

Documentation – Joakim Borg

Github Link

<https://github.com/JoakimBorg/JSforWebDev-HK>

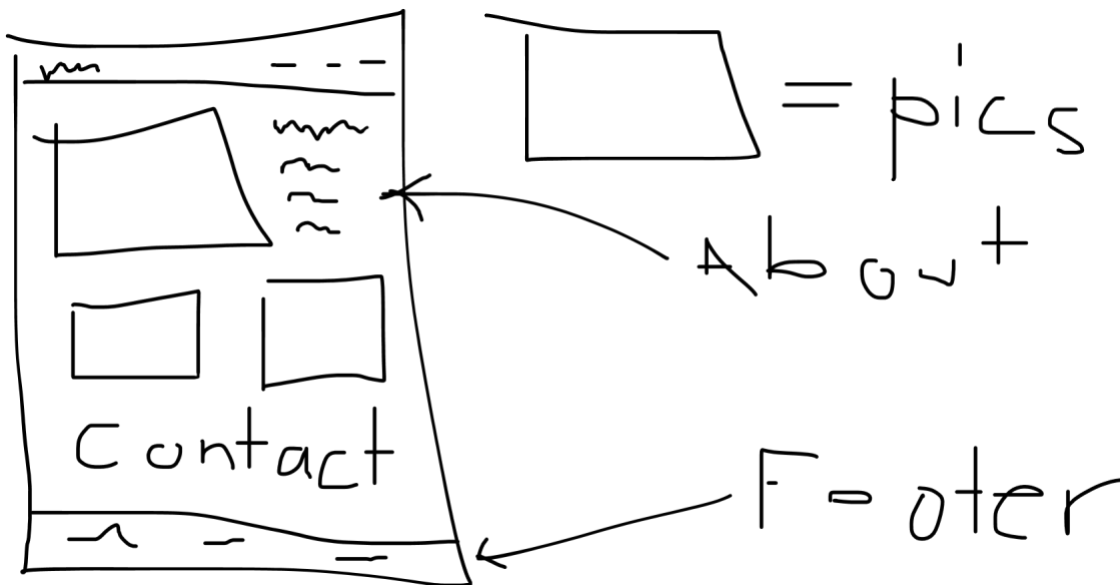
Design Iteration

My initial idea for a design was basing it around the three colors used on the page (purple, yellow and blue). I wanted to use them to separate the sections from each other so it could be easier for the eye to navigate the page.

My hero frame I decided to put on top and put my name as an absolute text over. It's not the prettiest but gets the job done, that was the first change I made from the design I was told to implement.

The second change I made was I put a lot of pictures in two columns formatted for the mobile/desktop layouts. I like pictures and decide to make an entire section out of it.

I don't have a screenshot of my first attempt, and since this is not the first time I work with HTML/CSS, I had a fairly clear image of how this should be done. This is a drawing of how I thought it should look like initially.



As you can tell, the design changed quite radically during the process.

I think the colors on the page is what really makes it work. It stands out and separates

things from each other and I use them in hover/focus as well to make sure the user can see what's going on.

Technical Challenges

Three problems I encountered:

1. Columns
2. Images
3. Media

Columns

Using flex with columns was the main unlock for the page. This made everything easier, from the small lines to the images.

```
.bilder-column-wrapper {  
  display: flex;  
}  
  
.bilder-img-wrapper {  
  display: flex;  
  border-radius: 12px;  
  overflow: hidden;  
  margin: 12px 12px 0px 12px;  
}
```

I use it almost everywhere and with *display: flex* and sometimes *flex-direction: column*.

Images

The image section had some issues with the scaling but *object-fit: cover* solved quite a lot of issues for me. The images can't have a fixed height/width since they are both portrait and landscape.

```
img {  
  width: 100%;  
  height: auto;  
  object-fit: cover;  
}
```

Media

I decided to go with a *max-width: 600px*. It's not a common width but I felt it to be necessary since I had issues when there were several *@media* sizes. None really made it work. But together with the columns I felt it was about here it should be.

```
@media only screen and (max-width: 600px) {
```

Personal Reflection

The thing that surprised me most was actually that I thought it would be JS, not HTML/CSS. And since I know HTML/CSS quite well I finished it in only a couple of days.

The CSS property giving me the most headache overall was the *width: 100%* / *width: auto*. I still have no idea why they don't really act like they should, especially on divs.

My final design differed quite a lot, I felt a sense of inspiration when making it. I made sure that I followed my inspiration and especially the color lines and images differ quite radically.