

## Innehåll

Hur projektet genomfördes .....	1
Inledning.....	1
Förberedelse .....	2
Design och programmering .....	2
Webbplatsens utveckling.....	2
Programmering.....	2
HTML koden .....	2
Innehåll i head .....	2
Innehåll i header (även navigation) .....	3
Innehåll i container.....	3
Innehåll i content.....	3
CSS koden .....	4
JavaScript koden .....	4
Testning .....	5
Rapportskrivning .....	5
Skiss över webbplatsen .....	6
Startsida, rapport (index.html) .....	6
QR-kod till webbsidan index.html.....	7
Källförteckning.....	7

## Hur projektet genomfördes

### Inledning

Jag tänkte göra en webbplats för föreningen Eskilstuna Vapenhistoriska Förening, de har en befintlig webbplats enligt [www.evhf.se](http://www.evhf.se). De nya webbsidorna ska vara responsiva och mer användaranpassade. Den befintliga webbplatsen är vad jag förstått gjord med wordpress och då skapas programkoden med hjälp av någon form av verktyg. Detta genererar då i sin tur den kod som används, dvs HTML, CSS, JavaScript osv.

Det sättet att göra sidorna får inte användas, utan här ska man i stället skapa programkoden i HTML, CSS osv själv. Så den nya koden blir begränsad då det gäller att få till bildhanteringen, som jag skulle vilja kunna göra med bildspel (under länkarna för 'Bilder' och 'Skarpskyttekåren' på [www.evhf.se](http://www.evhf.se)). I övrigt ska webbplatsen vara enligt den befintliga.

**Bengt Franklin, Kursen WD454F – Responsiv webbdesign**

Att skapa innehållet på engelska har inte gjorts, detta skulle vara önskvärt att få till senare för föreningen. Det skulle vara ganska lätt att göra det på ett antal nya webbsidor. Det enklaste är kanske att göra dubletter av de svenskspråkiga sidorna. Det svåraste blir nog att få till någon form av språkvalssymbol och att underhålla ytterligare webbsidor.

### Förberedelse

Information med text, bilder kopierades och sparades från den befintliga webbplatsen.

### Design och programmering

Började fundera på hur jag skulle göra designen av sidorna. Tanken är att göra sidorna liknande de befintliga, men förhoppningsvis mer användarvänliga och responsiva. De ska klara av minst tre olika typer av enheter, som tex mobiltelefon, surfplatta och datorskärm. Bestämde mig för att göra utvecklingen med 'Mobile first' och sedan göra anpassningar av designen till större skärmformat. Har tänkt att den minst skärmen (mobiltelefon) som man ska klara av att vissa åtminstone har 320px i bredd. För att göra webbsidan så att den bättre klarar olika enheter, så används en metatagg i <head>-taggen: <meta name="viewport" content="width=device-width, initial-scale=1.0"> ([Frain, Ben 2022, s. 12](#)).

Begränsat webbplatsens storlek till maximalt 1300px.

Programmeringen ska göras i HTML, CSS och JavaScript.

## Webbplatsens utveckling

### Programmering

Programmering inledes med att börja med startsidan för EVHF, dvs index.html. Det blev senare två olika startsidor för EVHF:s webbplats, index\_evhf.html samt index\_alt\_evhf.html, se under rubriken [Skiss över webbplatsen](#). Därefter skapade jag script-fil med css, dvs mincss.css. Har lagt en class=container med allt innehåll i body-taggen. Denna class använder CSS grids i css-filen med grid-areor i body-taggen enligt:

header, navigation, evhf, content, mail\_bilder, main\_left, main\_right samt footer.

Här valde jag efter ett tag en gulaktig bakgrundsfärg (gold).

### HTML koden

#### Innehåll i head

Meta-taggen <meta name="viewport" som beskrevs ovan, så här länk till CSS- och JavaScript-filer placeras. Här har jag även lagt länkar till en CSS-fil som återställer stylesheets, reset.css samt efter den min egen kodade CSS-fil mincss.css.

**Bengt Franklin, Kursen WD454F – Responsiv webbdesign***Innehåll i header (även navigation)*

Taggen header, innehåller meny-valen, som är byggd så att om skärmen har mer än 960px bredd, så används class=navigation som visar de olika menyvalen i klartext. Om skärmen har en mindre bredd än 960px, så används en så kallad 'hamburgermeny' för menyvalen. För detta har jag använt en liten del programkod gjord i JavaScript, filen minjava.js, se mer under [JavaScript koden](#).

*Innehåll i container*

Allt innehåll som visas på webbsidans area, dvs klasserna (class): header, content samt footer.

*Innehåll i content*

Innehåller dessa klasser: evhf, main\_bilder, main\_left, main\_right.

*Innehåll i evhf*

Webbsidan rubrik på innehåll

*Innehåll i main\_bilder*

Innehåller bilder på de flesta av webbsidorna. För startsidan (dvs index.html) har jag här gjort ett försök med lite interaktiv kodning i JavaScript, som byter en eller två smala bilder (white\_picture.jpg) då det borde få plats med bilder med bredare innehåll. På det sättet har jag fått till upp till två till bilder, då bildformatet borde tillåta det. För detta används i JavaScript-koden en kontroll av skärmbredden, ungefär en gång i sekunden och eventuellt byter bild/bilder. Kanske inte så användarvänligt, men var kul att prova på. Var inte så lätt att få till på ett bra sätt. Har för detta även lagt bilderna i en flexbox med namnet container\_flexbox. För detta har ganska mycket av programkod gjorts i CSS-filen.

Angående flexbox se tex [Frain, Ben \(2022\). s. 93 och framåt](#).

På webbsidan evhf.html har jag i stället använt flexboxen, container\_flexbox\_evhf med två bilder, om viewport (visningsområdet på skärmen) är mindre än 600px så läggs de två bilderna under varandra annars visas de bredvid varandra.

På webbsidan bilder.html där 12 stycken bilder visas har jag använt en flexbox, container\_flexbox\_bilder som består av grupper med tre stycken bilder, dessa är container\_flexbox\_bilder1, container\_flexbox\_bilder2, container\_flexbox\_bilder3 samt container\_flexbox\_bilder4. De första två av dessa ingår i container\_flexbox\_bilder1-2 och de andra i container\_flexbox\_bilder3-4. Detta för att kunna gruppera bilderna olika, då viewport mindre än visas bilderna under varandra. Då viewport är minst 600px läggs de första 3 bilderna (dvs container\_flexbox\_bilder1) bredvid varandra, nästa bilder under dessa o.s.v (totalt i fyra 'rader'). Då viewport är minst 1200px läggs de första + andra, totalt 6 bilder (dvs container\_flexbox\_bilder1 och ...2), bredvid varandra. De övriga under dessa bilder.

På webbsidan skarpskyttekaren.html där endast 4 stycken bilder visas har jag gjort en liknade lösning, men där finns det bara en flexbox, container\_flexbox\_skarp som består av två under grupper, container\_flexbox\_skarp1 och container\_flexbox\_skarp2.

*Innehåll i main\_left*

Vänstra sidan på webbsidans innehåll, om bredden på skärmen är mindre än 600px, annars kommer denna del före innehållet i main\_right.

För att få till detta på webbsidan har jag i CSS-filen använt layouthantering enligt grid, se tex [Frain, Ben 2022, s. 129](#) och framåt.

**Bengt Franklin, Kursen WD454F – Responsiv webbdesign***Innehåll i main\_right*

Högra sidan på webbsidans innehåll, om bredden på skärmen är mindre än 600px, annars kommer detta innehåll efter main\_left.

*Footer*

Har valt en gulaktig bakgrundsfärg för detta, samma som för menyraden.

Har behållit tre länkar till 'startsidor', nämligen för 'Uppgift RWD2, Rapporter m.m.', 'Alt. 1: Startside EVHF' samt 'Alt. 2: Startside EVHF'. Det blir en del problem och svårigheter att hitta tillbaka till startsidor, då jag valt att ha två stycken sådana för föreningen.

Har även behållit koden där JavaScript skapar svar om vilket skärmformat som används, samt även koden för att göra HTML-validering av sidorna. Dessa har varit väldigt användbara vid utvecklingen av sidorna och kan ganska enkelt tas bort om man ska visa 'vanliga' användare webbsidorna.

## CSS koden

Började även tidigt med en CSS-fil, som skulle användas för alla webbsidorna och det har fungerat ganska bra. Men för en del hantering för några av webbsidorna har jag fått lägga till egen CSS-kod. Detta då alla webbsidorna har exakt samma uppbyggnad.

För att få rensning av CSS, har jag även använt en reset.css fil, se Jennys exempel i kursen under 3. Begrepp och centrala principer och under punkten Skapa en responsiv webbplats:

[https://hkr.instructure.com/courses/6014/pages/skapa-en-responsiv-webbplats?module\\_item\\_id=303465](https://hkr.instructure.com/courses/6014/pages/skapa-en-responsiv-webbplats?module_item_id=303465).

När det gäller textstorlek har jag försökt att använda fluid typography med css-taggen clamp. Hoppas det blivit lagom stor text och användarvänligt att betrakta. Se tex under:

<https://www.smashingmagazine.com/2022/01/modern-fluid-typography-css-clamp/>

Textstorleken varierar då beroende av viewportens bredd. Var lite trassligt att få till vilka värden man borde använda och är inte säker att jag fick till de optimala. Man anger ett minsta värde, ett önskat värde samt ett maximalt värde på typsnittets storlek. Försökte använda denna länk, för att hitta ett bra värde:

<https://modern-fluid-typography.vercel.app/>

## JavaScript koden

För att göra hanteringen av olika bilder i webbsidan

[https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/index\\_evhf.html](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/index_evhf.html)

så har jag använt JavaScript-kod. Den gör så att de fem bilderna ibland bara 'ser ut' som 3 och ibland visas helt. Det finns också en del av dessa bilder som har smalare varianter och vilka som används görs genom att kontroll mot viewport (skärmens visningsområde) görs och beroende på detta används olika

**Bengt Franklin, Kursen WD454F – Responsiv webbdesign**

bilder. Koden i JS som används för detta körs ungefär 1 gång per sekund. Kanske lite onödig och krävande för uppgiften, med det var kul att experimentera lite. Det har även varit ganska tidkrävande att göra.

JS innehåller även lite kod (funktionen nowpressMeny) för att hantera hamburgarmenyn och den har inte varit samma problem att få till. Den används då man klickar på någon av valen i mobilmenyn (mobile-menu) och används för samtliga webbsidor.

### Testning

Testningen har gjorts stegvis under tiden jag utvecklat och försökt att förbättra och rätta sidorna. Prövade med olika färger på bland annat bakgrundsfärger och även med olika ramar runt sidorna. Hoppas jag fått fram färger som inte ger för stora kontraster och blir användarvänliga.

Har testat på bland annat lite olika mobiltelefoner, både med olika operativsystem som Android och IOS, samt i olika webbläsare (IE, Firefox samt Chrome). I de flesta webbläsare tycker jag att det fungerar ganska bra. Framförallt bilder har varit svårast att få kontroll på.

### Rapportskrivning

Efter en del funderande kom jag även att göra länkar till rapporten och två olika webbplatserna som skulle kunna vara EVHF:s webbplats startsidor.

index.html                      Vart sida med länkar till rapporterna (rapporten i pdf-fil och den muntliga) och de två EVHF:s startsidorna.

[https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/index.html](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/index.html)

index\_evhf.html                Startsidan för EVHF, med lite olika inladdning av bilder.

[https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/index\\_evhf.html](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/index_evhf.html)

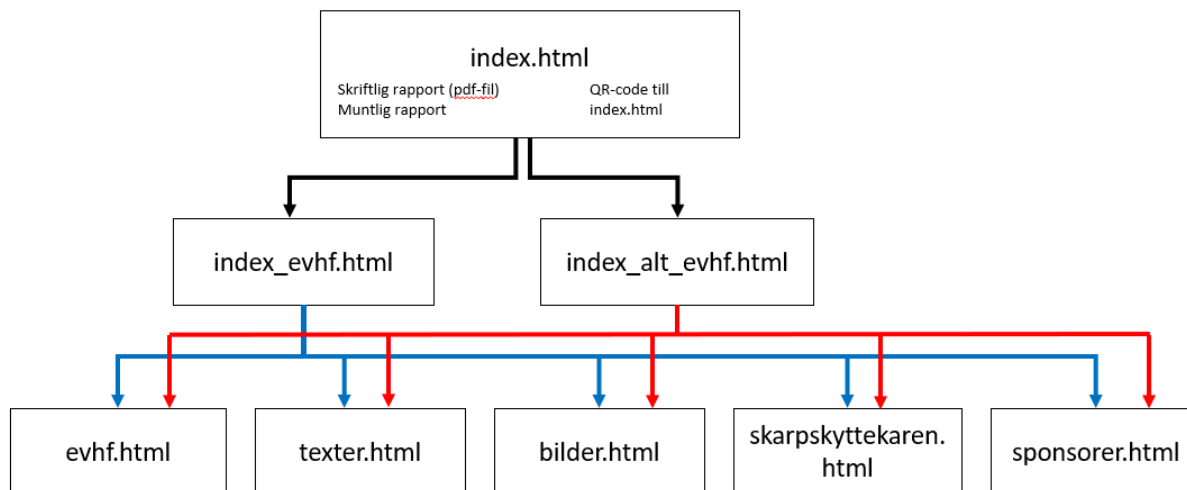
Detta beroende av storlek på skärm och med hjälp av lite JavaScript-kod, med hjälp av exempel på hur man kan koda enligt w3schools.com under länken

[https://www.w3schools.com/js/tryit.asp?filename=tryjs\\_intro\\_lightbulb](https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_lightbulb)

Index\_alt\_evhf.html            Alternativ startside för EVHF, med olika bilder beroende 'bara' på skärmens storlek. Detta med taggen <picture> [\(Jehl, s. 134\)](#). Den gör ungefär på samma sätt som JavaScript-koden, men styrs av Media Queries istället. Här finns det även fem bilder som visas, men det beror på viewportens bredd vilken som läggs ut.

[https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/index\\_alt\\_evhf.html](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/index_alt_evhf.html)

## Skiss över webbplatsen



Har använt samma CSS-fil (mincss.css) för samtliga webbsidor, detta gör att man får samma struktur, färger mm genomgående. Tanken är naturligtvis att man har en av webbsidorna `index_evhf.html` eller `index_alt_evhf.html` som startsida med namnet `index.html`. Från denna startsida är det nu via menyn länkat till de andra undersidorna.

## Startsida, rapport (index.html)

Innehåll i menyraden och länkar till:

- Skriftlig rapport (pdf-fil), muntliga rapporten samt QR-code till webbsidan (index.html)
  - o [https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/index.html](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/index.html)
- Startside EVHF (index\_evhf.html)
  - o [https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/index\\_evhf.html](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/index_evhf.html)
- Alternativ Startside EVHF (index\_alt\_evhf.html)
  - o [https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/index\\_alt\\_evhf.html](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/index_alt_evhf.html)

## QR-kod till webbsidan index.html



QR-koden finns även med på webbplatsen startsida, dvs index.html på URL:

[https://www.webshare.hkr.se/BEFR0010/RWD2\\_EVHF/#](https://www.webshare.hkr.se/BEFR0010/RWD2_EVHF/#)

## Källförteckning

Frain, Ben (2022). *Responsible Web Design with HTML5 and CSS*. Birminghamn – Mumai.

Jehl, Scott (2014). *Responsible responsive design*. New York: A Book Apart.

<https://www.smashingmagazine.com/2022/01/modern-fluid-typography-css-clamp/>

W3schools.com, [https://www.w3schools.com/js/tryit.asp?filename=tryjs\\_intro\\_lightbulb](https://www.w3schools.com/js/tryit.asp?filename=tryjs_intro_lightbulb)