A screenshot of a cell phone

Description automatically generated

Project Exam 1

Anders Styve

Word count

Summary: 199 | Main text: x xxx

Table of Contents

[Summary 3](#__RefHeading___Toc312_3630458238)

[Intro 4](#__RefHeading___Toc314_3630458238)

[Design 4](#__RefHeading___Toc318_3630458238)

[Technical 7](#__RefHeading___Toc326_3630458238)

[WCAG Guidelines, content management and SEO 9](#__RefHeading___Toc322_3630458238)

[References 11](#__RefHeading___Toc324_3630458238)

## Summary

This report covers the written part of Project Exam 1 at Noroff’s Frontend-development class.

The purpose of the exam was to create a blog site from scratch, putting together all the skills aquired over the first year of studies. The blog is created with HTML, CSS and JavaScript, but with the main content being hosted on a headless WordPress site. Using the WordPress REST API, the content of the blog (posts, comments, images) gets fetched and displayed live on the finished website.

The main content of the report is split in three sections, each covering a relevant topic. The topics are "Design", "Technical" and "WCAG Guidelines, content management and SEO". Each topic is divided into three parts where the first part covers my thoughts on what went well, the second part explains what parts where more difficult or challenging and, finally, the third and last part is made up of thoughts and ideas of what I would do differently on a project like this a next time.

At the very end of this report is a full reference list of all sources used in this exam assignment. This includes both images and written references.

Link to final website: https://objective-brown-e68010.netlify.app/

## Intro

This report will be divided into three sections, each covering a specific topic related to the exam assignment. The three different sections are: “Design”, “Technical”, and “WCAG guidelines, content management and SEO”. In each of those three sections, I will elaborate and discuss what I thought went well on the project, what did not go so well, and what I would have done differently next time.

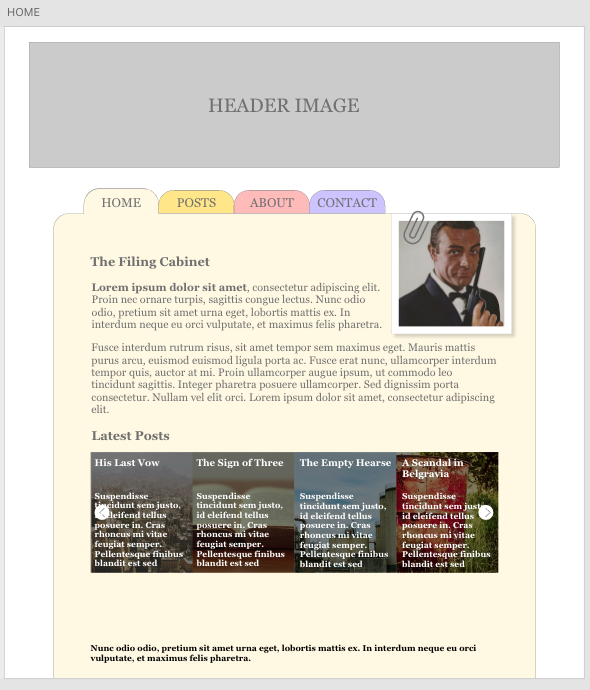
## Design

In this section of the report I will explain my thoughts and ideas behind the design of the website. This includes both the initial layout created before the actual programming of the site, and the design choices that were taken throughout the process of developing the site.

***What went well on the project***

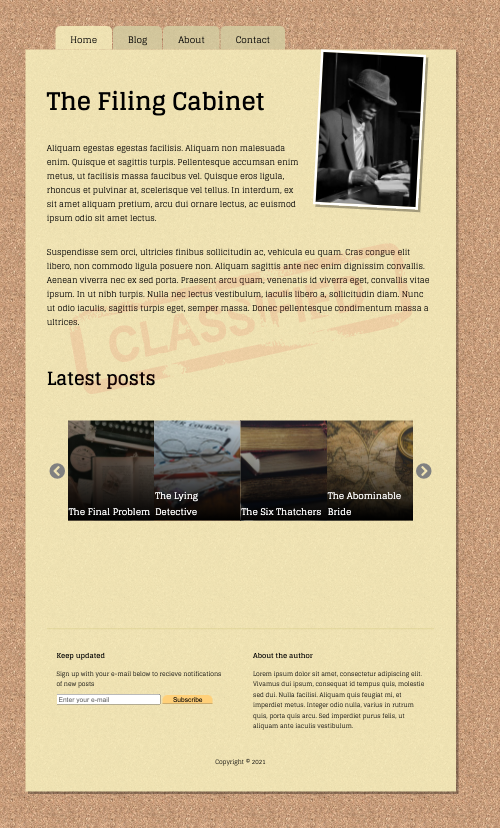
The inspiration for this blog website initially came from classic older movies, shows or video-games involving investigation or detective units. I wanted to create a blog that looked and felt like an old filing cabinet consisting of several blog posts, each covering an unsolved mystery. This also served as an inspiration for the name of the blog.

Based on those initial ideas and concepts, the first design layout for the blog ended up looking like the following image:



As shown in the initial design, I wanted the blog to consist of a home page containing a brief introduction, an image of the author, and a section showing the latest blog posts in an image carousel. The navigation bar is designed to look like the kind of navigation one could see from older archives.

Some changes were made when transitioning the design from that initial layout to a functional website, but the initial concept stayed the same. The picture below shows the final version of the blogs home/index page.



The blog's background was changed from plain white to a pattern that looks like a cork-board. This was done both because it was more aesthetically pleasing and because it gave the blog section a nice contrast against the background. The colour of the blog was also changed to a more yellow tone trying to mimic the natural colour of old documents. At last, the header image was removed, as it served no real purpose and only increased the height of the blog, making it longer to scroll through.

**What was difficult/didn’t go well on the project**

To further develop the idea of this archive or filing cabinet concept, I initially wanted the blogs posts to look like they had been written on an old typewriter. After finding and trying to implement several alternatives, this feature had to be discarded as the fonts were too hard to read, and not user friendly enough.

Finding a background pattern that had all the functionalities that was wanted also took a bit more time and effort than anticipated. This was mostly due to the fact that the background had to be the right color, as well as having the proper license for usage, and being repeatable multiple times without showing. Having had difficulties regarding the file size of images used as website backgrounds before, this was also something that had to be considered when finding the optimal background pattern.

**What would you do differently the next time**

Even though I did research and collected inspiration from various other blogs and websites, a thing I would do differently another time is spending even more time to design different initial drafts or layouts. This is particularly relevant on an assignment with a relatively large time frame like this.

When first designing the layout for this blog, I was so focused and excited about my initial design concepts that I may not have spent sufficient time exploring other ideas. Even though one could have initial favourite designs or layouts, using time trying to create something entirely different could be time well spent as one could stumble upon ideas or solutions that could be implemented into the initial design.

## Technical

This section of the report includes all the technical parts behind the creation of the blog. This includes the programming of the website, and the transition from an initial design to a fully functional blog with all necessary features.

**What went well on the project**

This assignment’s scope was to create a website that uses WordPress as a database to store all its content. By using API calls to this WordPress database, the content would then be displayed on the website. Not only should the website be able to make a call to the WordPress database to fetch and display information, but the website should also be able to send or post information directly to the database to be stored there.

After setting up the WordPress site and making it headless and connected with my website, I created several blog posts directly in WordPress. These posts consisted of a title, a body of text, and an image. An API call to the WordPress database was made to display these posts on the blog, which returned a set number of posts and displayed them accordingly on the site.

The first issue was to initially only show ten posts and then having a button underneath that, when clicked, showed more posts underneath the initial ten. This was solved by utilizing the WordPress REST API’s built-in “per\_page” argument, where the default value is ten. So when the initial API call is made, by default, only ten results will be shown.

By binding a function that makes a call with the “per\_page = 2” argument to the “Load More” button, the returned posts will be the posts numbered 11 to 20. These posts will then be displayed underneath the initial ten. By solving it this way, one also secures that the newest blog posts will be displayed at the top of the site, as a new post will automatically be number one of the ten first shown posts.

To implement search functionality to the site, I used another built-in argument in the API called “search”. This was implemented into the site and connected to the displayed search input field. A new API call with the argument is sent when a search query is entered, and results are shown directly on the website accordingly.

One of the main challenges involved in this assignment was creating a comment section on each blog post that allowed users to view, write and submit comments directly connected to a specific blog post. This was solved was by creating a function that gathers the values of the input elements that the users fill in. After these values are validated, a fetch request with a “POST” (rather than “GET”) method is sent to the API. If this is successful, a body containing the string values of the input is posted to the WordPress database on the correct blog post ID.

Another functionality added to the site is the opportunity for users to enter their e-mail in a newsletter subscription form, or filling out a contact form to get in touch. These functionalities were both added by creating a fictional blog post for each of them to store the inputted data. These fictional blog posts would then use WordPress’s built-in categories filter to differ from the actual blog posts. In practice, this means that the user’s inputted e-mail or contact values is posted as a comment on a blog post named “Newsletter” or “Contact Form”. As these blog posts are not categorized as regular blog posts per se, they will be hidden and only visible to an admin user after logging in.

**What was difficult/didn’t go well on the project.**

Working with WordPress as a database did have some limitations and possibly demanded a little more effort to achieve the preferred outcomes. This was especially challenge that occurred when trying to combine the correct endpoints and arguments when fetching data from the API.

A concrete example was figuring out how to fetch an image related to a specific blog post. The image URL was not included as an endpoint in the standard URL when connecting to the API. A string containing “?\_embed” had to be included in the API URL to retrieve said images. This was just one example of a situation where retrieving the wanted data from WordPress was not as straightforward as preferred and demanded some time and effort in configuring the right paths for accessing the API.

**What would you do differently next time.**

One of the most significant drawbacks I have had through the programming of this website is that the way it functions now, it sends several API calls back and forth to the WordPress database. Every time a user clicks, searches or comments on a blog post, a new API call gets sent. Of course, most of the calls is needed to have the functionality that is wanted, but I do also believe that there could be ways to increase the efficiency and reduce the loading of the website by utilizing, e.g. local arrays, to a greater degree.

The feature of having local arrays to store, sort and filter information to reduce the amounts of API calls back and forth is definitely something I consider worth exploring the next time working on a project like this.

## WCAG Guidelines, content management and SEO

This section explains how I worked according to the WCAG guidelines during this assignment. In addition, my thoughts and decisions regarding content management and search engine optimization (SEO) will be discussed.

**What went well on the project**

One of the first objectives I had when starting to develop this site was managing the site’s contrast and colours. I initially wanted to create a site where the page elements would be as good as they needed to be to avoid any chance of confusion or miss-clicking from the users. To further optimize the readability of the site, I decided to use only two different fonts and limit the number of different font sizes.

A challenging feature to implement regarding contrast and readability was the title text overlay on the images in the “Latest posts” image carousel on the front page. Having text on top of images can often be an issue when it comes to the user being able to effortlessly reading the text. I solved this issue by having a gradient overlay on top of the images going from black with low opacity at the bottom to an almost transparent top. This made the bottom half of the images darker and increased the contrast to the white title text, enhancing the readability.

Other necessary implementations that were made was fetching and creating image alt texts directly from the WordPress API so that the alt texts always would be up to date and in line with the content of the images. With extensive help from the services of Hotjar, collecting and record user input and data, several optimizing changes were made throughout the whole process of site creation. The user feedback from Hotjar that directly led to changes were everything from the blog posts lacking “back” buttons, clickable objects not having a pointer cursor to the footer not being locked in place, and not being able to close image modals on mobile devices. The gathering and use of user feedback definitely played a major role in minimizing the website’s unwanted features, as well as being inspiration to several new ones.

**What was difficult/didn’t go well on the project**

One specific issue that kept coming back was how to properly size the images that were supposed to be rendered on the site. Downloading and displaying the images on the site is one of the most time-consuming processes when rendering a website, and high is highly influential when it comes to both the actual and the perceived speed of the website.

This was a challenge because the assignment specifically stated that the blog post image should be clickable and, when clicked, opened into a modal giving the user a larger resolution view of that image. This meant that the original image resolution had to be bigger than the resolution displayed on the site in case any user wanted to click the image to view it in a larger resolution.

**What would you do differently next time**

I have learned throughout the process of creating this website that it is never too early to involve external users and invite them to have a look and an opinion regarding your site. I felt that it is easy to get stuck in one’s own ideas of how a site should look and “feel” and forget about small details that others may discover within seconds of viewing your site.

One particular example was the lack of a “back” button to navigate backwards from a specific blog post to the overview of the blog posts. This function did not cross my mind at all during the site’s creation but was immediately requested by two separate external users during their initial view of the site. This shows how easy it is to get stuck and “not see the forest for all the trees” when designing or programming sites intended for users to interact with.

Another feature that I most certainly would like to spend more time exploring next time is everything related to the total file size and rendering speed of the entire website. A good example is looking into the next-gen formats for saving and displaying images or pictures on a website. With the end-users demanding faster and more streamlined websites than ever before, I think looking into this would be an excellent long-term investment in front-end development.

## References

**Text References**

* Contrado Digital. «How to Post Comments to the WordPress REST API Version 2», 2016. Internet: <https://www.contradodigital.com/2016/04/06/post-comments-wordpress-rest-api-version-2/> [Accessed 27 August 2021]
* Google. (2021). Lighthouse [Browser Extension]. Internet: https://developers.google.com/web/tools/lighthouse
* Hotjar. (2021). Internet: <https://www.hotjar.com/> [Accessed 30 August 2021]
* MDN Web Docs, Mozilla. «Using Fetch», 2021. Internet: <https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch> [Accessed 31 August 2021]
* Meta Box. «The Fastest Way to Load Google Fonts In WordPress (Part 2)», 2020. Internet: <https://metabox.io/load-google-fonts-faster-wordpress/>[Accessed 22 September 2021]
* REST API Handbook, WordPress. «Comments», 2021. Internet: <https://developer.wordpress.org/rest-api/reference/comments/>Accessed [27 August 2021]
* Section.io. «How to Make an Image Carousel Using Basic HTML, CSS and JavaScript», 2021. Internet: <https://www.section.io/engineering-education/how-to-make-an-image-carousel-for-your-website/> [Accessed 25 August 2021]
* Tetchi Blog. «How to post comments using the WordPress REST API», 2019. Internet: <https://www.tetchi.ca/how-to-post-comments-using-the-wordpress-rest-api> [Accessed 27 August 2021]
* Wikimedia Foundation. «List of Sherlock Episodes», 2021. Internet: <https://en.wikipedia.org/wiki/List_of_Sherlock_episodes> [Accessed 19 August 2021]

**Photo References**

* Bulbfish. Pexels, 2021. Internet: <https://www.pexels.com/photo/vintage-music-business-money-9264012/>[Accessed 29 August 2021]
* Classified stamp. FreeSVG, 2021. Internet: <https://freesvg.org/classified-stamp>[Accessed 20 September 2021]
* Cork board pattern. Tiling Textures, 2021. Internet: <https://www.tilingtextures.com/cork-board/>[Accessed 15 September 2021]
* cottonbro. Pexels, 2021. Internet: <https://www.pexels.com/nb-no/bilde/mystisk-forstorrelsesglass-forskning-retro-7319068/>[Accessed 29 August 2021]
* cottonbro. Pexels, 2021. Internet: <https://www.pexels.com/photo/geometric-building-on-street-with-tramway-5235178/>[Accessed 06 September 2021]
* cottonbro. Pexels, 2021. Internet: <https://www.pexels.com/photo/geometric-building-on-street-with-tramway-5235178/>[Accessed 06 September 2021]
* cottonbro. Pexels, 2021. Internet: <https://www.pexels.com/nb-no/bilde/teknologi-musikk-datamaskin-lyd-8382621/> [Accessed 06 September 2021]
* Gonullu, M. Pexels, 2021. Internet: <https://www.pexels.com/nb-no/bilde/ute-gammel-park-anlegg-6907994/> [Accessed 29 August 2021]
* Hazelwood, S. Pexels, 2021. Internet: <https://www.pexels.com/photo/stacked-books-1333742/>[Accessed 29 August 2021]
* igovar. Pexels, 2021. Internet: <https://www.pexels.com/nb-no/bilde/statue-historie-innendors-militaeruniform-9306496/>[Accessed 29 August 2021]
* Ilina, Y. Pexels, 2021. Internet: <https://www.pexels.com/photo/white-book-page-on-black-and-gray-typewriter-9107393/> [Accessed 29 August 2021]
* Ilina, Y. Pexels, 2021. Internet: <https://www.pexels.com/photo/white-printer-paper-on-brown-wooden-table-9107394/>[Accessed 29 August 2021]
* Koppens, Y. Pexels, 2021. Internet: <https://www.pexels.com/nb-no/bilde/vintage-retro-retning-guide-697662/> [Accessed 29 August 2021]
* Koppens, Y. Pexels, 2021. Internet: <https://www.pexels.com/photo/letters-and-an-eyeglass-on-table-1809342/>[Accessed 29 August 2021]
* Kozlov, A. Pexels, 2021. Internet: <https://www.pexels.com/photo/geometric-building-on-street-with-tramway-5235178/>[Accessed 29 August 2021]
* Roseclay, D. Pexels, 2021. Internet: <https://www.pexels.com/photo/close-up-photo-of-black-typewriter-977930/>[Accessed 29 August 2021]
* Smith, J-M. Pexels, 2021. Internet: <https://www.pexels.com/photo/brown-paper-envelope-on-table-211290/>[Accessed 29 August 2021]
* Smykova, D. Pexels, 2021. Internet: <https://www.pexels.com/nb-no/bilde/bord-reise-vindu-bok-8369435/> [Accessed 29 August 2021]