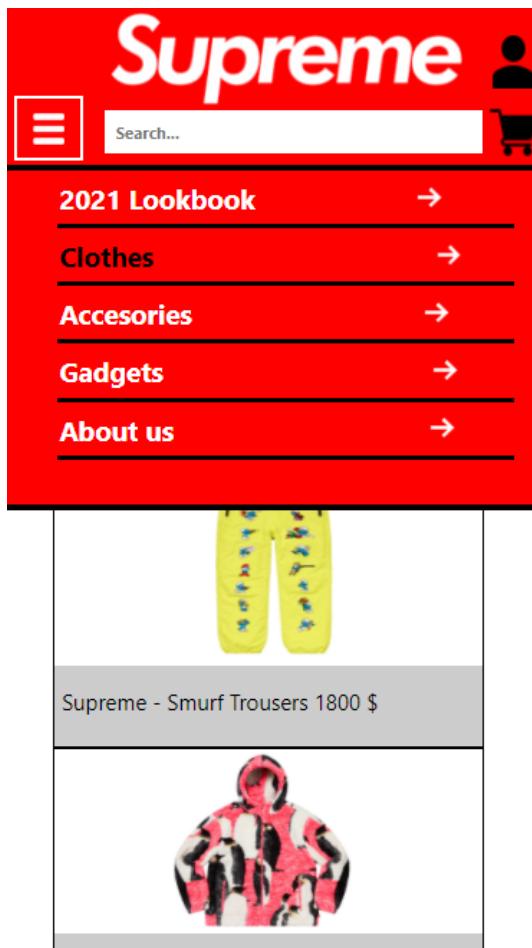
Figur 16: Skärmdump på css koden som ändrar klasserna när fönstret är visst litet, t.ex på mobiler.

Denna ändring tog extremt lång tid och jag blev tvungen att manuellt ändra storleken på nästan varenda bild så att proportionerna höll när man skalar ner till en mindre skärm.

Jag använde mig även av ett till javascript från w3schools för att lägga till en hamburg-meny som ska visas när man är på en mobil. Detta ska vara en knapp som öppnar och stänger en meny varje gång man klickar på den och ska inte vara synlig på större skärmar.



Figur 17: Mobil versionen av webbsidan med hamburg-menyn dockad i höger sida



Figur 18: Lista visas när man klickar på ikonen, och försvinner om man klickar igen.

```
<script>
function hamburger_menu() {
  var x = document.getElementById("thirdnavhide");
  if (x.className === "thirdnavhide") {
    x.className = "thirdnav";
  } else {
    x.className = "thirdnavhide";
  }
}
</script>
```

Figur 19: Scriptet som visar/döljer listan för varje klick på hamburg-menyn knappen

När allt detta var klart gjorde jag några sista ändringar och bad en kompis kolla på hemsidan och se ifall något inte var användarvänligt eller hade fel kontraster/typesnitt osv.

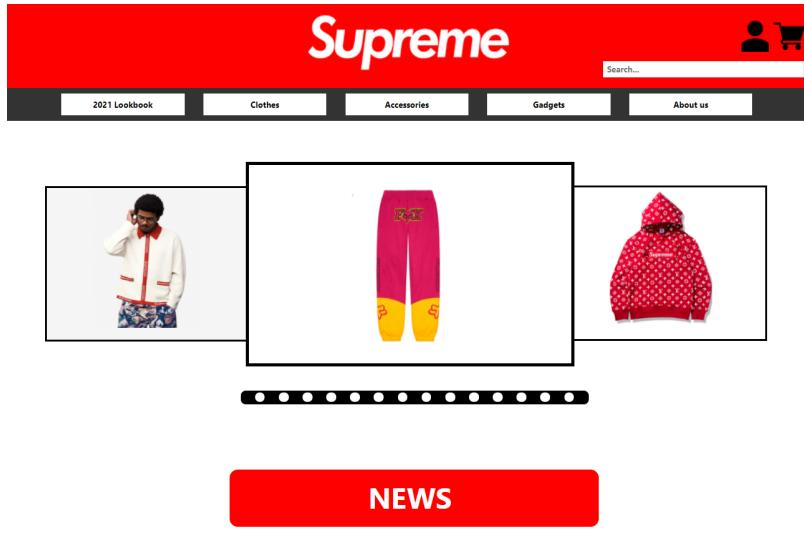
Sidan lades sidan upp på en webbserver och körde w3schools online valideringsprogram.

Resultat

Webb vs Smartphone

Här presenteras en rad skärmdumpar av de olika sektionerna i webbsidan på både Iphone 8 och Webb där det går att se skillnaderna och den responsiva designen. Bilderna visar förstås inte allt och det går inte att visa animationer/hovers. Så för en komplett överblick se länken nedan som tar en direkt till sidan.

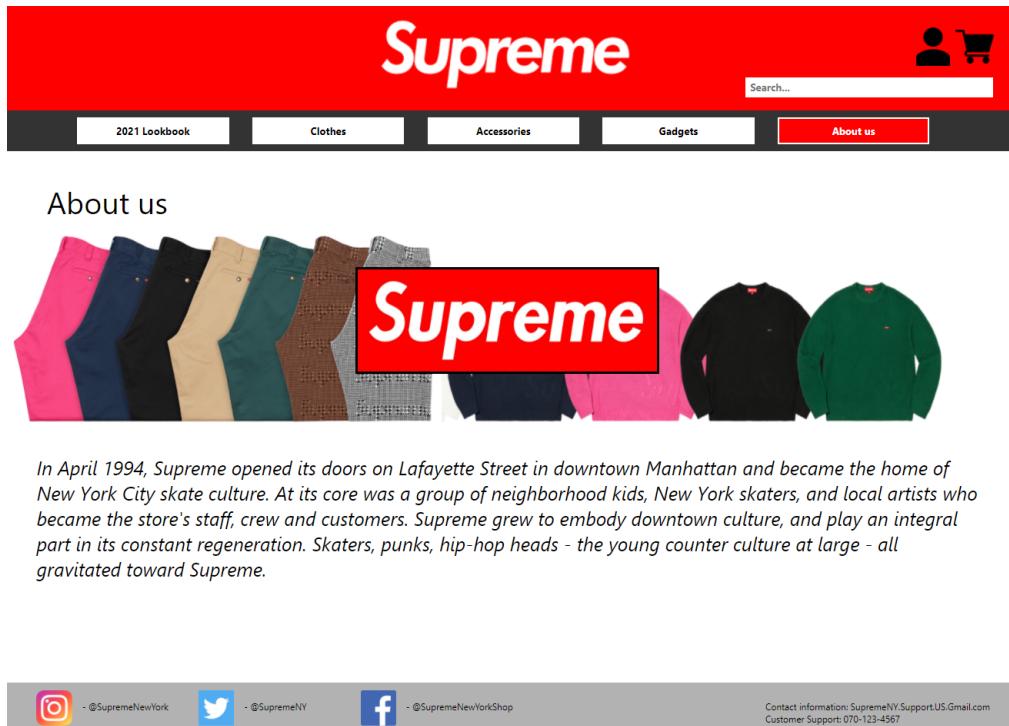
länk till sidan: <http://130.239.115.172/user076/Supreme>



Figur 21: Skärmdump av Index-sidan för webb



Figur 22: Skärmdump av index-sidan för iphone 8

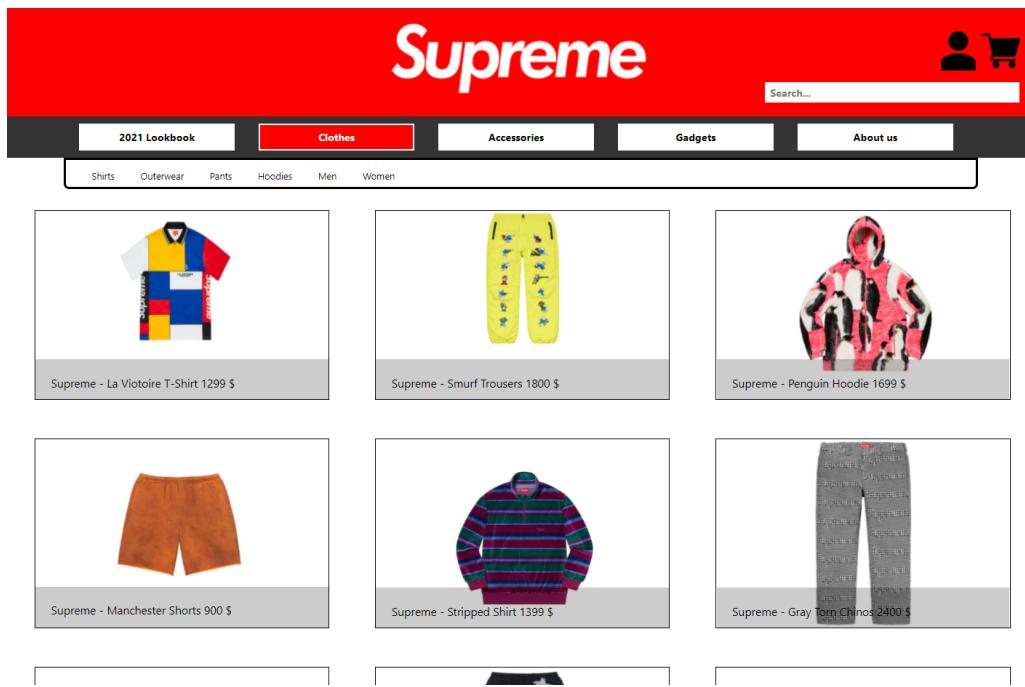


Figur 23: Skärmdump av "About us" sidan för webb

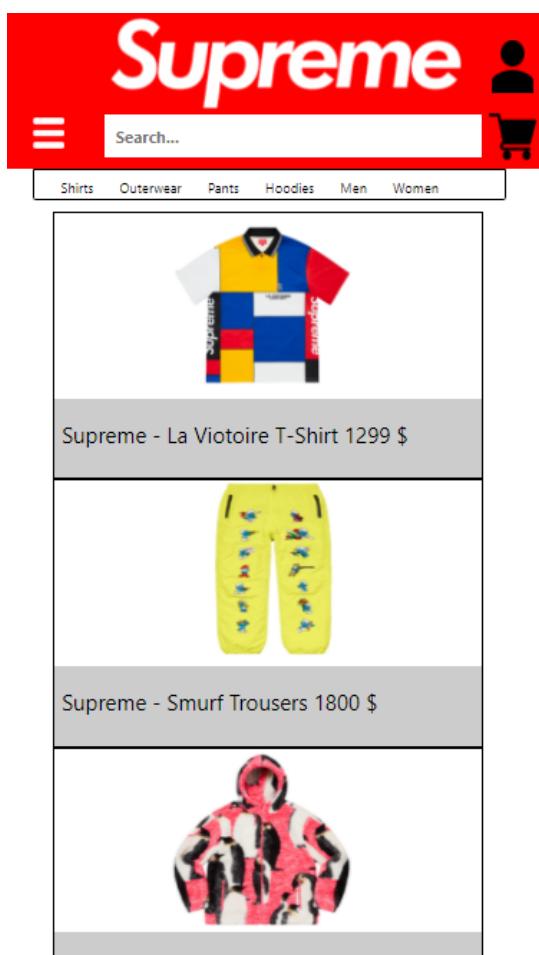


In April 1994, Supreme opened its doors on Lafayette Street in downtown Manhattan and became the home of New York City skate culture. At its core was a group of neighborhood kids, New York skaters, and local artists who became the store's staff, crew and customers. Supreme grew to embody downtown culture, and play an integral part in its constant regeneration. Skaters, punks, hip-hop heads - the young counter culture at large - all gravitated toward Supreme.

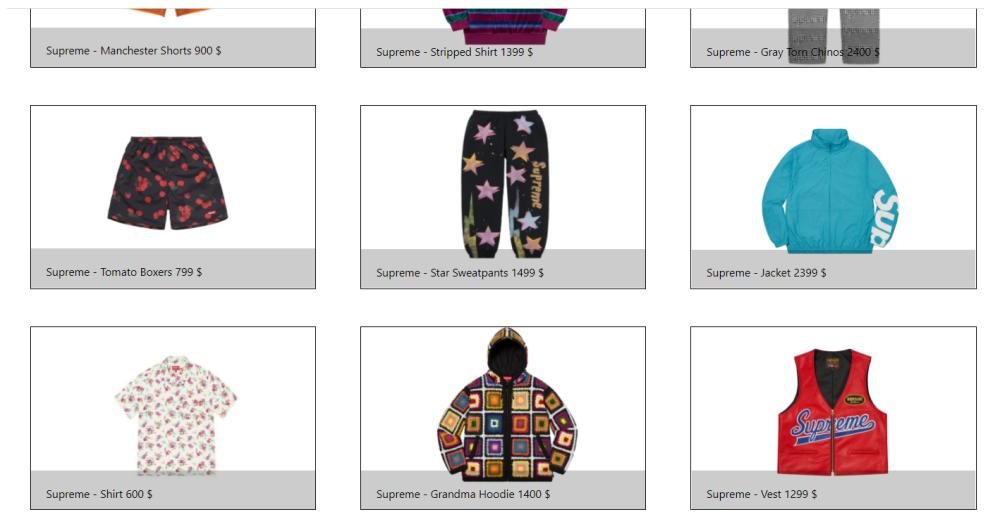
Figur 23: Skärmdump av "About us" sidan för Iphone 8



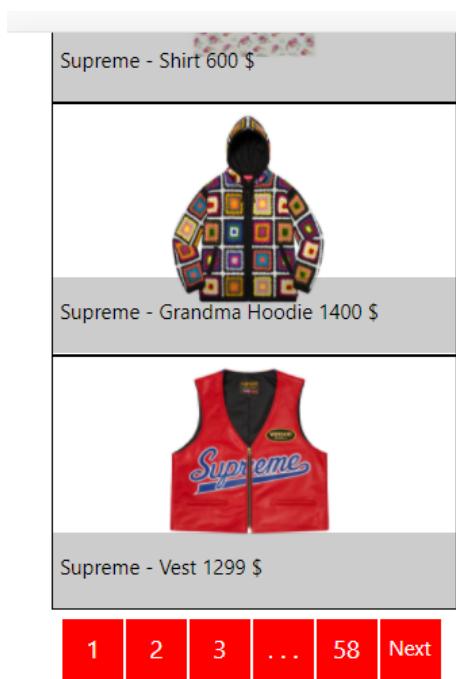
Figur 24: Skärmdump av "Clothes" sidan för Webb



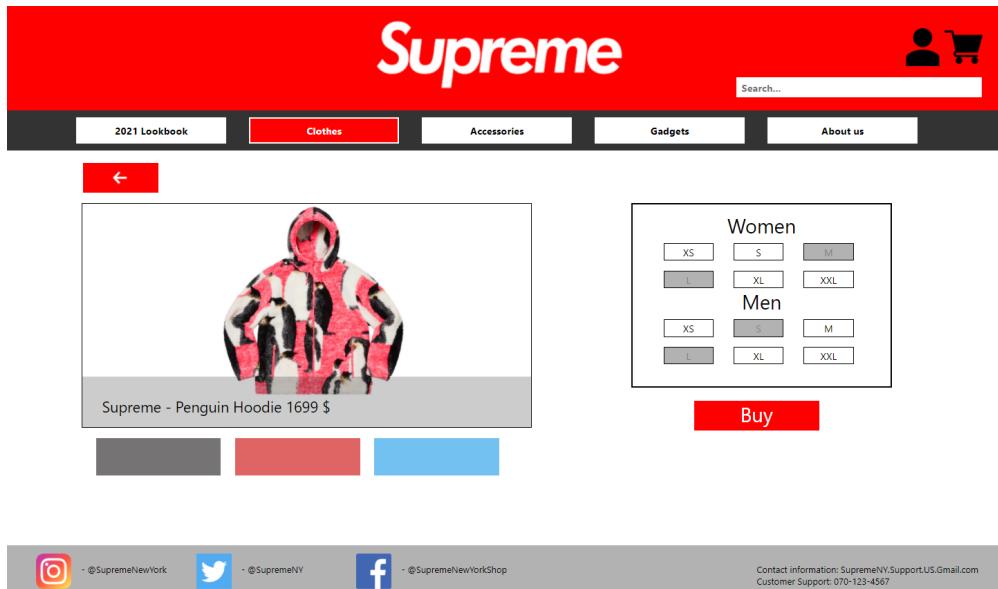
Figur 25: Skärmdump av "Clothes" sidan för Iphone 8



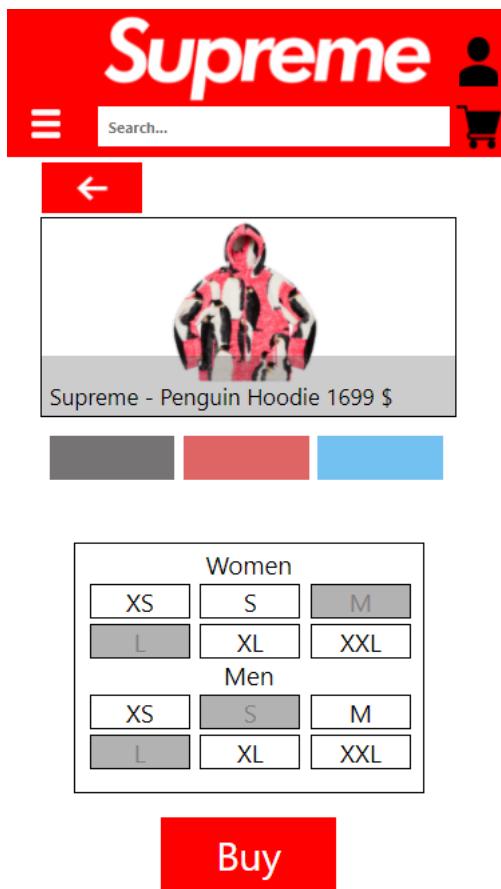
Figur 26: Skärmdump av botten av "Clothes" för webb



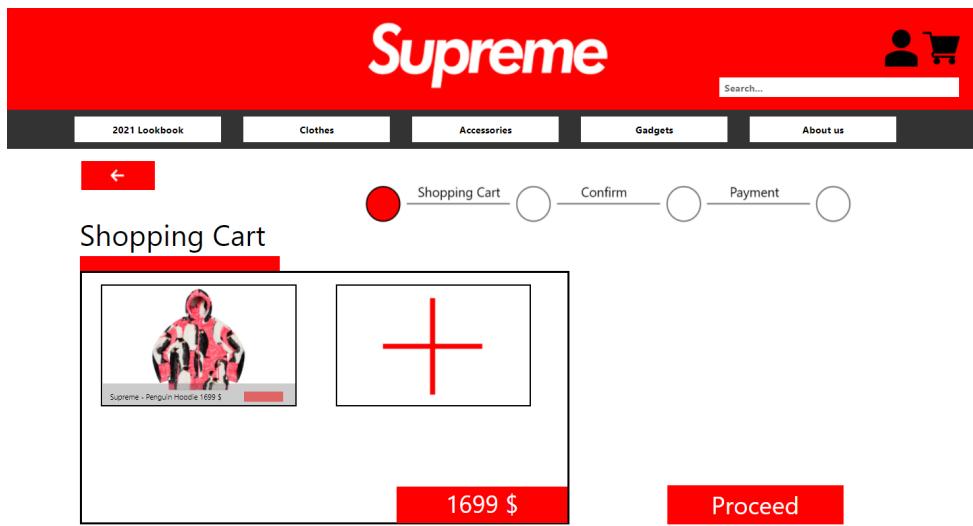
Figur 27: Skärmdump av botten av "Clothes" för iphone 8



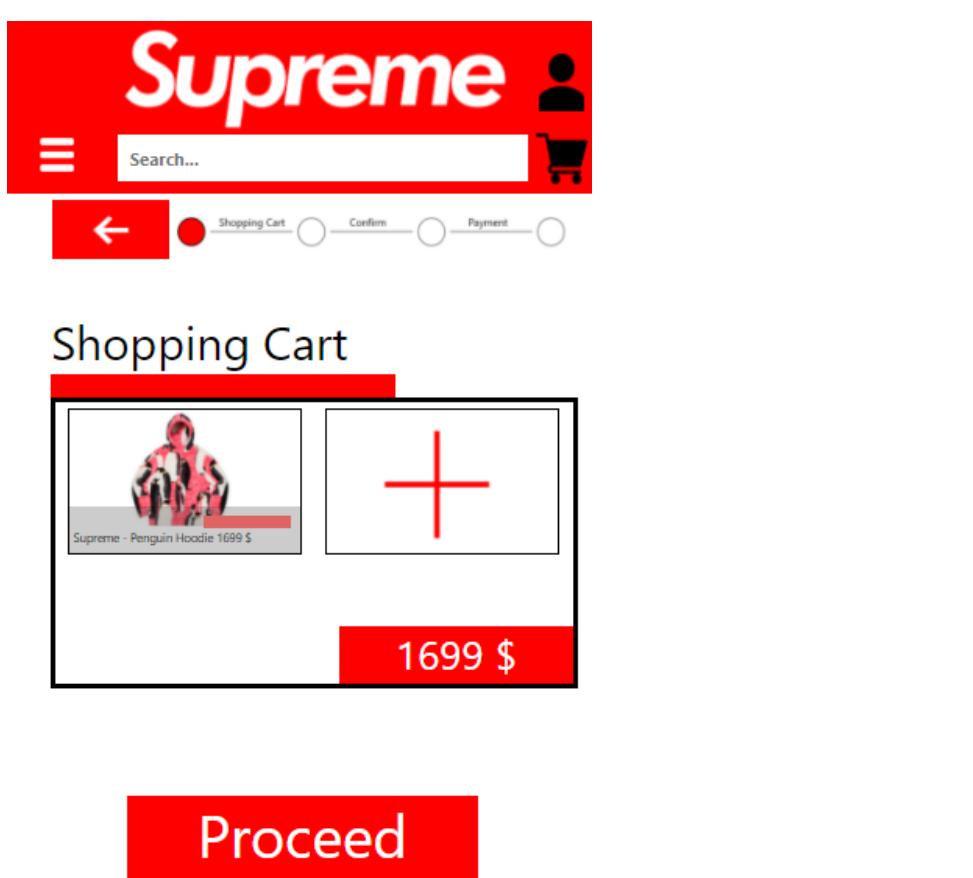
Figur 28: Skärmdump av "Pingvin hoodien" sidan för webb



Figur 29: Skärmdump av "Pingvin hoodien" sidan för Iphone 8.



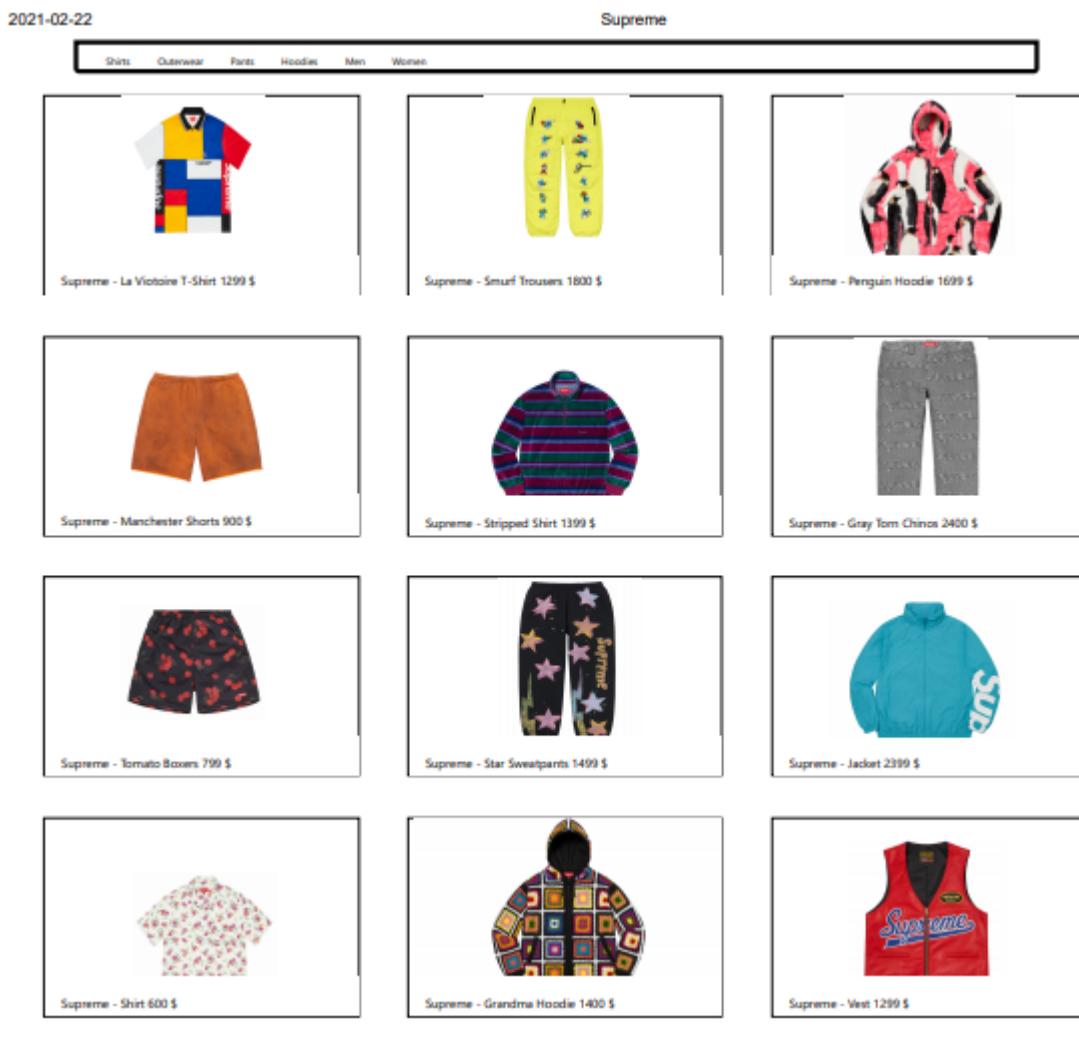
Figur 30: Skärmdump av kundvagnen för webb



Figur 31: Skärmdump av kundvagnen för Iphone 8

Utskrift

Jag skapade ett tredje css dokument som använder ett @media print som ändrar utseende vid utskrift. Här har jag tagit bort alla menyer och stora knappar och headers/footers som inte är nödvändiga vid en utskrift.



Figur 32: Utskrift for Clothes

2021-02-22

Supreme

About us

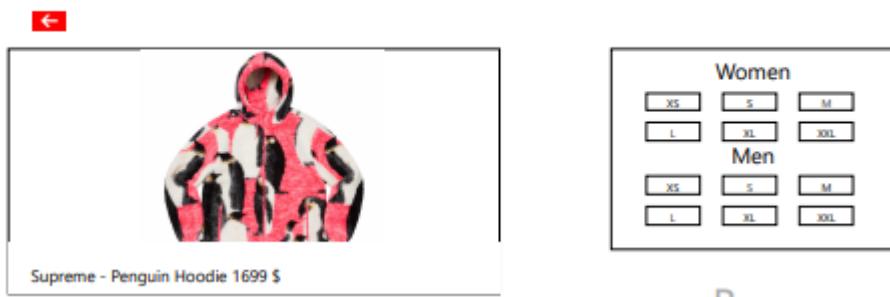


In April 1994, Supreme opened its doors on Lafayette Street in downtown Manhattan and became the home of New York City skate culture. At its core was a group of neighborhood kids, New York skaters, and local artists who became the store's staff, crew and customers. Supreme grew to embody downtown culture, and play an integral part in its constant regeneration. Skaters, punks, hip-hop heads - the young counter culture at large - all gravitated toward Supreme.

Figur 33: Utskrift för About us

2021-02-22

Supreme



Figur 34: Utskrift för Pingvin hoodien

Validering

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for http://130.239.115.172/user076/Supreme/Index.html

Checker Input

Show source outline image report [Options...](#)

Check by address [address](#) [file](#) [url](#)

<http://130.239.115.172/user076/Supreme/Index.html>

[Check](#)

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

1. **Error** Attribute `href` not allowed on element `button` at this point.
From line 18, column 9; to line 18, column 82
`<button href="javascript:void(0);"`

Attributes for element `button`:

[Global attributes](#)
[disabled](#) — Whether the form control is disabled
[form](#) — Associates the element with a `form` element
[formaction](#) — URL to use for `form submission`
[formenctype](#) — Entry list encoding type to use for `form submission`
[formmethod](#) — Variant to use for `form submission`
[formnovalidate](#) — Bypass form control validation for `form submission`
[formtarget](#) — [Browsing context](#) for `form submission`
[name](#) — Name of the element to use for `form submission` and in the `form.elements` API
[type](#) — Type of button
[value](#) — Value to be used for `form submission`

2. **Error** An `img` element must have an `alt` attribute, except under certain conditions.
For details, consult [guidance on providing text alternatives for images](#).

From line 19, column 13; to line 19, column 103
`<img src = "Images/Hamburger menu.PNG" style = "width: 7vw; height: 7vw;"`

Figur 35: Validering av index-sidan

25vw; height: 24.7vw;overflow: hidden;">✉

43. **Error** An `img` element must have an `alt` attribute, except under certain conditions.
[For details, consult guidance on providing text alternatives for images.](#)
From line 213, column 13; to line 213, column 129
`✉`

44. **Error** An `img` element must have an `alt` attribute, except under certain conditions.
[For details, consult guidance on providing text alternatives for images.](#)
From line 221, column 13; to line 221, column 131
`✉`

45. **Error** An `img` element must have an `alt` attribute, except under certain conditions.
[For details, consult guidance on providing text alternatives for images.](#)
From line 232, column 15; to line 232, column 100
`✉`

46. **Error** An `img` element must have an `alt` attribute, except under certain conditions.
[For details, consult guidance on providing text alternatives for images.](#)
From line 237, column 15; to line 237, column 98
`✉`

47. **Error** An `img` element must have an `alt` attribute, except under certain conditions.
[For details, consult guidance on providing text alternatives for images.](#)
From line 242, column 19; to line 242, column 103
`✉`

Figur 36: Slutet av valideringen för index-sidan

Valideringen ger många fel var av nästan alla är för att jag inte har något "alt" attribute på mina bilder, jag har kollat igenom felen och det verkar inte vara några allvarliga. Resterande sidor ser ungefär likadan ut och jag känner inte att det är någon ide att lista upp dem alla.

Problem & Reflektioner

Denna uppgift har varit bland de lärorikaste jag har gjort. Jag har lärt mig extremt mycket om css,html och webbutveckling generellt. Mycket på egen hand via internet då jag helt hade missat att det fanns teori om html/css på kurssidan.

Det tråkiga däremot var att det skedde under en sådan stressig period då jag har en massa annat runt omkring samt att uppgiften endast hade en vecka på sig. Jag kände att det var för lite tid för att få till en komplett implementation. Jag har jobba ihjäl mig på denna uppgift och jag var inte ens nära på att få in den i tid. Om det var mitt eget fel vet jag inte, men jag har snittat 8+ timmar per dag på denna uppgift den senaste veckan och knappt hunnit färdigt.

Det har även uppstått en massa problem för mig under denna uppgift som jag har suttit med många timmar vilket har segat ner mig också. Till exempel hade jag problemet att det alltid går att scrolla lite åt höger och jag vet fortfarande inte varför det är så. Jag tror jag har provat allt och felsökt varenda element jag har. Till och med startat om projektet och börjat från scratch, jag har kommenterat bort varenda rad kod och noggrant undersökt varendaste element men förstår fortfarande inte vad som orsakar det.

Ett annat problem jag hade var som nämnt tidigare att omvandla alla värden från px till vw. Det tog också lång tid att förstå sig på hur @media fungerar och hur man gör för att ändra ett element vid en viss skärmstorlek.

Jag satt även en hel dag på att försöka förstå varför jag inte kunde ändra storleken på en bild med hjälp av css. Tills att jag lärde mig att man måste använda !important efter width deklareringen.

Allt detta har gjort att uppgiften tog lång tid att göra klart och jag tror att jag har upplevt varenda problem man kan tänka sig när det kommer till html. Men det är just därför jag har lärt mig så otroligt mycket också, i retrospekt är jag inte nöjd med strukturen av koden och hade jag gjort om uppgiften nu idag hade det sett helt annorlunda ut. Men nu vet jag hur man gör och jag känner mig mera självskräckande med webbutveckling på det stora hela!