

bedre nætter

Bedre Naetter:
the new front-end with
Hyvä Themes



Bedre Naetter is one of the Europe's biggest manufacturers and suppliers of beds, mattresses, sofas, duvets, pillows and bedsheets. Being an e-tailer that operates through bedrenaetter.dk allows them to offer high-quality products at excellent prices as compared to physical shops.

About bedrenaetter.dk

Bedre Naetter has a multi-level complex ecommerce that includes many custom solutions and dedicated widgets. They also launched new marketing campaigns with many external widgets that allow to insert pictures, videos and graphics. Originally, the solution was built in Magento, while the front-end was created in Luma.

After a few years of maintenance, the front-end started to slow down. Key indicators from Google Lighthouse were getting worse. All pages were slow, modules and scripts were loading literally indefinitely. Images weren't optimized for uploads, only the largest size were used. Additionally, new content wasn't replacing the old one but was just added to it. There were also a lot of modules from external suppliers. In the end, everything slowed down significantly.

The necessity to rebuild the ecommerce arose and time mattered.



Solution

First, we considered PWA and other headless technologies. However, the high level of ecommerce complexity ruled this option out. Additionally, it would have been hard to create and thus time consuming and cost-ineffective.

After deliberation, we decided to recommend to our client a completely new approach to front-end: Hyvä Themes.



Hyvä
themes



About Hyvä Themes

Hyva Themes is a new front-end for Magento 2. Technically it is a Magento 2 theme, but it was built with a completely redesigned approach. Hyva Themes is also lighter and simpler as compared to the standard Magento front-end - Luma.

Hyva is a concept, a set of instructions that developers have to follow to get a proper result. It's a base that improves the Magento layout, and the page structure while disposing of unnecessary things. Hyva brings you a tool that does not require a high entry threshold. New people working on a project will be able to relate easily as the number of dependencies in the code is reduced. Thanks to a set of well-chosen components, Hyva allows achieving good performance results and shortening of page loading time at the same time.

Hyva uses Tailwind, one of the best styling frameworks. Compared to the others it is much lighter and offers granular control. Thus, you can control everything, every single letter. Hyva also uses AlpineJS, which is one of the lightest Java Script frameworks for mobile content, especially compared to Vue.js and React.



You can learn more about Hyva here.

Magento Upgrade



Before we started Hyva it was necessary to update the Magento version from 2.2.7 to 2.4.2 (now it is 2.4.3). That decreased server response time by $\frac{1}{3}$. A lot of code had to be adapted due to a very large amount of custom solutions - not only additions from the Magento Marketplace but also a lot of our own code providing numerous functionalities that are not available elsewhere.

Implementation of Hyva

In the beginning, after it went into production, there were no gigantic surprises. Cosmetic stuff like fixing font-type flaws. At the same time, the client changed his mind about different aspects (e.g. layout) in the course of Hyva's implementation. Despite that, it went smoothly: we adapted not only the code but also the CMS (be careful that Hyva uses classes, rarely identifiers, classes repeat and you need to spend time identifying elements) and product pages, which had their own HTML. We had to rewrite it and adapt to Alpine.JS and Tailwind CSS.

We managed to do a lot but it turned out that there were external pages that we did not have access to e.g. a linked blog. It did not affect Magento but Google indexed it differently thus we managed to distinguish indexing of homepage and subdomains.

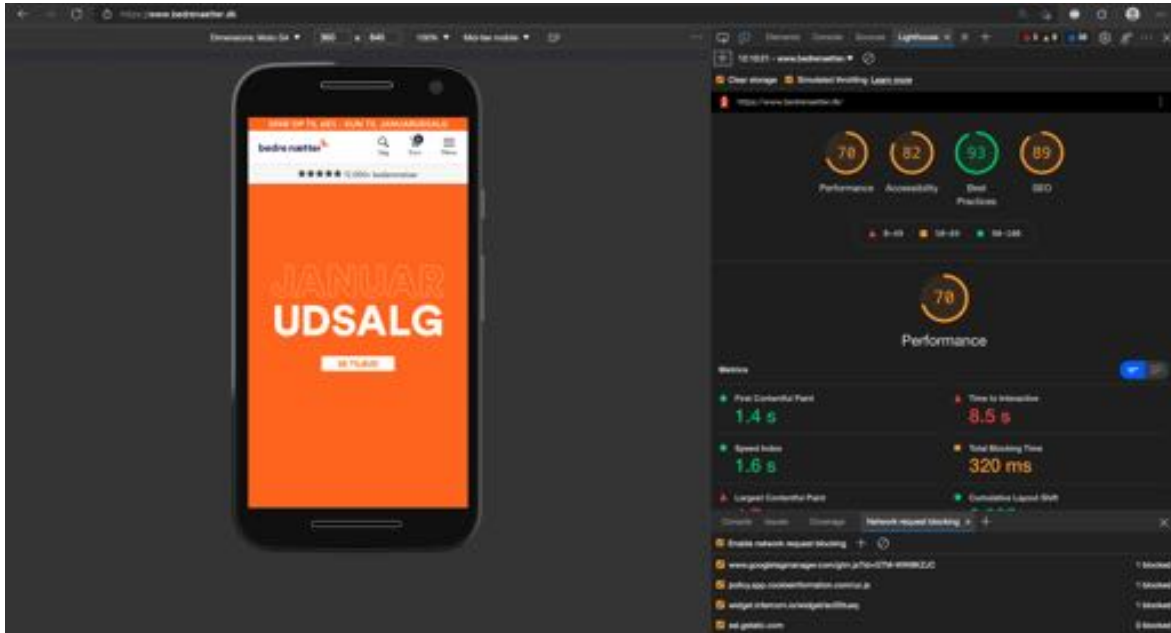


Results

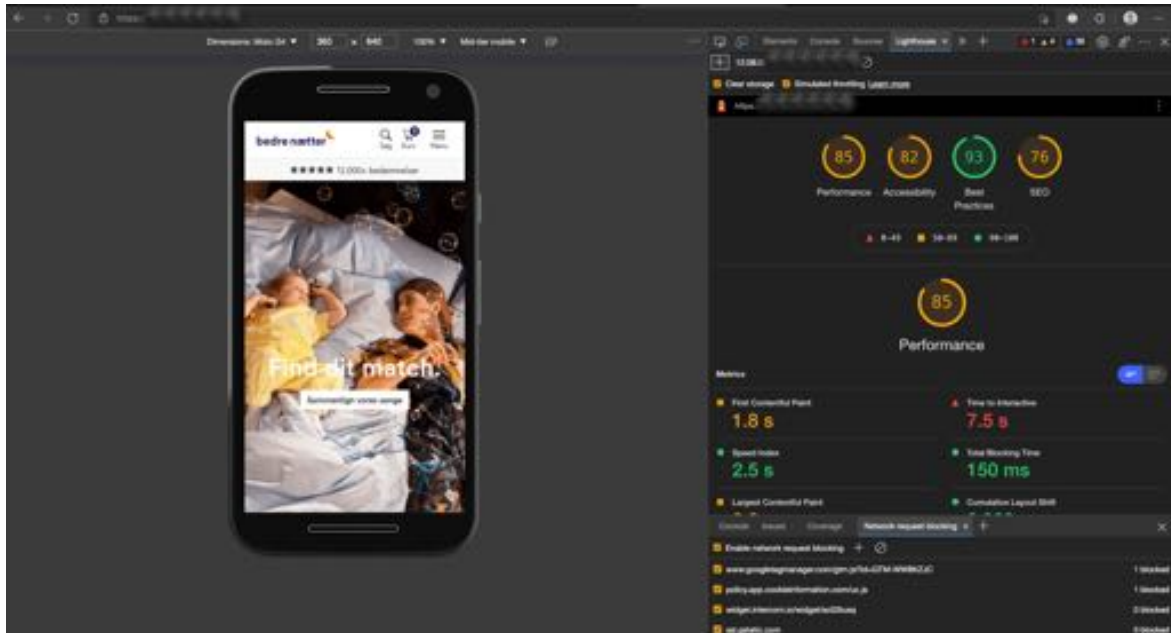
Before Hyva, Bedre Naetter had very poor results in Google Lighthouse: 14-20. Now it's 60.

However, it requires advance planning and proper handling almost constantly due to lot of client activities. The loading time also depends on the speed of the user internet connection.

Production:



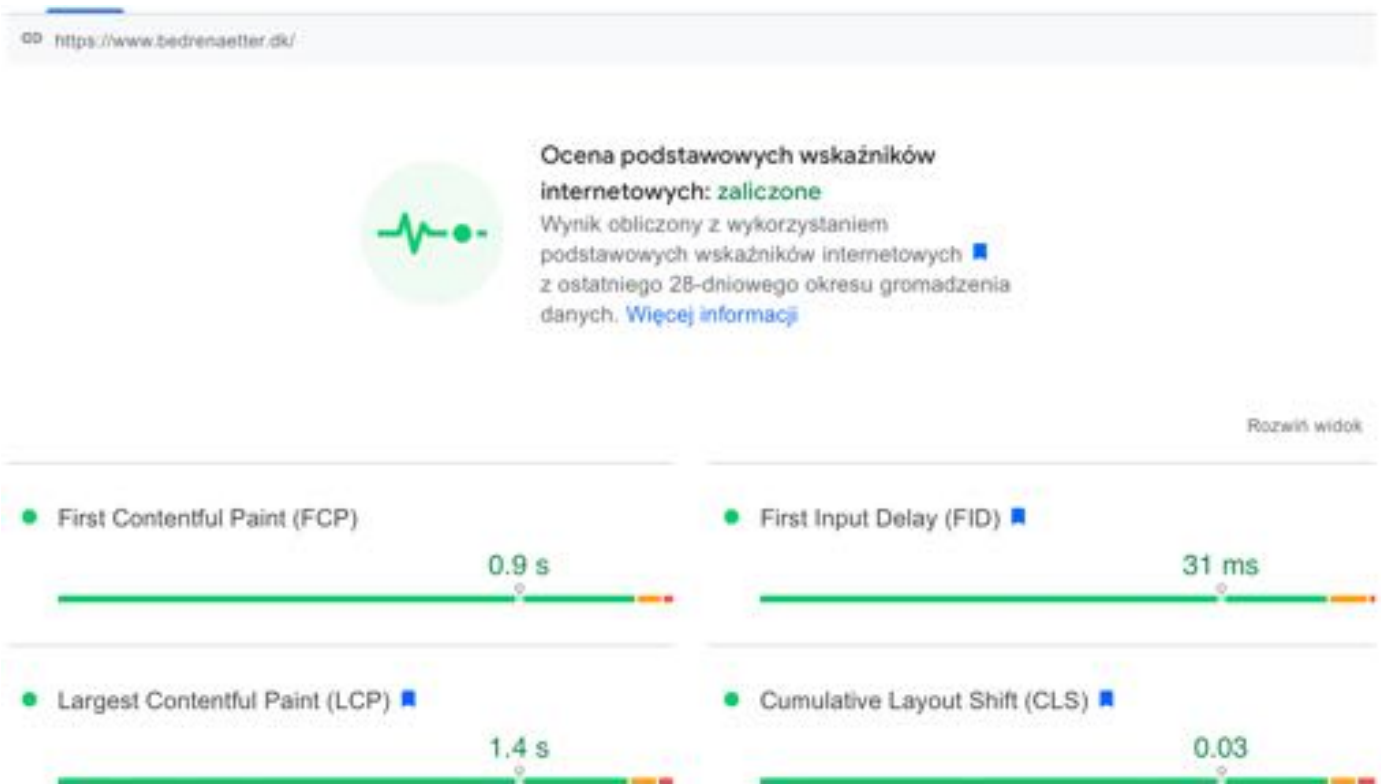
Test environment:



Before Hyva, the product card page made 300-500 requests to the server. This is a typical product page with a large number of native and external widgets scripts for tracking users (Google Tag Manager), Instagram photos, a help chat (the help chat stand alone had 111 server requests) etc.

With Hyva there are between 100 and max. 130 server requests.

Page speed is also much better now:



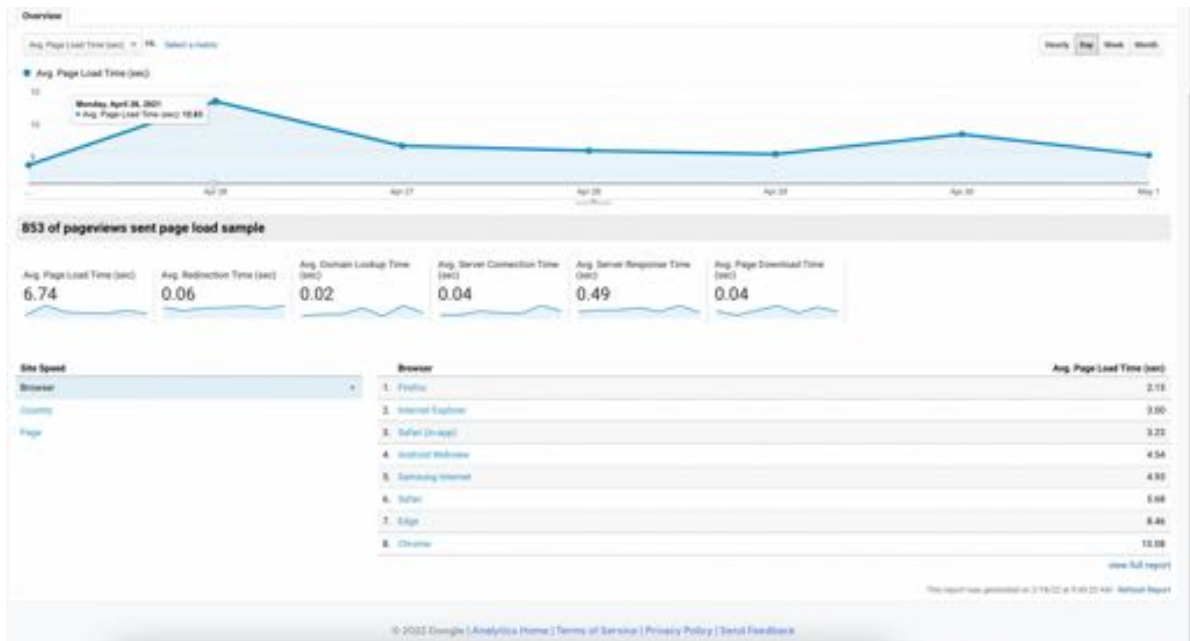
What is important, **all indicators are in green**:

- First Contentful Paint (FCP) = **0.9 sec**
- Largest Contentful Paint (LCP) = **1,4 sec**
- First Input Delay (FID) = **31 ms**
- Cumulative Layout Shift (CLS) = **0,03**

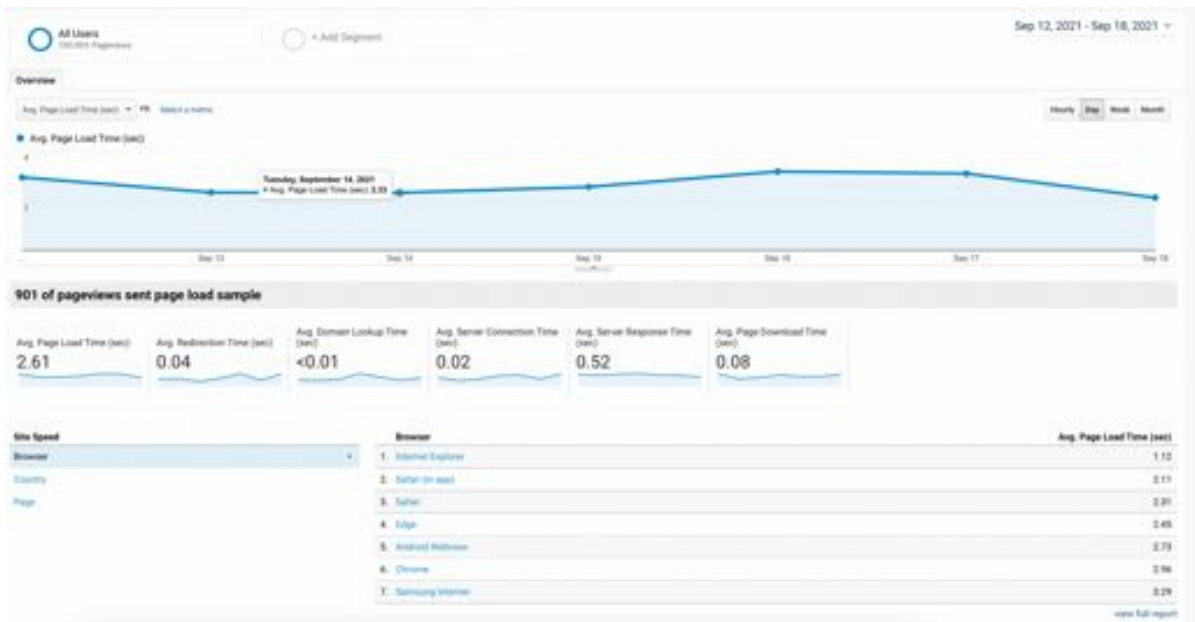
Bedre Naetter ecommerce has a big potential to achieve very good results. This require further work, discipline in ecommerce management and an increase of content technical quality. Currently, we are working on optimization and on completing various new tasks, e.g. user accounts, checkout, etc.

Comparison in Google Analytics

Before Hyva: Avg. Page Load Time: **6.74 sec**



After Hyva: Avg. Page Load Time: **2. sec**



Coding Challenges

Magento coding model

The challenge for the Magento developers was to stop thinking about the Magento coding model and to switch to a different solution. Some things were completely different than in Magento, e.g. the JS handling.

Separate stage

Another different challenge was that the stage was completely separate. Once Magento had been upgraded to the new version we needed to upgrade Hyva. The client was changing pictures, content, products, marketing campaigns continuously. Thus, we had to adjust Hyva to all the above-mentioned changes on a regular basis.

Customization creates issues

Hyva does not support everything yet. The problem appears even with the simplest customization of products. Hyva had a problem with grinding the price change up - the price did not increase. We needed to do it manually and it was like swimming through porridge. The code in Bedre Naetter is very complicated and although the client doesn't use all of its all functionalities now, we needed to make it compatible in case they want to use it in the future.





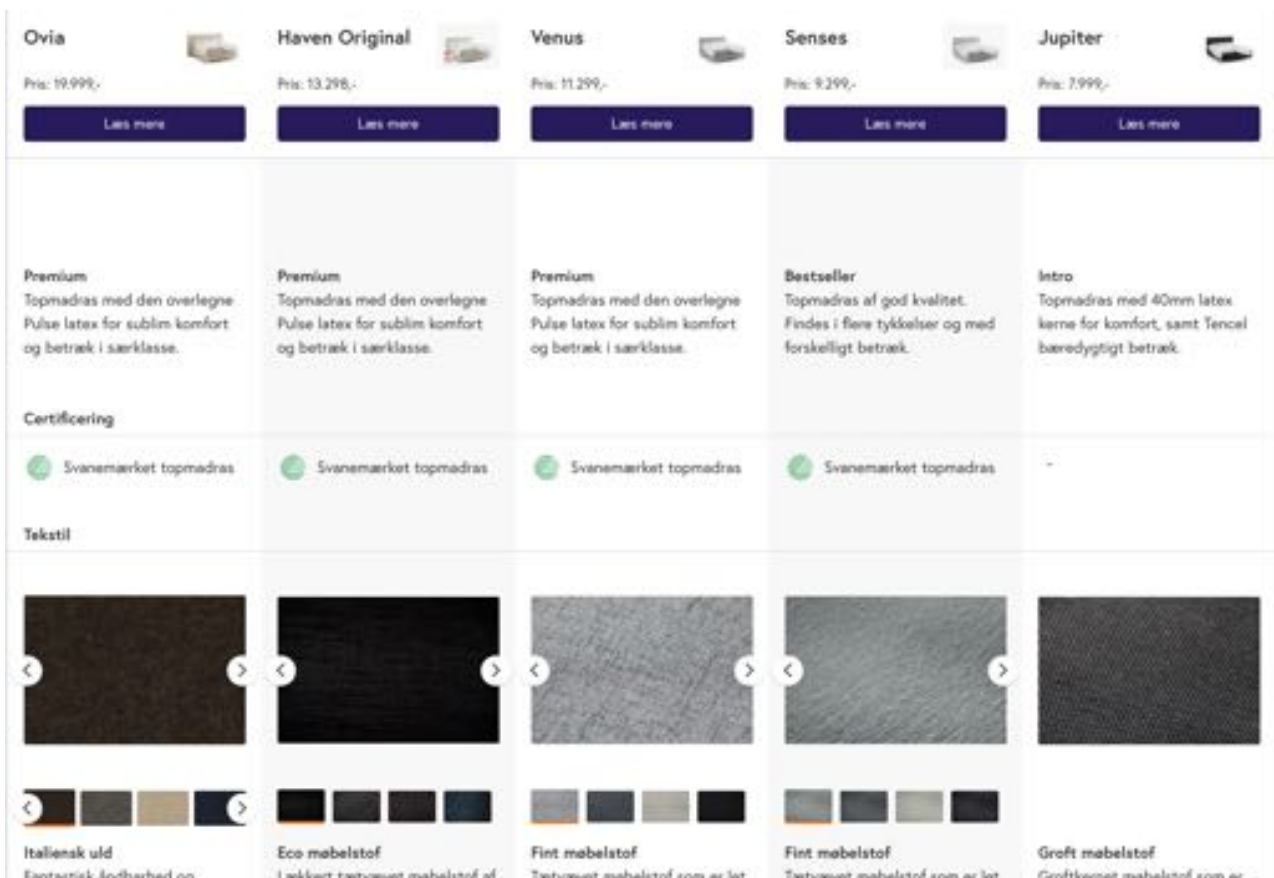
Technical Challenges

Custom widgets

While it was important to us to keep the original look and functionalities, that was not always possible.

For example, it was impossible to adapt the product comparison page to the Hyva. The top-notch is synchronized with the bottom irrespectively if you scroll up or down or move it sideways. We had to adapt this page and all of its numerous widgets to all types of devices: smartphones, desktops, tablets, tv, etc. which was very time-consuming. It was converted from Magento's JavaScript dependencies to plain Java Script + Tailwind CSS.

It was done properly but it will be reworked again because the client decided to change the functionality on mobile devices.



Google Tag Manager

Google script is suboptimal and slows down the whole ecommerce. Bedre Naetter has a Google Tag Manager inserted in the header of the page and this script consists of a dozen smaller scripts tracking users. It isn't asynchronous - when this script starts loading, it blocks the page until it's loading is finished.

It is one of the top 3 scripts that will lower scores after implementing Hyva. Two others to improve in the future are Intercom (help chat) and information about the cookies consent box (CCB).



Cookies Consent Box

CCB is managed by an external company that is responsible for cookies. This script is very slow. When we implemented Hyva, we adapted it to the Hyva appearance. However, after that, the client changed its shape to make it look different. The box covers the entire page and loading late. It is also big and being the largest contentful paint (LCP - one of the Google Lighthouse key indicators), means that it slows the whole ecommerce down so it loses performance points (approximately 40 points).

Having seen that, we needed to recode it one more time.



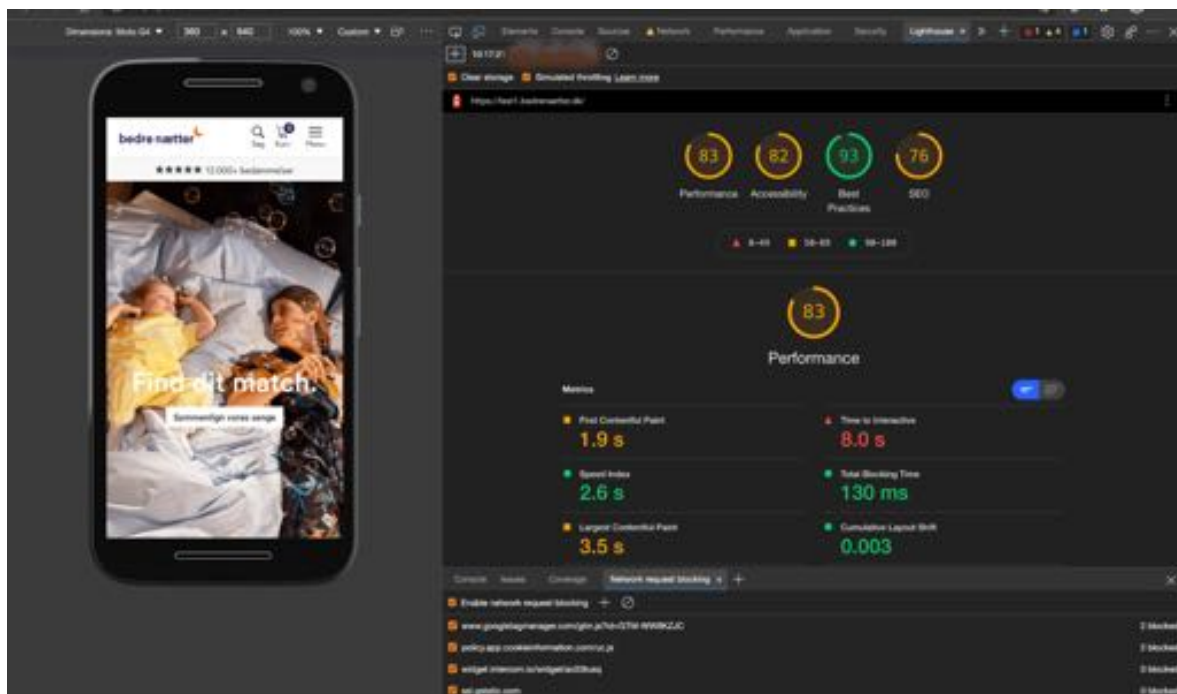
On test environments bedrenaetter.dk needs **7 sec** to load. On production it is 13 sec: **GTM takes away 2 sec, CCB 1,5 sec and Intercom 2 sec.**

Menu

We also had to implement a new main menu. The old one was an issue while Magento was generating a menu for desktop and mobile at the same time. There was also a lot of useless HTML.

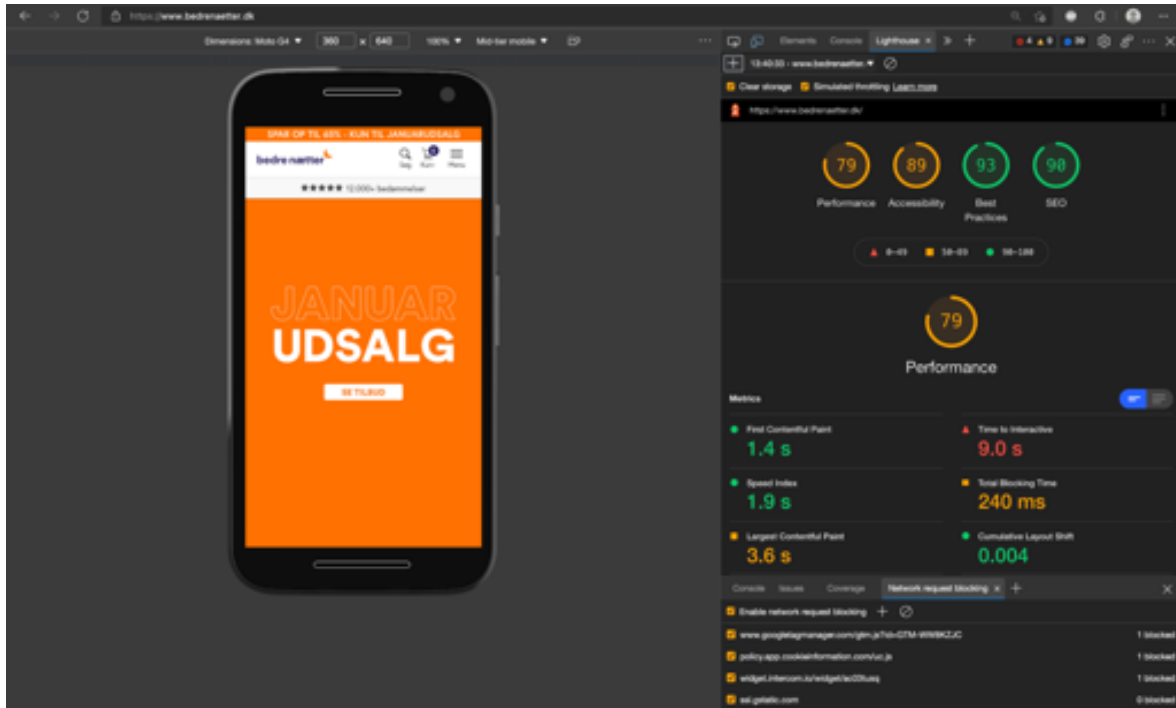
We changed the code so that menu is generated only once. As a result this improved performance on the test server to 83 points.

Test environment:



After implementation.

Production:

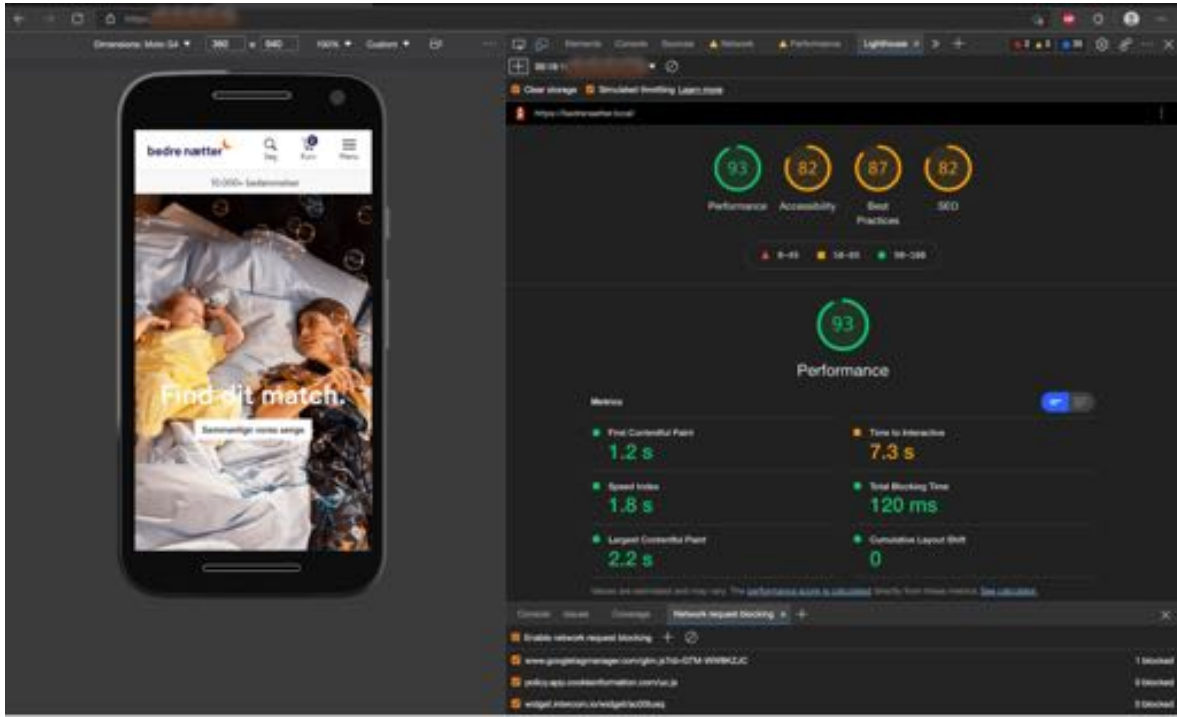


Video on the main page

Bedre Naetter has a video on the home page. It is uploaded only after a user scrolls to where this video is. The biggest element on the page loads first and the rest loads later. However, sometimes there might be display issues if the client makes a mistake and upload a new video in the wrong place.

The above loading sequence allows reaching **93 performance points** at local env.

Test environment:



Waiting for the future

Checkout

At the time, there was no checkout module in Hyva. We have postponed the existing checkout rework until Hyva will launch this module officially. We preferred to wait until it will be checked and stable rather than implementing an unstable solution.

We have added our own custom functionality of comments to order at every stage. That was not available with Hyva. It's easier to choose Hyva module and add the functionality to write everything ourselves. The latter could be incompatible with Hyva checkout in the future. Thus checkout stayed on the old theme for the time being.

Custom Form

Company custom forms, a module that allows you to create your own forms on the website, e.g. a contact form or send the application form - is a very extensive one. We will probably also wait for Hyva to rewrite it like with the checkout. We would spend a lot of time without the guarantee that it will be compatible with all Hyva functionalities. It must be compliant with Hyva standards.



Page Builder

Starting from version 2.4.2. we didn't have a page builder to build content for the page. In version 2.4.3. Magento introduced this functionality to the administration panel but it was not compatible with Hyva (from Alpine.JS) because the tags it uses, e.g. @click, caused some of the scripts contained in these tags to be removed by Magento.

We have turned this page builder off on Bedre Naetter with version 2.4.3 because it doesn't work properly with Hyva version 1.1.8. Now Hyva version 1.1.9 is available but we will wait for the next one because only the latest version is compatible with the page builder and Alpine.JS. We will introduce the higher version of Alpine.JS and possibly it will require further adjustments on the website.

From a management point of view, the page builder is a nice and needed thing. From our perspective it's another thing that needs to be taken care of – it must be at a high level so that it does not cause frustration and problems.



Work Model

We worked on the **Kanban board**. We have divided the project into specific areas, then into tasks and subtasks which were assigned as tickets to people within the team. We have concentrated on the main page which had the biggest prio (MVP). Then we started working on category pages, product pages and widgets.

Fortunately, we had a very **flexible and Agile response from client**.

Thus, preparing templates for these pages (structure), custom functions and a lot of module calls was easier. After the structure was ready, we started working with content and functionalities.

The flagship example was Amasty Layer Navigation, a large module responsible for displaying filters and subcategories in a given category. We reworked it with custom functions displaying parallel categories.

Another challenge was to adapt widgets to a specific client's ecommerce solutions. Glide.js was especially helpful here - for more information [follow the link](#).

 **Summary**

Hyva is a young product with big potential. We had to spend a few extra hours to add some functionalities that Hyva didn't offer yet. What is good that in Hyva the entry threshold is low, much better than in Magento, and it's very simple to work with it.

A lot depends on the design: in a very simple design everything is ok. The problem arises when there is something custom, artistic or irregular in a design. Configurability is difficult. You need to have extensive knowledge to know how to prepare everything correctly. But compared to Magento, where the styles are set, in Hyva it is simpler and faster – a style compilation in Magento takes ages, in Hyva it takes a moment.

It's important that you need to develop a method of working with JS in the form of repeatable blocks because Magento caches these blocks. If the block appears more than once, there may be conflicts. Hyva does not give guidelines on how to do it from the beginning to the end or on how to produce as little code as possible but to work under foreseen circumstances.

Also, you have to come up with the question of deployment, this is not an ordinary Magento flow. Choose the right moment. Hyva does not provide guidelines for the time and place of deployment so use logic.

You can see the main differences in the performance. Hyva moved forward, we adjusted the modules to make them more suited to Hyva. For example the video slider was adjusted to Hyva.

There is also real support from Hyva itself - the community lives in Slack, maybe someone asked for something once, you can search it. There is an answer to simple things. Willem and the rest of their team are very active and helpful. They reply within hours.

A great tool for performance

Hyva itself gives a lot of possibilities. A lot depends on the client's workflow. What content they put in, what CMS blocks with custom styles. usually, we have to produce the styles ourselves.

Hyva is faster, however, better points in Google Lighthouse require a lot of knowledge about performance. Hyva it's a good tool but you have to use it properly.



Hyva makes it possible to achieve great performance, with our knowledge we can bring it to your project!

We will find the best solution for you! Get in touch!

