

TALLINN UNIVERSITY OF TECHNOLOGY
Faculty of Information Technology
Department of Software Science

IT40LT
Viktoria Majerovits 134677

**CREATING WORDPRESS TEMPLATE
WITH UNIQUE DESIGN USING MODERN
UI/UX PRINCIPLES FOR THE COMPANY
NOVOTRADE INVEST AS**

Bachelor's thesis

Supervisor: Kristina Murtazin
Master's degree

Tallinn 2017

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond
Tarkvarateaduse instituut

IT40LT

Viktoria Majerovits 134677

**UNIKAALSE DISAINIGA WORDPRESSI
MALLI LOOMINE, KASUTADES
KAASAEGSEID UI/UX
DISAINIMISPRINTSIPE ETTEVÕTTE
NOVOTRADE INVEST AS JAOKS**

Bakalaureusetöö

Juhendaja: Kristina Murtazin
Magistrikraad

Tallinn 2017

Author's declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Author: Viktoria Majerovits

22.05.2017

Abstract

The aim of this thesis is to document and to demonstrate in practice how to create a good looking web page template for WordPress CMS. The work will be based on a real life scenario: the template will be developed for Estonian petroleum company Novatrade Invest AS. The work could be considered as a further manual for creating similar templates for similar companies. An important part of the work is documenting the process of planning the UX, creating and prototyping the UI, gathering of functional and non-functional requirements and implementing of the design. After reading this work reader should understand basic steps for each analysis and implementation phase, basic principles of developing the template and the problems he or she may encounter during that process.

This thesis is written in English and is 59 pages long, including 7 chapters, 29 figures and 0 tables.

Annotatsioon

Käesoleolva töö eesmärk on dokumenteerida ja näidata, kuidas luuakse praktikas WordPressi malli kvaliteetse UX ja UI-ga. Töö hakkab põhinema reaalelu stsenaariumil: malli luuakse Eesti nafta firma Novotrade Invest AS jaoks. Seda tööd saab käsitleda juhendina, mis teeb selgeks, kuidas luuakse sarnaseid malle sarnaste veebilehede jaoks. Töö põhiidee on dokumenteerida, kuidas saab planeerida UX'i, prototüüpida kasutajaliidest, koguda ja realiseerida funktsionaalseid ja mittefunktsionaalseid nõudeid ja kasutada neid teadmisi malli arendamisel. Pärast töö lugemist tekib arusaamine, milliseid samme tuleb teostada iga analüüsi ja arendamise faasis ning kuidas saab malli arendada ja millised probleemid tekivad selle protsessi jooksul.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 59 leheküljel, 7 peatükki, 29 joonist, 0 tabelit.

List of abbreviations and terms

CMS	<i>Content management system</i> , the web-based application for managing of a website
HTML	<i>Hypertext Markup Language</i> , the standard markup language for defining the structure of the website
CSS	<i>Cascading Style Sheets</i> , the language for describing the style of an HTML document
PHP	<i>Server scripting web programming language</i>
Flexbox	<i>CSS3 web layout model</i> , provides predictable behaviour to elements on a page with different screen sizes [33]
UI	<i>User interface</i>
UX	<i>User Experience</i>
OOCSS	<i>Object-oriented CSS</i> , the methodology for writing CSS
SMACSS	<i>Scalable and Modular Architecture for CSS</i> , the methodology for writing CSS
BEM	<i>Block Element Modifier</i> , the methodology for writing CSS
DRY	<i>Do Not Repeat Yourself</i> , the writing style to express messages clearly and simply

Table of contents

Author’s declaration of originality	3
Abstract.....	4
Annotatsioon.....	5
List of abbreviations and terms	6
Table of contents	7
List of figures	9
List of code snippets	10
1 Introduction	11
1.1 Theoretical Background and Problem	11
1.2 Problem Statement.....	12
1.3 Methodology.....	12
1.4 Thesis Structure	13
2 Website Requirements.....	15
3 Analysis of the most popular Content Management Systems	17
4 UX/UI Principles for building a Website	21
4.1 Do not make a user think.....	21
4.2 Create effective Web pages design.....	21
5 Making of website design.....	25
5.1 Wireframe	25
5.2 Reasons to create wireframe using HTML and CSS.....	25
5.3 Creating Homepage sketch.....	25
5.4 Choosing the right CSS Methodology.....	29
5.4.1 OOCSS	29
5.4.2 SMACSS	30
5.4.3 BEM	31
Final Choice.....	33
5.5 Creating Homepage wireframe.....	33

5.6 Visual design	35
5.6.1 Color Palette	35
5.6.2 Background Image.....	35
5.6.3 Parallax Background Scrolling.....	35
5.6.4 Animation	36
5.6.5 Shades and gradients	36
5.6.6 Navigation	37
5.6.7 Icons	37
5.6.8 Fonts	37
5.6.9 Final Homepage design	37
5.7 Responsive design	39
6 Developing the WordPress template	41
6.1 WordPress installation on a local computer	41
6.2 Organizing WordPress entry files.....	42
6.3 Adding a theme preview image on WordPress	43
6.4 Organizing and plugging in existing style sheets	44
6.5 Creating the Header Section	44
6.6 Logo.....	45
6.7 Creating a custom Navigation Menu	46
6.7.1 Styling of the Navigation Menu	48
6.8 Main Settings.....	49
6.9 Making background image customizable	50
6.10 Creating a Custom Static Front Page Template.....	52
7 Summary.....	53
References	57

List of figures

Figure 1. Arngren Homepage.....	23
Figure 2. Google Homepage	24
Figure 3. Homepage Sketch 1	26
Figure 4. Homepage Sketch 2	26
Figure 5. Homepage Sketch 3	27
Figure 6. Homepage Sketch 4	27
Figure 7. Homepage Sketch 5	28
Figure 8. Homepage Sketch 6	28
Figure 9. Homepage wireframe created with HTML and CSS	34
Figure 10. Color Palette for Website.....	35
Figure 11. Animation for products	36
Figure 12. Button with gray shadow	36
Figure 13. Current page link in Navigation Menu	37
Figure 14. Contact information with icons.....	37
Figure 15. Homepage Visual Design	38
Figure 16. Homepage version on a small screen (iPad Pro screen)	40
Figure 17. Preview of the new theme.....	43
Figure 18. Site Title.....	44
Figure 19. Site Title printed in the browser.....	45
Figure 20. Logo is not displayed.....	45
Figure 21. Pages for the Navigation Menu.....	46
Figure 22. New option - Menus	47
Figure 23. Custom Menu.....	47
Figure 24. Menu Settings	48
Figure 25. Final result for the Header and Footer sections	49
Figure 26. Site Title and Tagline (Administration Panel)	49
Figure 27. Site Title and Tagline.....	50
Figure 28. New Subsection – Header.....	51
Figure 29. New background picture	52

List of code snippets

Code snippet 1. New data with database name	42
Code snippet 2. Information about the theme	42
Code snippet 3. Structured approach – header, footer and sidebar areas	43
Code snippet 4. Code to include the CSS file	44
Code snippet 5. Function bloginfo() to display the Site Title	45
Code snippet 6. Wrong code to display the logo.....	45
Code snippet 7. Function get_bloggerinfo() to display the Logo.....	45
Code snippet 8. Function add_theme_support() to display Menus	46
Code snippet 9. Function register_nav_menu() to create custom menus.....	47
Code snippet 10. Function wp_nav_menu() to display navigation menus.....	48

1 Introduction

1.1 Theoretical Background and Problem

Novotrade Invest AS is one of the largest petrochemical plants in Estonia. At the moment the Novotrade Invest AS has a website, which was built some years ago with the use of Joomla platform. Right now the company website uses ordinary and widely popular Joomla template. Because of that site looks outdated and abandoned. The company is not satisfied with the following situation and plans to reconstruct their website.

The company Novotrade Invest AS is fast growing business, therefore it is important for the company to have a website that will look professional. The presentable website is useful for attracting additional clients and increasing the sales.

Nowadays, it became very popular to take a ready-made template for the site and use it. It is the simplest and cheapest method of creating your own site, but this method carries with it disadvantages too. All existing free templates are already used across various web pages. It can be seen that they all really look alike and the company will not stand out among them. Of course, it is possible to fit the website into the necessary requirements and modify the template beyond recognition. But this method will take a lot of time and effort, so creating the new template may be cheaper, easier and faster in terms of development and price.

Another problem with the existing “free” themes is that despite a huge number of free settings most useful settings are usually paid, so “free” templates may not be totally free. The configuration of such templates makes the client’s and developers live way harder because of a huge number of settings. Since the client asked to create the content management as easy as possible, then such approach would not be suitable. The previous version of the website was created using Joomla CMS and according to the client words, it is very difficult to manage the content of the website.

It was decided that the new website will have a unique design and will use more simple, more documented and more popular CMS platform. The current thesis will document how to perform developing of the template and tell which steps are required to fulfill the above-mentioned goal.

1.2 Problem Statement

The objective of this work is to create a new website with a unique design for the company Novotrade Invest AS. The thesis will mainly concentrate on creating template, which will differentiate the company from its competitors.

The design should respond to the theme and identity of the company. It is very important to consider the company's target audience (people after the age of 40 years).

Usability is one of the crucial variable, which is responsible for the ease of use of the website. It should be simple and very clear, and it does not require any deep thought. The website should be easy to use and navigate. That is why another important part of the thesis will be an analysis and planning of usability and design.

1.3 Methodology

In order to create a professional looking website, it is very important to follow certain rules that will allow eventually achieve the expected result.

For creating a website we consider to pass through the following steps:

- Information gathering from the client. Creation of website requirements for the interface.
- Analysis of the most popular content management systems to choose the most suitable CMS for the website.
- Consideration of UX/UI design principles for creating a more accessible web page.
- Design implementation. This process consists of four stages:
 - Sketching of the website.

- Prototyping the wireframe.
- Developing the design.
- Adapting the existing design for various mobile devices.
- Choosing the right methodology for writing CSS. Making a choice between SMACSS, OOCSS and BEM.
- Development consists of following stages:
 - Creating HTML layout.
 - Writing cascading style sheets for existing HTML. In order to create a mobile friendly version, the author heavily uses CSS3 flexbox properties and BEM methodology.
 - Creating PHP templates from the HTML file and dividing the page into the sections: header.php, footer.php, sidebar.php. Validation, improving and refactoring the code.
 - Adding functionality to manage site content through WordPress.

1.4 Thesis Structure

This work is divided into six parts.

Chapter 1 describes the theoretical background, problems and the objective of this work. It also illustrates the methodology, which will be used for creating a website.

Chapter 2 describes the main website requirements.

Chapter 3 describes the choice of the CMS. Are compared the most popular content management systems.

Chapter 4 describes which user experience and user interface principles should be considered in order to build a good looking and accessible website. This chapter provides more details about how to create an effective page design and which mandatory elements should present on the website structure.

Chapter 5 describes the process of building the website structure and design. It also addresses on a decision which CSS methodology to choose. This chapter consists of four steps: Homepage sketch, Homepage wireframe, Homepage Visual design, Responsive design.

Chapter 6 describes basics of creating the template for WordPress from scratch. The development of the functionality will be described in detail.

2 Website Requirements

The author gained insights and recommendations from the conversation with the customer. During the conversation were identified the goals of creating a new site for the company. The customer had provided a general idea of the company. From the author's side were clarified details about design and functionality. The target audience of the site was described, which helped to choose the appropriate direction of the design. On the basis of all that, the author has compiled a list of requirements for the website.

Functional requirements:

Functional requirements describe system operations and specify what the system should do.

List of functional requirements:

2.1.1. To build a new unique custom company-specific theme from scratch and do not use existing CMS templates due to uniqueness and copyright principles.

2.1.2. The website should be based on the CMS platform that will be easy to use.

2.1.3. The website should support 3 languages.

2.1.4. A person on the website should have the opportunity to contact the company through the contact form.

2.1.6. To add an ability to update website by the client (CMS). The client should be able to fill the website with materials, add images and format information on the web page.

2.1.7. Avoid providing extra functionality in managing the website's content through the WordPress CMS.

2.1.8. On the main page of the site should be Live exchange rates calculator and Brent crude oil prices.

Non-functional requirements:

Non-functional requirements describe how the system performs functionality.

List of non-functional requirements:

2.2.1. Design should be simple, unobtrusive, but original. It should take no more, than 10 seconds for a user to find a contact form.

2.2.2. To choose the suitable color palette according to the company's logo. The logo is blue. The palette should be chosen with the use of the color wheel similar to Adobe Kuler.

2.2.3. On the Home page of the site should be an image with a molecular chain.

2.2.4. The user interface (UI) should be convenient and focused on the target audience. The main target audience is people after the age of 40 years. In particular, this means that font size should be larger than 16px.

2.2.5. The website should be mobile friendly. All web pages should be responsive. The information should be displayed correctly on different screens. The mobile version should fully comply with its function.

2.2.6. The website should be easy to use and navigate. This means that global menus should not have any sublevels.

2.2.7. Navigation should be logical. It implies that global menu should contain following sections: Company, Markets, Products, News, and Contacts.

2.2.8. To place emphasis on the contact information (phone, email, address). When a person visits the site he can see straight away that information. He should not search the contact information on the entire site and on its separate pages. The contact form should be present on the front page.

3 Analysis of the most popular Content Management Systems

CMS (Content Management system) is a web-based application for managing of a website.

These statistics show in percentages the usage of content management systems for May 2017 [43]:

1. WordPress 59.0%
2. Joomla 6.9%
3. Drupal 4.7%
4. Magento 2.5%
5. Blogger 2.2%
6. Shopify 1.5%
7. TYPO3 1.5%
8. Bitrix 1.4%
9. PrestaShop 1.3%
10. Other 19%

Let us consider advantages and disadvantages of the three most popular CMS and find the perfect choice for our website.

WordPress advantages:

- WordPress platform is free and open-source.
- Quick and easy installation.
- Customizable – available different themes, plug-ins, customizations [44].
- Community support is superb. Most of the questions you wanted to ask are already answered. Also, WordPress has a free support forum to ask a question and receive answers [44].
- Beginner-friendly.

- Has simpler and more beautiful content managing UI than Drupal and Joomla.

WordPress disadvantages:

- When compared to Drupal, the WordPress is not the perfect platform for advanced and complex content websites. WordPress is more suitable for small websites such as blog or news site [45].

Joomla advantages:

- Joomla is open-source and free.
- Community support. Joomla has a free support portal to ask a question and receive answers [44].
- Not too technical – possible to run a website without technical support [44].
- Joomla is perfect for social networking websites and for e-commerce shops as it is possible to make settings easily and quickly [44].
- Good middle ground between Drupal and Wordpress.

Joomla disadvantages:

- Not possible to host it for free like WordPress [44].
- When compared to WordPress, the Joomla has mainly paid plug-ins.
- Requires more technical understanding compared to WordPress. Yet less than Drupal.
- Joomla's code is more complex than WordPress.
- Joomla does not have such amount of plugins as WordPress.
- Joomla is not as popular as WordPress.
- Client had issues with understanding how to manage content through Joomla.

Drupal advantages:

- Drupal is free.

- Client had issues with understanding how to manage content through Joomla.
- Improved Performance - Drupal pages are faster to load than WordPress or Joomla pages [44].
- Drupal is the more tunable CMS than WordPress or Joomla [44].
- Perfect choice for creating big and complex websites.

Drupal disadvantages:

- Not possible to host it for free like WordPress.
- Technical support option is expensive.
- Hard to maintain the website, requires the technical awareness from the user.
- In overall, Drupal developers cost more and so the final product.

The final choice:

All three content management systems listed above are open-source and free. The choice of the platform depends on the complexity of the website, experience, and budget.

Drupal is the most powerful platform, but it also requires experience and technical skills. From the point of view of the user, Drupal is a complex platform: Drupal leaves more options to configure, which makes from it a perfect choice for non-trivial webpages. However, the main advantage of Drupal is also its main disadvantage: usually, clients encounter problems managing the website content because of an excessive amount of options.

Joomla is perfect for e-commerce or social networking websites. In the author's view, Joomla is simpler than Drupal but harder than WordPress in terms of content management. Joomla disadvantage is that it is less supported by the community: not so many plugins and themes.

The final result is WordPress. It is simple and intuitive. Simplicity is a key both for beginners and professionals. By creating the website on WordPress we do not have to

worry about that the client will have problems with the filling the new website with a new content. It is the best choice as we need to have an easily administered website. It provides wide but simple functionality, it has better community support and more documentation compared to other CMS systems.

4 UX/UI Principles for building a Website

User Experience (UX) Design is the process to determine what the experience will be when a person interacts with the system. It is a process of designing how people will feel about the system using it or how they will use it [15].

User Interface (UI) Design is focused on a good look presentation of a system. UI design makes a product's development, layout and content more attractive for users. It is responsible for the visual assets of a product's interface [18].

The difference between UX and UI design is that UX design is a more technical field and UI design is more graphic design. UX and UI design are connected with each other and they work together [18]. To start the process of UX and UI designing is very important to study about your customers to understand what they need and to take this information into account.

In order to create an attractive, effective and user-friendly website interface the author researched the main UX and UI principles of design and found various rules for enforcing the good UI. The list of such rules is given below.

4.1 Do not make a user think

One of the non-functional requirements is that the website should be easy to use. A user who will use Novotrade Invest AS company's website should be able to use it without spending any effort. So, each page should be self-evident. Self-evident means that user can use it without questions, just by looking at the web page. Every action on the web page should be obvious. All buttons and links should be obviously clickable. Users should not spend their time thinking about what is clickable and what is not. Mindless choices are one of the main things that help to make a website easy to use. [1]

4.2 Create effective Web pages design

One of the main questions is how to design effective Web pages? In order to make an effective web page we need to understand how people really use the Web. [1]

Users do not like to read pages. Usually, they quickly scan the page and click on the first link that catches interest. People prefer to scan pages instead of reading because usually users are in a hurry and they do not have the time to read everything. Users are interested to find something interesting, something that matches their interests. Often a user focuses on words that seem to match something interesting or useful for him. It seemed to suggest that there are large parts of texts on the website or even full web pages, that user not even looks at. [1]

According to the book “*Don’t Make Me Think*” written by Steve Krug there are five important things that we can do to make sure that users see and understand as much of our website as possible. [1]

1. “Create a clear visual hierarchy on each page”

The appearance of pages should have a clear visual hierarchy. Visual hierarchy helps users to organize and prioritize the content of web page, so the process of page scanning becomes faster and easier. The most important information should have large and bold text and it is better to put it on the top of the page. Things that are similar should be grouped visually together in a similar visual style. Sections that are part of the one section should be grouped together, so the user should understand visually which part belongs to which section. [1]

2. “Take advantage of conventions”

It is very important to use conventions on a web page. They are very useful as they help users faster to figure out about Web page without any effort. For instance, we can use an envelope icon near with email address or a phone icon instead of label “Phone number”. A graphical representation is very popular and it is a good visualization tool to present some information. If a text is accompanied by an icon or a beautiful image it will make the text presentation more attractive and structured [14].

3. “Break pages up into clearly defined areas”

We should give the user an ability to choose interesting category for him. The user can decide which category is useful for him and which category he can ignore. Related elements should be grouped together. [1]

4. "Minimize noise"

Visual noise makes difficult to perceive the information on the web page. One of the important tricks in making something pretty is the absolute absence of something pretty. We should try to get rid of visual noise on the Web pages. Plenty of space helps to read and understand the content of pages. To improve readability and website performance we can use large content margin, padding and paragraph spacing. [2] Good, generous whitespace can make interfaces look inviting and simple. So, we should put space between our lines, between elements and groups of elements. [3] We should try to get rid of useless words and phrases. It helps to make the useful content to be more prominent, pages shorter and reduces the noise level of the page.

On a Figure 1 you could observe an example of the website where everything is placed too closely. It is really difficult to concentrate on something or to find needed information:



Figure 1. Arngren Homepage

The best example of usage of whitespace is Google. The user is focused on the main aim of the page – search.

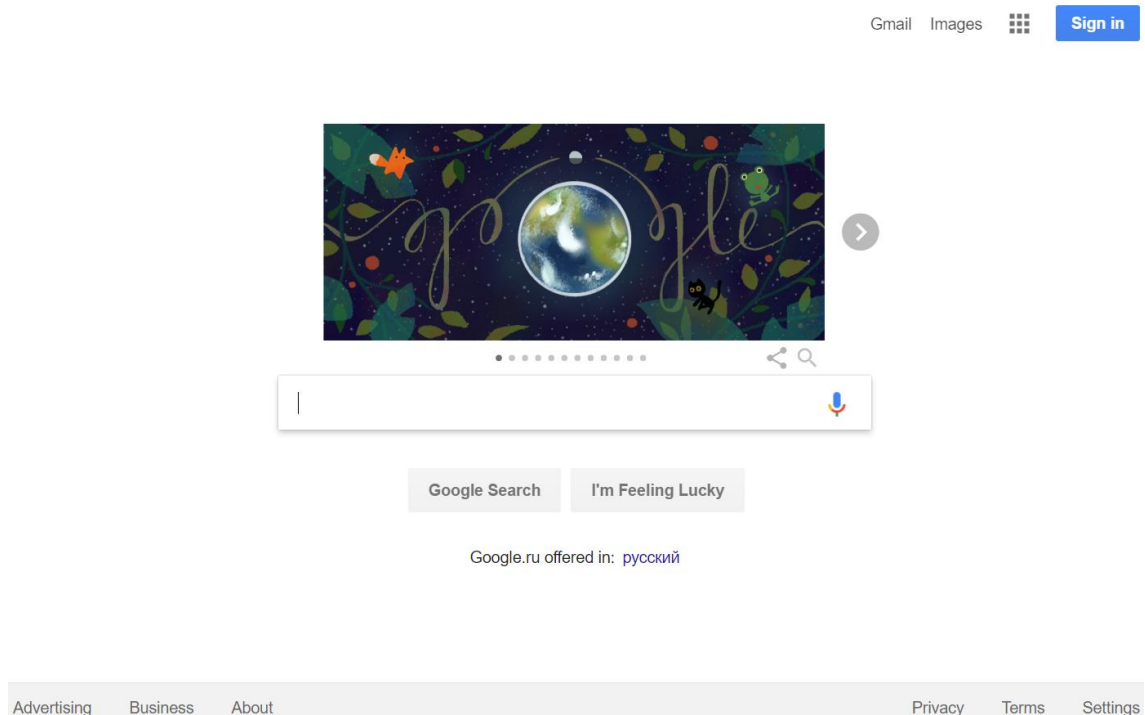


Figure 2. Google Homepage

5. “Make it obvious what’s clickable”

The aim is to get rid of the question marks over user’s head. If we force users to think about something unnecessary we are wasting the limited amount of patience. All buttons should look like buttons and links should look like links. [1]

5 Making of website design

5.1 Wireframe

A wireframe is a basic structure with the proposed content of the website. The wireframe brings together information architecture with visual design [34]. Helps to prioritize website content and determine how and where to allocate items. The wireframe should be simple and colors should be in gray tones. There are many options to create a wireframe, for example using tools such as Photoshop CC, Axure RP, Microsoft Visio or using HTML and CSS. [5]

5.2 Reasons to create wireframe using HTML and CSS

The author decided to create the wireframe using HTML and CSS for following reasons:

- It saves time in the development process because the main HTML and CSS styles will be already written [16].
- The website layout will be already structured and organized.
- Gives an opportunity to create a mobile-friendly template [16].
- Gives more realistic view about the future design [17].
- It gives the correct presentation that we are creating exactly the website. It is not just a PDF or JPEG file [17]. The HTML wireframe is more realistic.
- It allows making changes to the code. Changing HTML code is much more effective than changing the static design tool [17]. So, the website development will be faster and easier.
- It is a foundation for the final product [17].

As we can see there are sufficient amount of reasons why it is better to develop wireframe using HTML and CSS.

5.3 Creating Homepage sketch

Referring to book „A Project Guide to UX Design”: “*Don’t be afraid to get started with pencil and paper—just be sure to bring a lot of erasers.*” Nearly every experienced

professional starts with paper and pencil. A sketch helps to explore many ideas that seem to work perfectly for the project. [5]

Before moving to creating a wireframe the author did 6 sketches. It helped to sketch and express ideas for the Homepage wireframe. Below are given the stages of ideas development by drawing sketches.

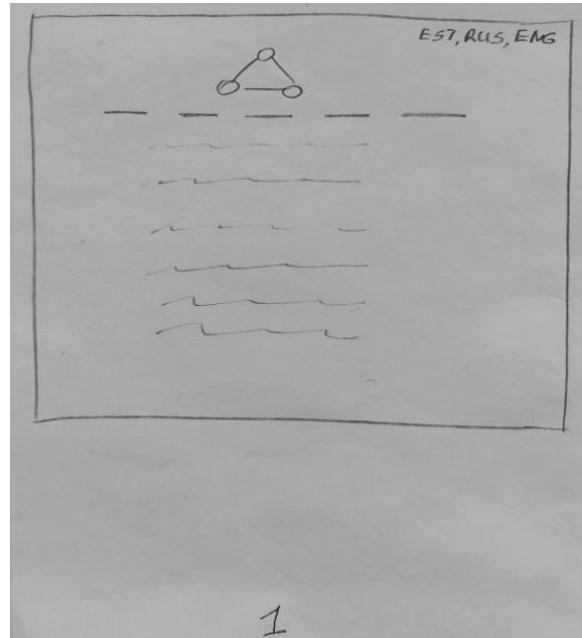


Figure 3. Homepage Sketch 1

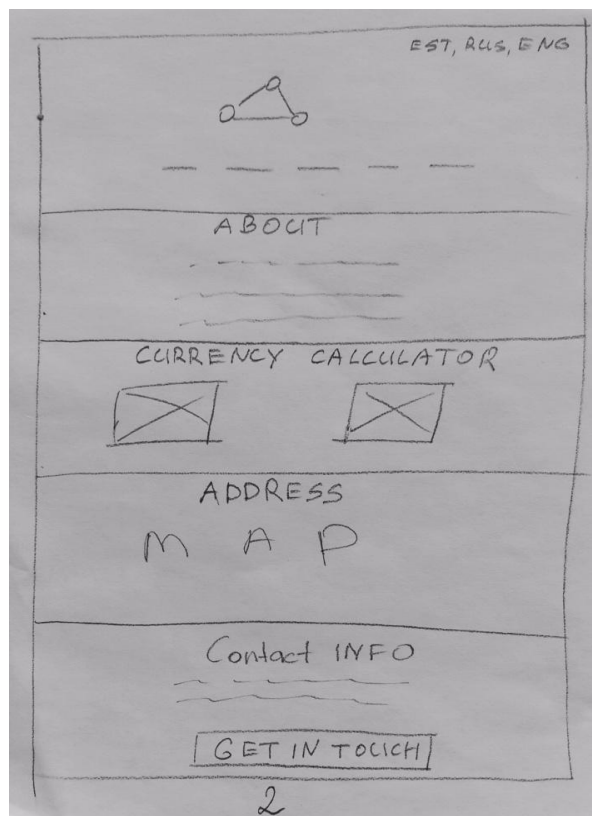


Figure 4. Homepage Sketch 2



Figure 5. Homepage Sketch 3



Figure 6. Homepage Sketch 4

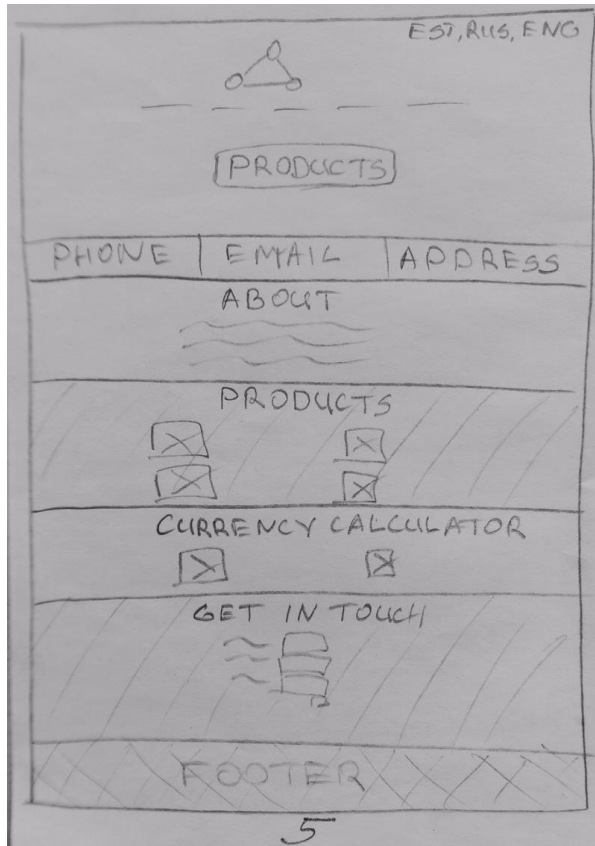


Figure 7. Homepage Sketch 5

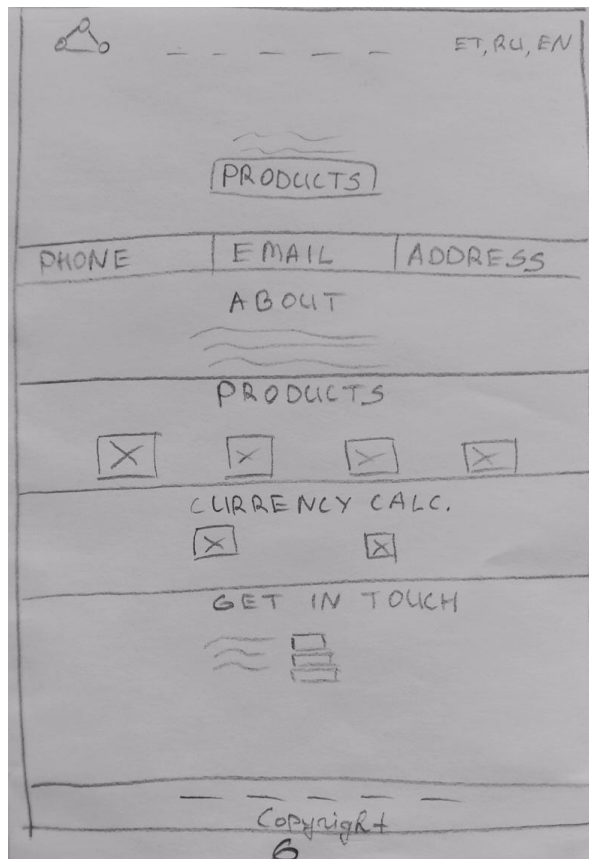


Figure 8. Homepage Sketch 6

Now the main idea of web page structure is clear. We can process to the next step of the website design implementation.

5.4 Choosing the right CSS Methodology

Before we start creating the wireframe, let us consider best methodologies, which can help us to write maintainable and understandable CSS code. Even though wireframe is mostly a prototype we still have to use CSS. In order to create this prototype as quick as possible, some methodologies could greatly improve time of development of wireframe.

CSS methodology is documented system for writing CSS. It allows us to develop and maintain the front-end easier [28]. It is very important to use methodology because it makes CSS and UI more structured and organized. It is useful to optimize our CSS code. If we work in a team we should provide well-formatted code because it would be easy to understand and use. The correct CSS code can increase development speed. If we want to change something small we can do it easier without breaking everything. The most popular methodologies are OOCSS, SMACSS and BEM [31]. Each methodology is lightly different because it offers different approaches to maintain the CSS. We will take a brief look at these methodologies.

5.4.1 OOCSS

OOCSS stands for Object-Oriented CSS. It was developed in 2009 by Nicole Sullivan [29]. This approach has two main ideas: we should separate the structure and skin, we should separate the container and content (do not refer to tags). In OOCSS methodology we use CSS class selectors [28]. We give to object a class and reference in our CSS.

The idea of OOCSS methodology is following:

```
.header {
    /* header contents */
}

h1.header-title {
    /* h1 element style, only for class header-title */
}
```

Code snippet 1. OOCSS methodology idea

The idea of Non-OOCSS methodology is following:

```

.header {
    /* header contents */
}

.header h1 {
    /* more specificity */
}

```

Code snippet 2. Non-OOCSS methodology idea

OOCSS methodology reduces the amount of code, because we can reuse it. This is principle of DRY (Do not repeat yourself). Unfortunately, our CSS might be hard to support, maintain and manage, because in our project might be too many classes. And this is the main disadvantage of this methodology [28]. Moreover, if we want to change the style of a particular element we should change not only CSS, but also add classes to the markup, because most classes are common [24].

5.4.2 SMACSS

SMACSS stands for Scalable and Modular Architecture for CSS. It was created in 2011 by Jonathan Snook. The main idea is to divide styles into 5 sections [30].

1. Base Rules – styles of the main elements, for example body, button, header. This section mainly uses element selectors, descendent selectors or child selectors. Classes and ID are not used [30].

Example:

```

body {
    margin: 0;
}

h1 {
    color: red;
}

```

Code snippet 3. SMACSS Base Rules idea

2. Layout Rules – layout styles, for example styles for header and footer sizes [30].

Example:

```

#header,
#footer {
    width: 700px;
}

.l-sidebar {
    width: 320px;
}

```

```
}
```

Code snippet 4. SMACSS Layout Rules idea

3. Modules Rules – blocks that can be used multiple times on a single page, for example widgets, navigation bars or dialogs. It is not recommended to use ID and element selectors defining the rule for a module. We should use class names [30].

Example:

```
.module > h1 {  
    padding-left: 10px;  
}  
  
.search-form {  
    background-color: #0E333D;  
}
```

Code snippet 5. SMACSS Modules Rules idea

4. State Rules – states of the modules and the skeleton. It is recommended to use the keyword *!important* to override the state style of a more complex rule set. These styles are used to specify the current state of an element [30].

Example:

```
is-blocked {  
    display: none;  
}
```

Code snippet 6. SMACSS State Rules idea

5. Theme Rules – describe styles of decorations, for example colours or images.

The naming convention is to use prefix consists of 2-4 characters of module or layout elements (for example nav, l-).

SMACSS approach simplifies the code writing and it attracts quite a number of developers. We do not need to use names for base styles as we can use type selectors such as h1, p or a [28]. Using SMACSS we can create independent modules, the code is easy to read.

5.4.3 BEM

The BEM abbreviation stands for Block, Element and Modifier. It was created in Yandex to develop sites that need to be done quickly and should be maintained for

many years [32]. Block represents an object and can be used in a number of places on the site (for instance a menu or a login form). Element is part of the block and do not have a standalone meaning (for instance a menu item or a login button). Modifier represents variations of a block, we can use it to change appearance or state, or behavior (for instance a menu for a footer or header is modified to look differently). It gives a strict way to order CSS classes into independent modules [25].

Block name may consist of: latin letters, digits, dashes [26].

The idea is following:

```
.block {}
```

Code snippet 7. BEM Block idea

Element name may consist of: latin letters, digits, dashes, underscores. The class name should contain the block name, two underscores and the element name [26].

The idea is following:

```
.block__element {}
```

Code snippet 8. BEM Element idea

Modifier name may consist of: latin letters, digits, dashes, underscores. The class name should contain the block name plus two dashes or the element name plus two dashes [26]. The idea is following:

```
.block--modifier {}  
.block__element--modifier {}
```

Code snippet 9. BEM Modifier idea

Advantages of the BEM methodology:

- Modularity – block styles are independent of other page elements. We change the document's structure and the CSS will not break [36]. We use only class names and there is no structural dependency [27].

- Reusability – as blocks are independent we can reuse them and it means that the amount of CSS code is decreasing [36].
- Structure – the structure of CSS code is simple and easily understandable [36].

Final Choice

All three described above methodologies are interesting and they share common goals. Using one of these methodologies will make our code more maintainable and organized. BEM, SMACSS and OOCSS methodologies suggest adding classes to HTML and referring to them on CSS writing styles. They separate the content from structure that helps to develop and maintain projects. We can choose methodology that seems to be more suitable for the type of the project. For big project with many developers it would be better to use BEM methodology as naming conventions gives more structured approach.

The author decided to choose the BEM methodology. It provides a more fine grained logic division between elements and this methodology (from the author's point of view) seems to be less complicated than the others methodologies.

5.5 Creating Homepage wireframe

At this type of stage of the website development, the author was focused on creating Homepage layout – the main website's structure. The wireframe was planned with consideration to details such as alignment, margins, paddings and space sizes.

The author started with writing the correct HTML code. The CSS styles like colors, gradients and fonts were ignored. Was used only one generic font. Instead of images, the author used rectangular boxes with expected image's sizes.

The author took into account all UI and UX design principles described above in *Chapter 4*. Below (Figure 9) is the Homepage wireframe based on the client's requirements:

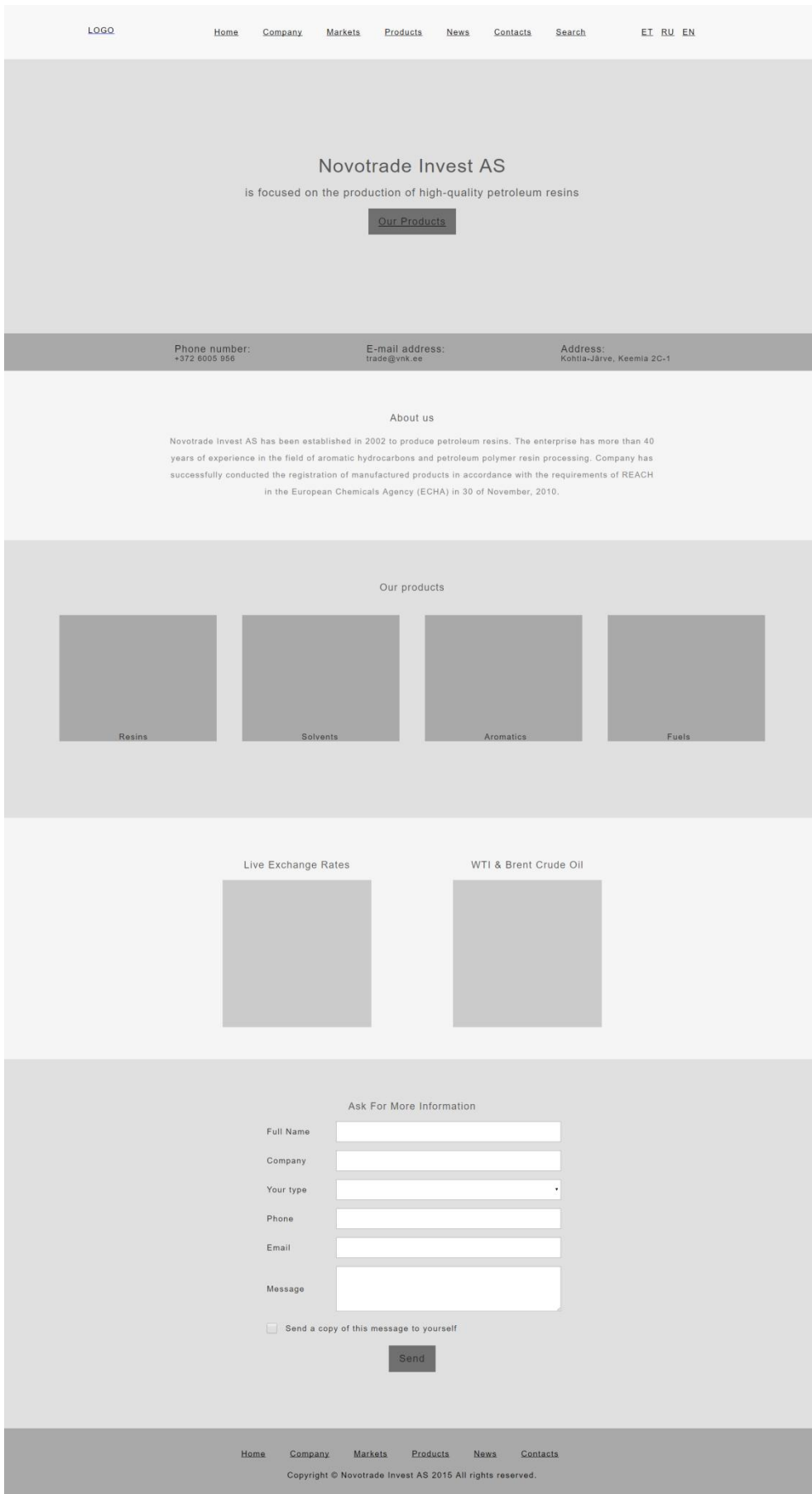


Figure 9. Homepage wireframe created with HTML and CSS

5.6 Visual design

At this type of stage of the website development, the author was focused on creating Homepage more attractive.

Figure 15 shows the final visual version of the Homepage. As we can see the visual design is a slightly different from the original Wireframe in Figure 9.

We will next consider factors that were taken into account creating the Visual design.

5.6.1 Color Palette

The website is based on the following color palette:

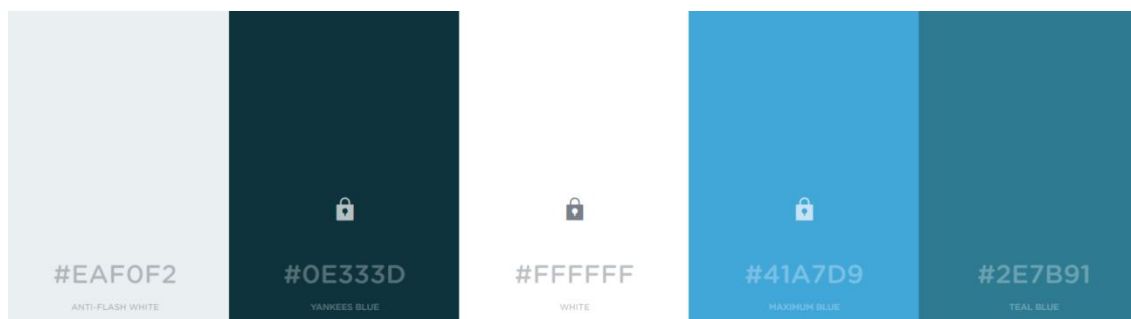


Figure 10. Color Palette for Website

To generate this palette was used the tool Colors.co [40]. This color palette was chosen according to the company's logo. The logo is blue (#41A7D9). The first color (from left to right) is used for background. The second color is used for body text and for background. The third color is used for headlines and background. The last two colors are used as accents. [4]

5.6.2 Background Image

The header background image is the Molecule chain (non-functional requirement 2.2.3.). A large image on the background of the site looks impressive and original. It allows making the website to stand out among others and to be memorable because it is an eye catching image that has an impact [14].

5.6.3 Parallax Background Scrolling

The background molecule image is with parallax effect. Parallax effect is a change in the visible position of an object relative to a distant background. It makes an illusion of

3D motion scrolling the web page because objects and background image scroll at different speeds [14]. This effect engages users and improves the user experience [35].

5.6.4 Animation

When hovering the mouse over a product the image size is getting bigger. This animation effect is small and it fits very naturally into the overall appearance of the page. A user is intuitively ready for it and it improves the experience of interaction with the page. An animated image draws the user's eye and it would benefit the company [39].



Figure 11. Animation for products

5.6.5 Shades and gradients

Our screens are flat and when creating a design we should imitate the physical world. We can make some elements to be 3D. To imitate the buttons to be real we can use gradients and shadows. The top of elements on the site should be lighter and the bottoms darker. [4] Products images have a dark bottom edge and they are lightly brighter at the top (Figure 11).

Buttons have a gray shadow:

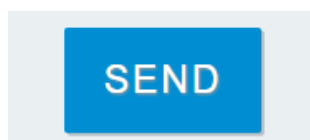


Figure 12. Button with gray shadow

5.6.6 Navigation

Navigation gives the user an understanding how the website is organized and tells what the website contains. The navigation is very important and it is an integral part of the web page. A good navigation allows people to get that information which they were seeking. If the user cannot find that what he is looking for he will leave the website. In order to make the website understandable and usable the navigation should be straightforward and intuitive. [1]

To make a perfect navigation we can use an HTML list of links and style them with CSS [19]. In navigation menu the current web page link is decorated by underline. It helps the user to understand the current place on the website [39] (non-functional requirements 2.2.6. and 2.2.7.).



Figure 13. Current page link in Navigation Menu

5.6.7 Icons

The contact information is easily accessible and it is accompanied by an icon (non-functional requirement 2.2.8.). The usefulness of icons was described in *Chapter 4.2 “Take advantage of conventions”*.



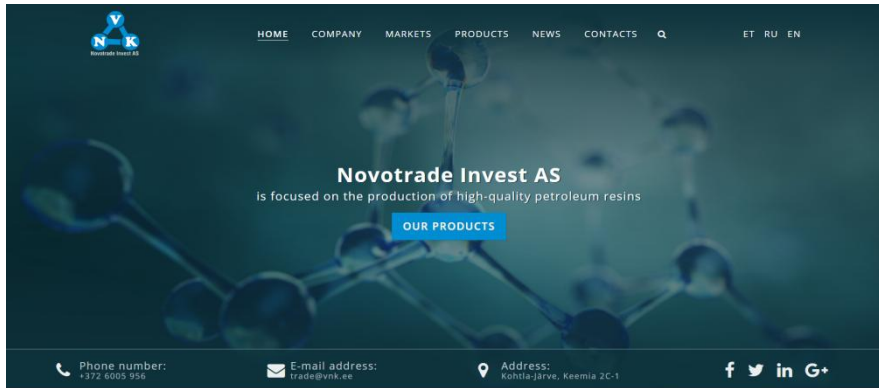
Figure 14. Contact information with icons

5.6.8 Fonts

The font is ‘Open Sans’. The font is easy-to-read, it is clean and simple. It was selected based on the target audience. To make the text more readable the smallest font-size is 18px (non-functional requirement 2.2.4). Also were used CSS properties *line-height* and *letter-spacing* to increase the readability.

5.6.9 Final Homepage design

After applying all those design principles we got the following result:



ABOUT US

Novotrade Invest AS has been established in 2002 to produce petroleum resins. The enterprise has more than 40 years of experience in the field of aromatic hydrocarbons and petroleum polymer resin processing. Company has successfully conducted the registration of manufactured products in accordance with the requirements of REACH in the European Chemicals Agency (ECHA) in 30 of November, 2010.

OUR PRODUCTS



LIVE EXCHANGE RATES

Euro Exchange Rate	
1 EUR	
Russian Ruble	63.9
American Dollar	1.12
10 May 17	

WTI & BRENT CRUDE OIL



ASK FOR MORE INFORMATION

FULL NAME
 COMPANY
 YOUR TYPE
 PHONE
 E-MAIL
 MESSAGE


Figure 15. Homepage Visual Design

5.7 Responsive design

The website is mobile-friendly. All pages are responsive, the information is displayed correctly on different screens.

To create the responsive design we can use Flexbox. Flexbox (Flexible Box Layout Module) is a new layout mode in CSS3. Elements, blocks can be compressed and stretched according to specified rules, occupying the necessary space. On different screen sizes, these elements and blocks behave predictably [41].

The website on small screen fully complies with its function:



[HOME](#)
[COMPANY](#)
[MARKETS](#)
[PRODUCTS](#)
[NEWS](#)
[CONTACTS](#)
[RU](#) [EN](#)

Novotrade Invest AS
is focused on the production of high-quality petroleum resins

[OUR PRODUCTS](#)

Phone number:
 +372 6005 956

E-mail address:
 trade@nink.ee

Address:
 Kohlarjärve, Keemia 2C-1

[f](#) [t](#) [in](#) [G+](#)

ABOUT US

Novotrade Invest AS has been established in 2002 to produce petroleum resins. The enterprise has more than 40 years of experience in the field of aromatic hydrocarbons and petroleum polymer resin processing. Company has successfully conducted the registration of manufactured products in accordance with the requirements of REACH in the European Chemicals Agency (ECHA) in 30 of November, 2010.

OUR PRODUCTS



LIVE EXCHANGE RATES

Euro Exchange Rate	
1 EUR	
Russian Ruble	63.9
American Dollar	1.12
18 May 17	

WTI & BRENT CRUDE OIL



ASK FOR MORE INFORMATION

FULL NAME

COMPANY

YOUR TYPE

PHONE

E-MAIL

MESSAGE

[SEND](#)

[HOME](#)

[COMPANY](#)

[MARKETS](#)

[PRODUCTS](#)

[NEWS](#)

[CONTACTS](#)

Copyright © Novotrade Invest AS 2015. All rights reserved.

Figure 16. Homepage version on a small screen (iPad Pro screen)

6 Developing the WordPress template

6.1 WordPress installation on a local computer

In order to work with the site on WordPress, we should install the WordPress on a local computer. It gives the following opportunities [37]:

- Fast files transfer. It is possible to transfer files, make changes to WordPress, install plugins without having to click on the link to the site.
- An offline local installation helps to experiment and no one will see what exactly we are doing and what changes we make on the site.
- Gives an access to all the features of WordPress, including all the plugins that are used for this platform.
- Development in offline mode. It is possible to download quickly the entire site and edit it offline.

XAMPP is the most popular PHP development environment. XAMPP is a completely free and is created with open source code to be easy to install and use. [38]

To install WordPress on a local computer the author did the following steps:

1. Downloaded the XAMPP software and launched Apache and MySQL servers.
2. Downloaded the latest version of WordPress and extracted the contents of the ZIP file called `wordpress.zip`.
3. As the WordPress uses a database, so the author created the database named “vnk” using the following URL <http://localhost/phpmyadmin/>. In file `wp-config-sample.php` the author edited the data according to the name of the database:

```
/** The name of the database for WordPress */
define('DB_NAME', 'vnk');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', '');
```

```
/** MySQL hostname */
define('DB_HOST', 'localhost');
```

Code snippet 10. New data with database name

4. Saved the file and renamed it into “*wp-config.php*”.
5. Installed WordPress using URL in browser ‘*localhost/wordpress*’. Then entered the Site Title “Novotrade Invest AS”, username, password and e-mail.

6.2 Organizing WordPress entry files

The aim of this work is to create a website with the own custom theme. The WordPress theme is the HTML code to create a structure that describes where different components will live on a website. The CSS code describes what those components will look like [6].

The first thing we need to do is to create a new folder with name “*vnk-theme*”. This is the directory: *C:\xampp\htdocs\wordpress\wp-content\themes*. Then we create a *style.css* and *index.php* files to set up a valid theme. In the *style.css* file, we should write information which WordPress needs. The WordPress will gather information about the theme. This is the standard information that WordPress needs to recognize the theme and to give its preview:

```
/*
Theme Name: Novotrade Invest AS theme
Theme URI: https://viktoriamajerovits.wordpress.com/
Author: Viktoria Majerovits
Author URI: Viktoria Majerovits
Description: The theme is created for the Estonian chemical
industry company Novotrade Invest AS. It is the perfect theme
for business web site.
The theme is with multiple custom options in the
Administration Panel. The theme works in many languages and
on any device.
Version: 1.0
License: GNU General Public License v2 or later
License URI: http://www.gnu.org/licenses/gpl-2.0.html
Text Domain: vnk-theme
Tags: industry, chemical, business, responsive, clean,
minimal, custom admin options, language-support.
*/
```

Code snippet 11. Information about the theme

The *index.php* file visualizes the theme. It is the basic file and the file that WordPress access by default as soon as a user hit the website. Inside the author created a basic website structure with a HTML. Since the goal is to output here the WordPress content through the author's theme there will be a structured approach [6]. The author divided the page into sections. The top of the WordPress theme - a header area (*header.php*), the content area (*content.php*) and the bottom - footer area (*footer.php*). This code by WordPress recognizes that the author included the header, content and footer inside the *index.php*:

```
<?php

get_header(); ?>

<?php if (have_posts()):
    while (have_posts()): the_post(); ?>
        <?php the_content(); ?>
    <?php endwhile; endif; ?>

<?php get_footer(); ?>
```

Code snippet 12. Main template for WordPress Home page.

6.3 Adding a theme preview image on WordPress

A preview image is important because it gives a nice little preview of what the theme will look like. To create a nice preview of the theme we can generate a PNG image with 1200x900 pixels. We can use the Homepage screenshot and the name can be only screenshot.png [20].

The result is following:

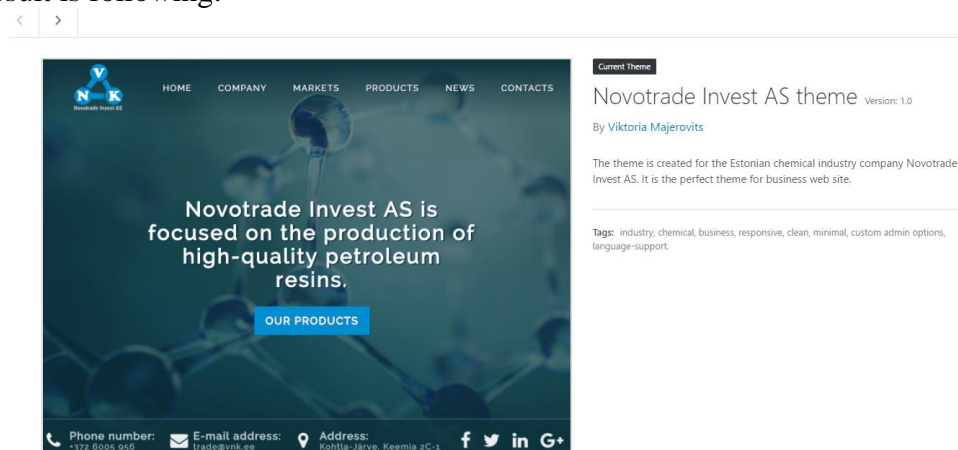


Figure 17. Preview of the new theme

6.4 Organizing and plugging in existing style sheets

The author has created a new file *vnk.css*. This is a basic file to customize and style the theme. To include CSS file the author has created a new important file *functions.php*. WordPress automatically searches for that file when it loads a theme. The file *functions.php* is essentially treated as a theme-specific plugin. Functions in the file *functions.php* will be automatically loaded whenever the theme is in use [6].

```
function vnk_load_scripts()
{
    wp_enqueue_style('vnk', get_template_directory_uri() .
'/css/vnk.css', array(), '1.0.0', 'all');
}
add_action('wp_enqueue_scripts', 'vnk_load_scripts');
```

Code snippet 13. Code to include the CSS file

In *header.php* file was also included the function *wp_head()* to initialize WordPress and ensure that all core functions work properly. Similarly, the *footer.php* must contain *wp_footer()* to allow WordPress functions to manage footer settings. Without it, unexpected results will occur when WordPress uses the custom theme. [8]

6.5 Creating the Header Section

In order to display information about the current site, the WordPress has a function *bloginfo()*. The information is gathered from the information supplied in User Profile and General WordPress Settings [8].

In author case this is the Site title “Novotrade Invest AS”:

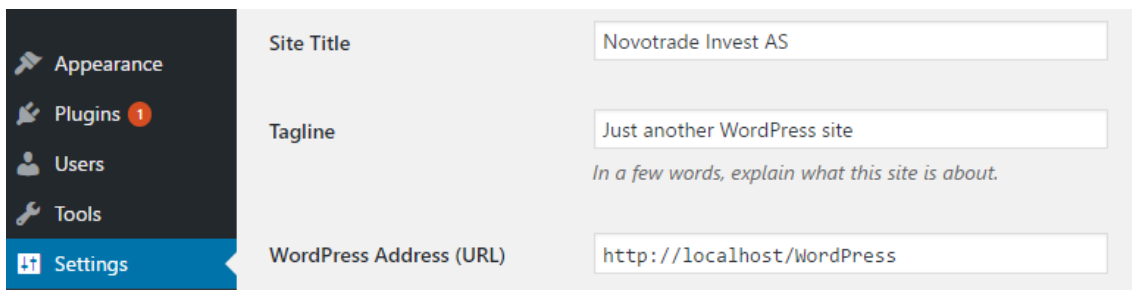


Figure 18. Site Title

This code prints a result to the browser:

```
<title>
  <?php bloginfo('name');
  wp_title(); ?>
</title>
```

Code snippet 14. Function bloginfo() to display the Site Title

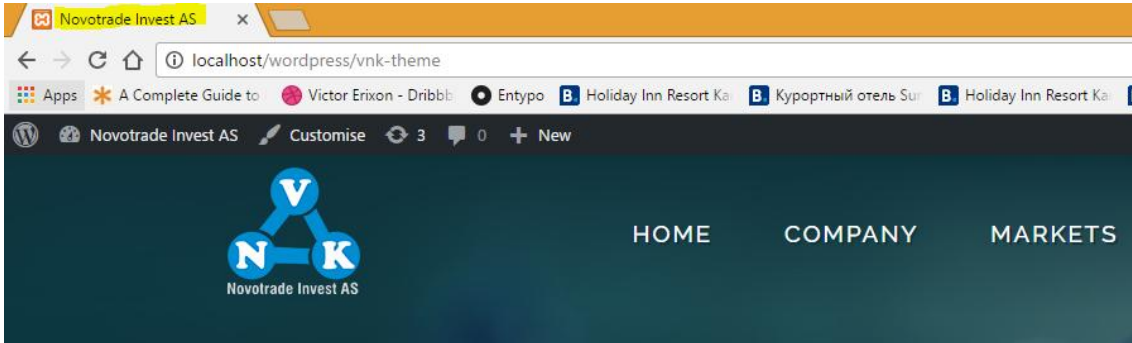


Figure 19. Site Title printed in the browser

6.6 Logo

The author used the following code to add the logo:

```

```

Code snippet 15. Wrong code to display the logo

But the logo of the website was not displayed:

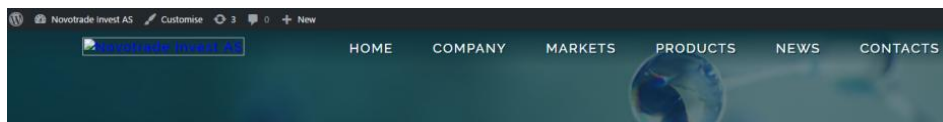


Figure 20. Logo is not displayed

In order to find and get the image path in WordPress we should use function *get_bloginfo()*. This function retrieves information about the current site [7].

This is the correct code to display the logo:

```

```

Code snippet 16. Function get_bloginfo() to display the Logo

6.7 Creating a custom Navigation Menu

How to create a custom menu and how to display specific pages on a specific section of the theme? First of all, we have to go inside the administration panel and create some generic pages structure. The author created Home, Company, Markets, Products, News and Contacts pages:

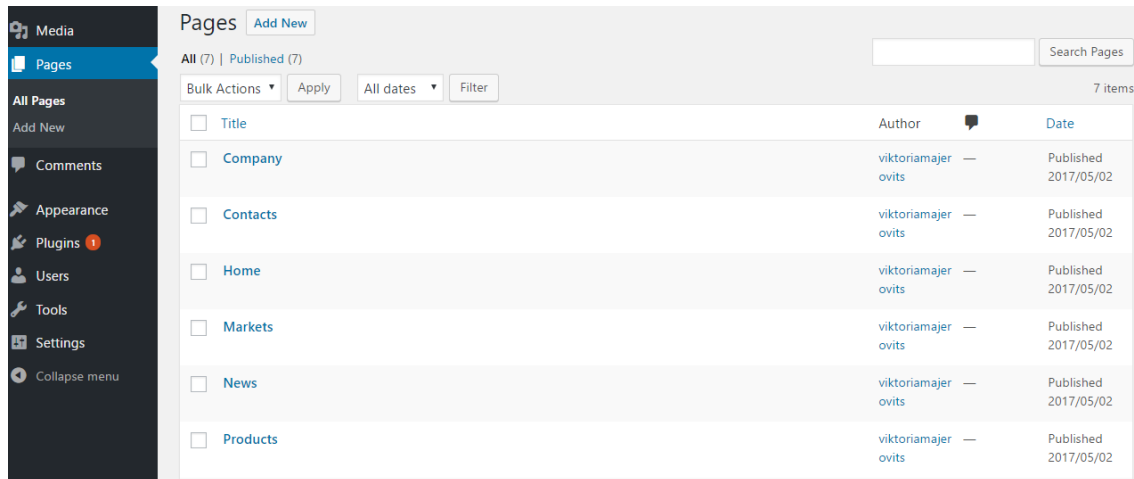


Figure 21. Pages for the Navigation Menu

Usually, inside the appearance menu we can find a subsection to generate our own menu, but here we cannot find it. By default, WordPress gives the ability to activate specific theme features through the *functions.php* file. We need to activate Navigation Menu feature to implement it inside the theme. To do that we can use the WordPress function *add_theme_support()* [21].

This is a new function to create the menu:

```
function vnk_register_nav_menu()
{
    add_theme_support('menus');
}
add_action('init', 'vnk_register_nav_menu');
```

Code snippet 17. Function *add_theme_support()* to display Menus

As a result inside the submenu there are the menus options:

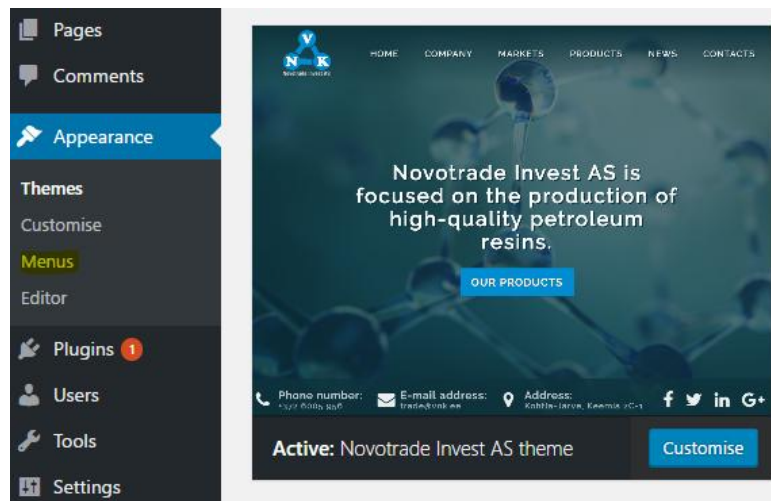


Figure 22. New option - Menus

Before creating a new menu and adding pages to generate the custom menu we should create a custom menu inside the functions panel. It is a specific unique menu that gives the ability to use it wherever we want in the administration panel. It allows to recall and decide which menu we want to display in which specific page's section. We can use the WordPress function `register_nav_menu()` to create custom menus for use in a theme [22]:

```
register_nav_menu('primary', 'Header Navigation');
register_nav_menu('secondary', 'Footer Navigation');
```

Code snippet 18. Function `register_nav_menu()` to create custom menus

The result is following:

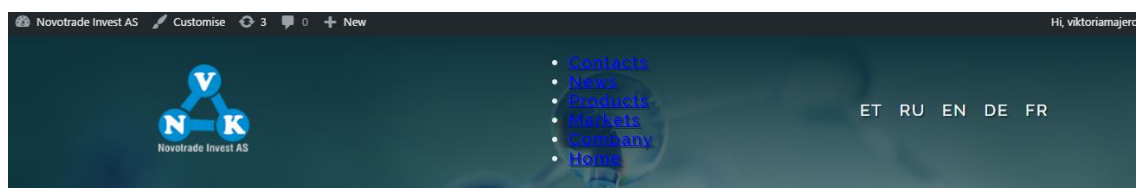


Figure 23. Custom Menu

Inside the menu's section appeared a new option for the Header and Footer Navigation:

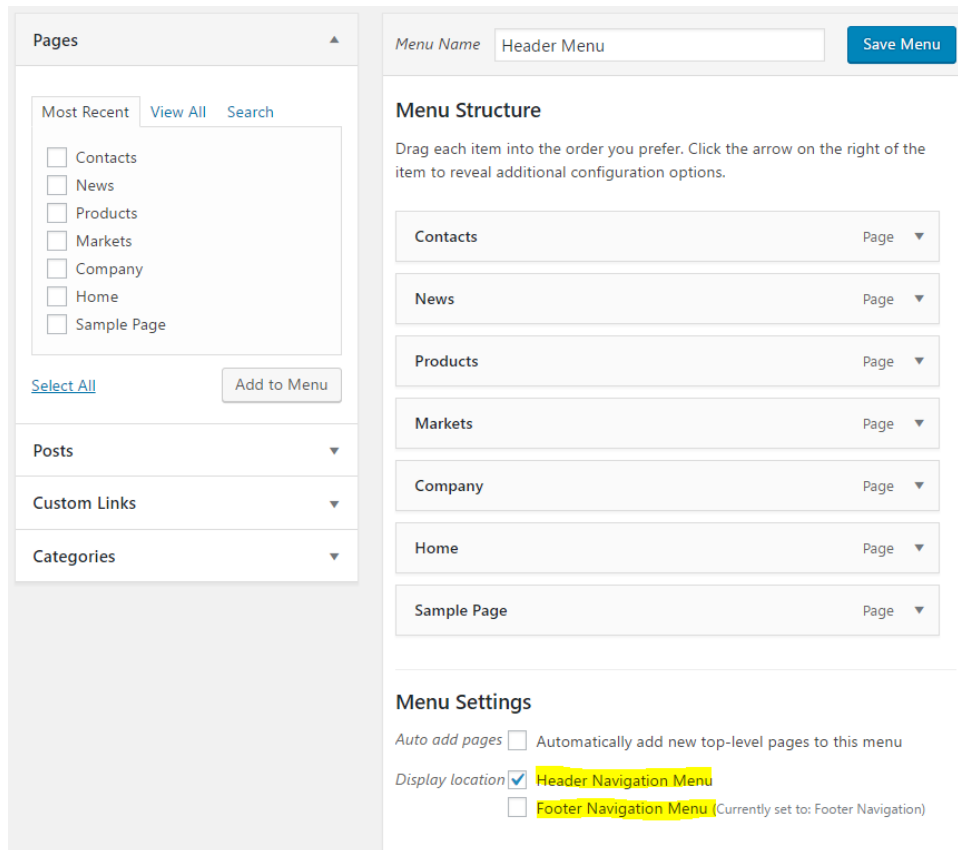


Figure 24. Menu Settings

To print menus in the footer and header sections the author added the following code in *footer.php* and *header.php* files:

```
<?php wp_nav_menu(array('theme_location' => 'primary'));
<?php wp_nav_menu(array('theme_location' => 'secondary')); ?>
```

Code snippet 19. Function `wp_nav_menu()` to display navigation menus

6.7.1 Styling of the Navigation Menu

The author already has styled the Navigation menu in *Chapter 5.6.6*. The styles do not work because WordPress automatically adds a new container menu and a list of properties. The author has removed the default WordPress container around the menu, `` tag wrapper and added a new file called *walker.php*:


```

<?php
    wp_nav_menu(array(
        'theme_location' => 'primary',
        'container' => false,
        'items_wrap' => '%3$s',
        'walker' => new Description_Walker
    ));
?>

```

Code snippet 20. Adding a new file walker.php

Inside the file *walker.php* the code was taken from the following source [10]. This file is useful because it helps to create links with own custom style. WordPress uses the `` and `` tags. Walker helps to create navigation menu in plain `<a>` tags [13].

This is the final results for the Header and Footer sections:

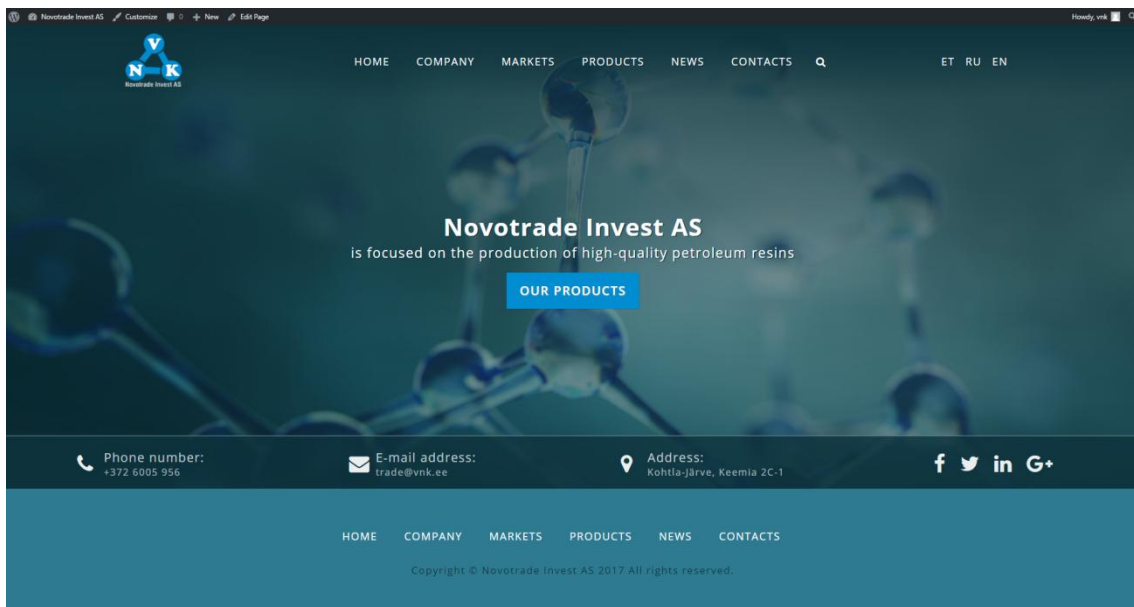


Figure 25. Final result for the Header and Footer sections

6.8 Main Settings

In Administration panel in General Settings is possible to configure the Site Title and Tagline for the website:

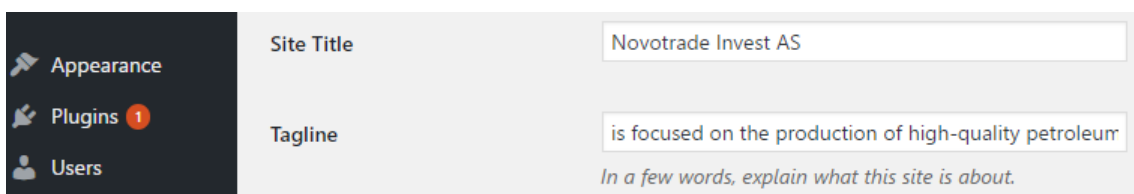


Figure 26. Site Title and Tagline (Administration Panel)

To display this information on the site we can use two WordPress tags:

```
bloginfo('name') and bloginfo('description').
```

The `bloginfo` displays information about the current site from the Administration panel. Name displays the „Site Title“ set in General Settings. Description displays the Tagline set in General Settings [11].

The modified HTML code is following:

```
<div class="site-title-tagline">
  <h1 class="site-title"><?php bloginfo('name') ?>
</h1>
  <p class="tagline">
    <?php bloginfo('description') ?>
  </p>
</div>
```

Code snippet 21. Modified HTML code to display the Site Title and Tagline

The result is following:



Figure 27. Site Title and Tagline

6.9 Making background image customizable

The client should be able to upload a custom header background image for the website without touching the CSS code. To support custom header image theme needs to use the `add_theme_support()` function.

The author added the code in `functions.php` file [12]:

```

add_theme_support('custom-header', array(
    'default-image' => '',
    'flex-height' => true,
    'flex-width' => true,
));

```

Code snippet 22. Function `add_theme_support()` to upload a header image

This image should be a CSS background image, so in *header.php* file was added the following code [12]:

```

<?php if (get_header_image()) : ?>
    <div class="page-header" style="background-image:
url(<?php header_image(); ?>);">
    <?php endif; ?>

```

Code snippet 23. Code for CSS background image

A new subsection appeared inside the appearance submenu:

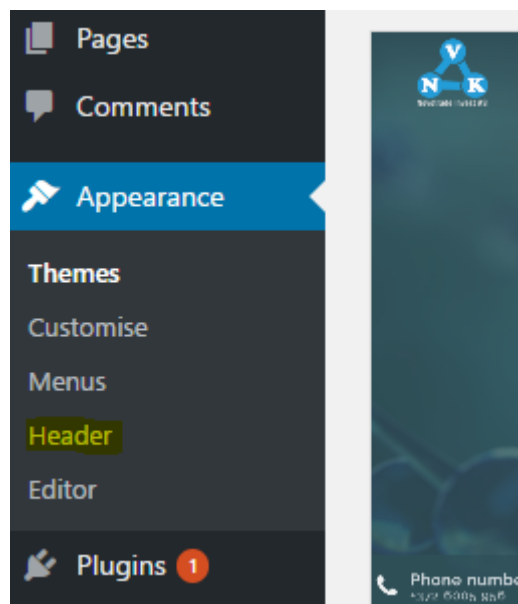


Figure 28. New Subsection – Header

The new background picture was added through the Administration Panel:

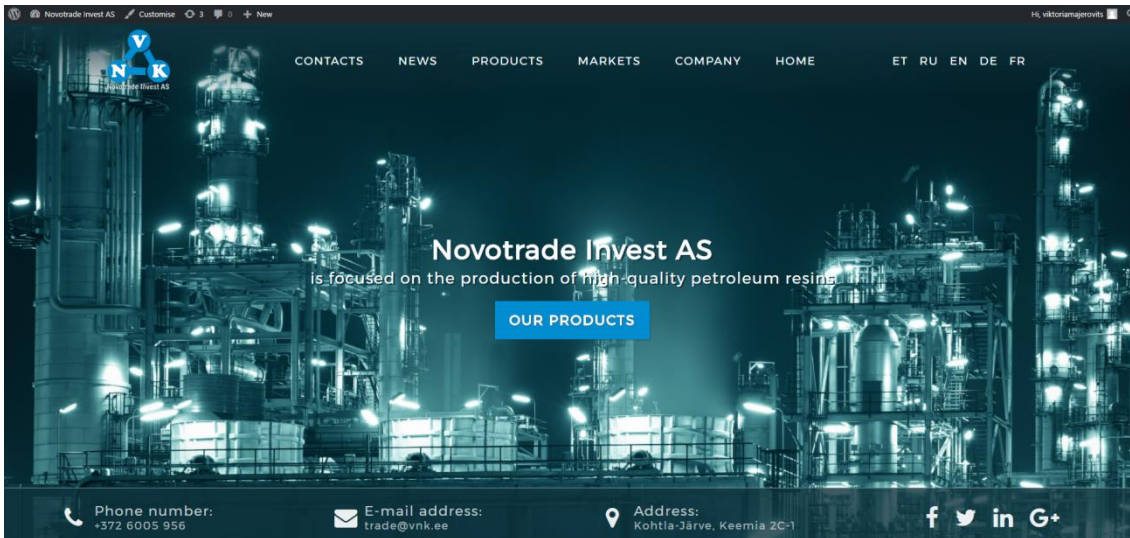


Figure 29. New background picture

6.10 Creating a Custom Static Front Page Template

At the moment on our Homepage, we have only two areas the header area and footer area. In order to add the content for the Homepage, we need to create a new file front-page.php. This file creates a custom website front page template [42]. In this file we need to add our HTML structure and include the header and footer areas.

```
<?php get_header(); ?>

<!-- HTML code for the static front page -->

<?php get_footer(); ?>
```

Code snippet 24. Structure for the custom static Front page

As the result, the Homepage looks exactly as it was on Figure 15.

7 Summary

The main purpose of this study was to document and to show in practice how to create a web page template for WordPress CMS. This work illustrates in details every process of creating the website starting from the planning stage and ending with the content management system development. As a result was created a template with an exclusive design for the company Novotrade Invest AS. In the future, this work may be considered as a further manual for creating websites for companies using WordPress content management system.

In order to achieve the goal this work was divided into five stages:

- Were identified the functional and nonfunctional requirements.
- The analysis was carried out on the most popular content management systems on the basis of which was selected the WordPress CMS.
- Were considered the main UX/UI principles for creating a more accessible web page.
- The design for the website was implemented and it went four phases: planning - sketching of the website, prototyping the wireframe, developing the design and adapting the existing design for various mobile devices.
- Were considered the most popular CSS methodologies and was chosen BEM methodology to develop the interface.
- Was added functionality to manage site content through WordPress.

During the process of study and implementation of this work, author greatly improved skills in the development of web pages. Author acquired new skills in a user interface and user experience design which could be used in practice. The most valuable knowledge was obtained by using CSS methodology called BEM.

It was the first experience of using BEM methodology and it was very successful trial. The author plans to continue to work as a front-end developer and the obtained skills could be very useful in the near future.

The code of this work is available at <https://github.com/viktoria17/loputoo>. The template is not yet finished and because of that, it cannot be used on a production web page. Some template parts also require confirmation and further improvements and clarification from the client.

As a result of this work, we created WordPress template which is based on the UI/UX principles. In the future, the company Novotrade Invest would be able to use it for their website and easily manage the content of their web pages.

Today it is very important for the company to have a good looking website. But in order it would be really professional and attractive it is necessary to spend a lot of time on it and solve certain problems. By the end of the work, it became more evident that sometimes it is easier to create your own WordPress theme than to waste the time of adapting the existing one, which may not fulfill the functional or non-functional requirements.

Kokkuvõte

Selle dokumendi peamine eesmärk on dokumenteerida ja näidata praktikas, kuidas luuakse WordPressi malle. See töö selgitab detailselt malli loomise protsessi, alustades planeerimisega ja lõpetades malli arendamisega. Antud töö tulemusena loodi WordPress malli firma Novotrade Invest AS kodulehe jaoks. Seda tööd saab käsitleda WordPressi malli arendamise juhendina.

Selleks, et saavutada eespool nimetatud eesmäärke, on töö jaotatud viieks osaks:

- On tuvastatud funktsionaalsed ja mittefunktsionaalsed nõuded.
- On tutvustatud ja arvesse võetud UX/UI disainimisprintsipiibid.
- Mall on arendatud, läbides 4 etappi: Veebilehe planeerimine (joonistamine paberil), wireframe'i prototüüpimine, disaini arendamine ja adapteerimine erinevate mobiiliseadmete jaoks.
- Läbi on vaadeldud populaarseimad CSS'i meetodid ja valitud BEM meetodika liidese arendamiseks.
- WordPressi mallile on antud võimalus majandada ja muuta sisu läbi kasutajaliidese.

Töö jooksul autor parandas oma oskusi veebilehtede loomises. Väga kõvasti kasvas kujundamise ja disainimise oskus. Autor sai palju teadmisi ja kogemusi UI'st ja UX'st. Eriti paranes oskus kirjutada CSS'i, kuna autor rakendas BEM meetodikat koodi kirjutamisel. See oli esimene kord, millal autor proovis kasutada BEM meetodikat ja võib pidada antud proovi väga edukaks.

Töö lõpuks saadi töötava malli, mis vastab ülalpool mainitud funktsionaalsetele ja mittefunktsionaalsetele nõuetele.

Töö lähtekoodi saab vaadata siin: <https://github.com/viktoria17/loputoo>. Töö esitamise ajaks mall ei olnud lõpuni tehtud. Seepärast ei saa vaadata kuidas ta töötab vnk.ee kodulehel. Mõned detailid vajasisid kinnitamist ja täiendavaid selgitusi kliendi poolt.

Tänapäeval on hea väljundusega koduleht üsna tähtis. Kuigi sai selgeks, et kui tahta midagi originaalset ja huvitavat, tuleb palju aega kulutada ja vahepeal lahendada

mittetriviaalseid probleeme. Tööprotsessi lõpuks sai selgeks, et vahepeal võib aega säästa tänu sellele, et alustada malli loomisega ise otsast peale, selle asemel, et otsida ja kohandada olemasolevaid lahendusi, mis võivad mõne aja pärast osutuda ebasobivaiks etteantud probleemi jaoks.

References

- [1] S. Krug, Don't Make Me Think! A Common Sense Approach to Web Usability, Second Edition, 2006.
- [2] C. Bank, "Understanding Web UI visual Hierarchy," 18 11 2017. [Online]. Available: <https://www.awwwards.com/understanding-web-ui-visual-hierarchy.html>. [Accessed 22 4 2017].
- [3] E. D. Kennedy, "7 Rules for Creating Gorgeous UI (Part 1)," 14 11 2014. [Online]. [Accessed 11 4 2017].
- [4] E. D. Kennedy, "7 Rules for Creating Gorgeous UI (Part 2)," 14 11 2014. [Online]. Available: <https://medium.com/@erikdkennedy/7-rules-for-creating-gorgeous-ui-part-2-430de537ba96>. [Accessed 13 4 2017].
- [5] Russ Unger, Carolyn Chandler, A Project Guide to UX Design, 2012.
- [6] Mick Olinik, Raena Jackson Armitage, The WordPress anthology, 2011.
- [7] "Code Reference," [Online]. Available: https://developer.wordpress.org/reference/functions/get_bloginfo/. [Accessed 25 04 2017].
- [8] "Code Reference," [Online]. Available: <https://developer.wordpress.org/reference/functions/bloginfo/>. [Accessed 25 04 2017].
- [9] "Code Reference," [Online]. Available: https://developer.wordpress.org/reference/functions/wp_nav_menu/. [Accessed 25 04 2017].
- [10] "Menu items description? Custom Walker for wp_nav_menu()," 05 04 2017. [Online]. Available: <https://wordpress.stackexchange.com/questions/14037/menu-items-description-custom-walker-for-wp-nav-menu/14039#14039>. [Accessed 30 04 2017].
- [11] "Code Reference," [Online]. Available: <https://developer.wordpress.org/reference/functions/bloginfo/>. [Accessed 01 05 2017].
- [12] "How To Use WordPress Custom Headers," [Online]. Available: <https://answers.themler.com/articles/42119/how-to-use-wordpress-custom-headers>. [Accessed 01 05 2017].
- [13] "custom-nav-walker-usage.php," [Online]. Available: <https://gist.github.com/kosinix/5544535>. [Accessed 30 04 2017].
- [14] "25 основных трендов веб-дизайна 2017 года по версии DesignNonstop," 10 01 2017. [Online]. Available: <http://www.designonstop.com/webdesign/trends/25-osnovnyx-trendov-veb-dizajna-2017-goda-po-versii-designonstop.htm>. [Accessed 07 05 2017].
- [15] "What is UX Design? 15 User Experience Experts Weigh In," 16 09 2015. [Online]. Available: <https://www.usertesting.com/blog/2015/09/16/what-is-ux-design-15-user-experience-experts-weigh-in/>. [Accessed 15 04 2017].

- [16] W. Staff, “A Case for Coding Your Wireframes,” 30 09 2019. [Online]. Available: <https://www.webdesignerdepot.com/2010/09/a-case-for-coding-your-wireframes/>. [Accessed 25 04 2017].
- [17] “HTML Wireframes,” 19 05 2014. [Online]. Available: <http://bradfrost.com/blog/post/html-wireframes/>. [Accessed 25 04 2017].
- [18] E. LAMPRECHT, “The Difference Between UX and UI Design - A Layman’s Guide,” 25 04 2017. [Online]. Available: <https://careerfoundry.com/en/blog/ux-design/the-difference-between-ux-and-ui-design-a-laymans-guide/>. [Accessed 09 05 2017].
- [19] Patrick J. Lynch, Sarah Horton, Web Style Guide, 3rd Edition, 2009.
- [20] “How to Add a Theme Preview Image on WordPress,” [Online]. Available: <https://www.whitecornercreative.com/how-to-add-a-theme-preview-image-on-wordpress/>. [Accessed 29 04 2017].
- [21] “Code Reference,” [Online]. Available: https://developer.wordpress.org/reference/functions/add_theme_support/. [Accessed 01 05 2017].
- [22] “Codex, Function Reference/register nav menu,” [Online]. Available: https://codex.wordpress.org/Function_Reference/register_nav_menu. [Accessed 02 05 2017].
- [23] P. Ranjit, “Understanding CSS Writing Methodologies,” [Online]. Available: <http://www.hongkiat.com/blog/css-writing-methodologies/>. [Accessed 10 04 2017].
- [24] “Способы организации CSS-кода,” 20 05 2015. [Online]. Available: <https://habrahabr.ru/post/256109/>. [Accessed 12 04 2017].
- [25] P. Catanzariti, “BEM and SMACSS: Advice From Developers Who’ve Been There,” 07 01 2015. [Online]. Available: <https://www.sitepoint.com/bem-smacss-advice-from-developers/>. [Accessed 12 04 2017].
- [26] “GET BEM, Naming,” [Online]. Available: <http://getbem.com/naming/>. [Accessed 15 04 2017].
- [27] M. Shirshin, “Scaling Down The BEM Methodology For Small Projects,” 17 07 2014. [Online]. Available: <https://www.smashingmagazine.com/2014/07/bem-methodology-for-small-projects/>. [Accessed 15 04 2017].
- [28] B. K. Potts, “A Look at Some CSS Methodologies,” 22 06 2015. [Online]. Available: <http://sixrevisions.com/css/css-methodologies/>. [Accessed 15 04 2017].
- [29] Wikipedia, “OOCSS,” 14 05 2017. [Online]. Available: <https://en.wikipedia.org/wiki/OOCSS>. [Accessed 17 05 2017].
- [30] J. Snook, Scalable and Modular Architecture for CSS. A flexible guide to developing sites small and large., 2011.
- [31] Y. Gorodetskiy, “No More Janky CSS: Using OOCSS, SMACSS and BEM,” [Online]. Available: <https://appendto.com/2016/10/no-more-janky-css-utilizing-oocss-smacss-and-bem/?nabe=4714834778390528:1,4834138299564032:1,5488687288942592:0>,

- 5685334715400192:0,6035677076783104:1. [Accessed 17 04 2017].
- [32] “БЭМ МЕТОДОЛОГИЯ,” [Online].
Available: <https://ru.bem.info/methodology/>. [Accessed 17 04 2017].
- [33] “Using CSS Flexible Boxes,” 17 03 2017. [Online].
Available:
https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout/Using_CSS_flexible_boxes.
[Accessed 18 05 2017].
- [34] J. J. Garrett, The elements of user experience: User-Centered Design for the Web and Beyond, Second edition, 2011.
- [35] D. M. Frederick, Effects Of Parallax Scrolling On User Experience And Preference In Web Design., 2013.
- [36] G. BEM. [Online]. Available: <http://getbem.com/introduction/>. [Accessed 15 04 2017].
- [37] J. Fylan, “How To: Install WordPress on your Computer using XAMPP,” 28 1 2017. [Online].
Available: <https://wplift.com/install-wordpress-xampp>. [Accessed 20 3 2017].
- [38] Wikipedia, “XAMPP,” 29 4 2017. [Online].
Available: <https://en.wikipedia.org/wiki/XAMPP>. [Accessed 30 4 2017].
- [39] C. Bank, “Understanding Web UI Elements & Principles,” 07 01 2017. [Online]. Available: <https://www.awwwards.com/understanding-web-ui-elements-principles.html>. [Accessed 23 04 2017].
- [40] Coolors, “Tool for generate color scheme,” [Online].
Available: <https://coolors.co/967d69-bca491-d3bba7-f7c69e-eda468>. [Accessed 19 04 2017].
- [41] w3schools, “CSS3 Flexible Box,” [Online].
Available: https://www.w3schools.com/css/css3_flexbox.asp. [Accessed 25 04 2017].
- [42] Codex, “Creating a Static Front Page,” [Online].
Available: https://codex.wordpress.org/Creating_a_Static_Front_Page.
[Accessed 10 05 2017].
- [43] “Usage of content management systems for websites,” 19 05 2017. [Online].
Available: https://w3techs.com/technologies/overview/content_management/all.
[Accessed 19 05 2017].
- [44] R. MENING, “WordPress vs Joomla vs Drupal,” 08 05 2017. [Online].
Available: <https://websitesetup.org/cms-comparison-wordpress-vs-joomla-drupal/>. [Accessed 20 05 2017].
- [45] R. Support, “CMS comparison: Drupal, Joomla, and WordPress,” 17 05 2017. [Online]. Available: <https://support.rackspace.com/how-to/cms-comparison-drupal-joomla-and-wordpress/>. [Accessed 20 05 2017].

