

UNIwersYTET RZESZOWSKI
WYDZIAŁ NAUK ŚCISŁYCH I TECHNICZNYCH
INSTYTUT INFORMATYKI



Krystian Filipek
134907

Informatyka

Technologie Internetowe - Witryna Virexon

Praca projektowa

Praca wykonana pod kierunkiem
dr. Katarzyna Garwol

Rzeszów 2025

Spis treści

1. Wprowadzenie	6
2. Opis założeń projektu	7
2.1. Wymagania funkcjonalne	7
2.2. Wymagania нефункционалне	8
3. Opis struktury projektu	9
3.1. Użyte technologie	9
3.2. Struktura katalogów	9
3.3. Konwencje kodowania i organizacja stylów	9
3.4. Implementacja efektów wizualnych i optymalizacja	10
3.5. Mechanizm Scroll Snap (Podstrony)	10
3.6. Responsywność (RWD)	10
4. Harmonogram realizacji projektu	11
4.1. Etapy realizacji projektu	11
4.2. Wykres Gantta	11
4.3. System kontroli wersji	11
5. Prezentacja warstwy użytkowej projektu	12
6. Instrukcja uruchomieniowa	13
7. Testowanie poprawności i stopnia optymalizacji kodu	14
7.1. Walidacja HTML oraz CSS - W3C	14
7.1.1. HTML	14
7.1.2. CSS	15
7.2. Audyt wydajności - Lighthouse	16
7.2.1. Strona główna	16
7.2.2. Podstrona	17
8. Podsumowanie	19
Bibliografia	20
Spis rysunków	21
Oświadczenie studenta o samodzielności pracy	22

1. Wprowadzenie

Strona internetowa fikcyjnej korporacji **Virexon** zajmującej się prywatnymi usługami wojskowymi oraz usługami związanymi z eksperymentalnymi terapiami długowieczności stworzona została na potrzeby pierwszego projektu (Zakres HTML + CSS) z przedmiotu **Technologie Internetowe** na 2 roku Informatyki na **Uniwersytecie Rzeszowskim**. Strona jako projekt spełniać ma typowe nowoczesne standardy w tworzeniu stron internetowych, którymi między innymi są **responsywność** i nowoczesny oraz zarazem intuicyjny układ, dzięki któremu użytkownik bez problemu odnajdzie się na stronie internetowej.

Stronę internetową zgodnie z określonymi wymaganiami stworzono tylko i wyłącznie przy pomocy technologii **HTML** oraz **CSS** bez użycia zewnętrznych bibliotek i frameworków.

The website of the fictional corporation **Virexon**, which provides private military services and services related to experimental longevity therapies, was created for the first project (HTML + CSS) in the **Internet Technologies** course for second-year Computer Science students at the **University of Rzeszów**. The project is designed to meet typical modern standards in website development, including responsiveness and a modern, yet intuitive layout, ensuring easy navigation for users.

The website was created in accordance with the specified requirements using only **HTML** and **CSS** technologies, without the use of external libraries or frameworks.

2. Opis założeń projektu

Celem projektu jest stworzenie responsywnej strony internetowej przy użyciu technologii HTML i CSS dla fikcyjnej korporacji/firmy badawczej zajmującej się nauką, wojskowością oraz technologią w prywatnym sektorze. Strona ma za zadanie zbudować wizerunek nowoczesnej, innowacyjnej marki oraz informować o podstawowych informacjach takich jak, chociażby misji, badaniach i zespole firmy.

Podstawowym problemem, który zostanie rozwiązany przez realizację tego projektu, jest brak cyfrowej tożsamości dla nowo powstałej firmy. Źródłem tego problemu jest współczesna specyfika rynku w dobie powszechnej cyfryzacji, gdzie brak wizytówki w sieci w postaci strony internetowej jest równoznaczny z brakiem wiarygodności w oczach potencjalnych klientów. Aby problem został skutecznie rozwiązany, potencjalny zespół musi posiadać wiedzę z zakresu technologii webowych w szczególności kaskadowych arkuszy stylów CSS3, języka struktury HTML5 oraz podstawowego zmysłu estetycznego w projektowaniu interfejsów UI/UX.

Rozwiązanie problemu przebiegnie w kilku zdefiniowanych krokach. W pierwszej kolejności nastąpi zaprojektowanie schematu każdej ze stron czyt. układ treści i nawigacji. Kolejnym krokiem będzie postawienie szkieletu strony przy pomocy technologii HTML5 zaś krokiem kolejnym będzie stopniowe wdrażanie stylów przy użyciu technologii kaskadowych arkuszy stylów. Końcowym i zarazem najważniejszym krokiem będzie optymalizacja responsywności strony na urządzenia mobilne takie jak smartfon bądź tablet poprzez Media Queries.

2.1. Wymagania funkcjonalne

Poniższe wymagania opisują, jakie operacje i interakcje umożliwia strona internetowa użytkownikowi końcowemu:

Nawigacja po sekcjach: Strona musi umożliwiać użytkownikowi płynne przechodzenie po konkretnych sekcjach strony (Home, Our Story, Services, Jobs, Contact) poprzez kliknięcie w odpowiedni załącznik prowadzących do wspomnianych sekcji.

Prezentacja treści: Strona musi wyświetlać sformatowane teksty oraz grafiki w sposób czytelny dla odbiorcy.

Interakcja z elementami graficznymi: Po najechaniu kursorem na zdjęcia bądź elementy paska nawigacji, strona ma odpowiednio reagować poprzez efekty takie jak powiększanie obrazków, podkreślanie elementów i zmiany kolorów.

Animacja powitalna: Podczas załadowania strony głównej, strona automatycznie ma odtworzyć prostą animację powitalną, aby przykuć uwagę użytkownika.

Wizualizacja formularza kontaktowego: Strona musi udostępniać w dolnej części strony głównej formularz kontaktowy z podstawowymi polami na dane z odpowiednio stylowanym interfejsem użytkownika.

Responsywność menu: Na urządzeniach mobilnych strona musi zmieniać układ paska nawigacyjnego na pionowy lub dostosowany do ekranów dotykowych, aby zapewnić podstawową czytelność załączników.

2.2. Wymagania niefunkcjonalne

Poniższe wymagania określają ograniczenia techniczne, standardy oraz cechy, jakimi musi charakteryzować się rozwiązanie:

Środowisko testowe

Testy projektu odbywały się lokalnie na komputerze o przedstawionej poniżej specyfikacji.

- **System operacyjny:** Microsoft Windows 11 Pro
- **Model:** Lenovo Yoga 7 2-in-1 14IML9
- **Procesor:** Intel® Core™ Ultra 5-125H 4.5 GHz
- **Pamięć RAM:** 16GB DDR4
- **Karta graficzna:** Intel® Arc™ graphics (Zintegrowana)
- **Dysk:** Samsung MZAL81T0HDLB-00BL2 1TB
- **Rozdzielczość ekranu:** 2880 x 1800
- **IDE:** Visual Studio Code

Technologie: Strona zgodnie z przyjętymi wytycznymi wykonana ma być przy użyciu czystego HTML5 oraz CSS bez zewnętrznych bibliotek, frameworków (np. Bootstrap, React itd) oraz języka JavaScript.

Ogólna responsywność: Strona musi poprawnie skalować się i być czytelna na urządzeniach zarówno mobilnych aż po monitory desktopowe.

Estetyka i spójność wizualna: Strona musi utrzymywać spójność kolorystyczną, wykorzystując spójną typografię w zależności od sekcji.

Wydażność: Strona powinna charakteryzować się krótkim czasem ładowania dzięki braku ciężkich bibliotek i zewnętrznych skryptów.

Kompatybilność: Witryna powinna wyświetlać się poprawnie zarówno na przeglądarkach wyposażonych w silnik Chromium jak i Gecko.

Czytelność kodu: Kod źródłowy (HTML i CSS) musi być uporządkowany, posiadać wcięcia oraz komentarze ułatwiające późniejszą ocenę, jak i edycję.

3. Opis struktury projektu

3.1. Użyte technologie

Zgodnie z przyjętymi wymaganiami, strona internetowa **Virexon** wykonana została tylko i wyłącznie za pomocą języka znaczników HTML5 oraz kaskadowych arkuszy stylów CSS3 bez żadnych zewnętrznych bibliotek oraz frameworków.

HTML5 wykorzystano do stworzenia podstawowego szkieletu witryny uwzględniając w niego znaczniki takie jak chociażby `<header>`, `<nav>`, `<section>` oraz `<footer>` co pozytywnie wpływa na pozycjonowanie (SEO) oraz sam układ strony.

CSS wykorzystano do stworzenia warstwy wizualnej opartej na nowoczesnych standardach, w tym Flexbox do układów oraz zmienne do prostego zarządzania kolorami. Użyte zostały również animacje klatkowe (`@keyframes`).

3.2. Struktura katalogów

Struktura katalogów została zaprojektowana z myślą o łatwym zarządzaniu zasobami i ewentualnym rozbudowaniu projektu. Cała struktura przedstawiona została w formie drzewka poniżej:

```
ti-project-1/
├── assets/ ..... Zasoby graficzne i ikony
│   ├── fotografie.webp ..... Używane do grafik w treści
│   ├── fotografie.avif ..... Używane do tła sekcji
│   └── favicon.svg
├── css/ ..... Arkusze stylów
│   ├── style.css ..... Główne style strony głównej
│   └── subsite.css ..... Style dla podstron
└── html/ ..... Pliki źródłowe stron
    ├── index.html ..... Strona główna
    ├── about.html
    ├── services.html
    └── jobs.html
```

3.3. Konwencje kodowania i organizacja stylów

Aby zapewnić czytelność i łatwość konwersacji kodu, w arkuszach stylów zastosowano następujące konwencje:

- **Zmienne CSS (:root):** Kolory oraz fonty zostały zdefiniowane w pseudoklasie `:root`. Umożliwia to szybką zmianę motywu kolorystycznego w jednym miejscu, bez konieczności edycji całego pliku (np. `-text-color-accent: #C1121F`).
- **Skalowalne jednostki REM:** W projekcie zastosowano technikę ustawienia bazowej wielkości czcionki dla elementu `html` na 50%. Dzięki temu 1rem odpowiada 8px (domyślnie przeglądarka ma 16px, 50% z 16px to 8px).
- **Reset stylów:** Zastosowano globalny reset (`margin: 0; padding: 0; box-sizing: border-box;`), aby wyeliminować różnice w domyślnym stylowaniu przez różne przeglądarki.

3.4. Implementacja efektów wizualnych i optymalizacja

W projekcie duży nacisk położono na płynność interfejsu i nowoczesny wygląd.

- **Optymalizacja renderowania:** W sekcji Hero oraz przy obrazkach wykorzystano właściwość `transform: translateZ(0)` oraz `will-change`. Wymusza to na przeglądarce skorzystanie z akceleracji sprzętowej (GPU), co zapobiega "migotaniu" i spadkom klatek podczas animacji.
- **Animacje wejścia:** Zastosowano animację `@keyframes jumpIn` dla nagłówków, która łączy zmianę przezroczystości z przesunięciem w osi Y, tworząc efekt dynamicznego pojawiania się treści.
- **Interaktywne elementy:** Przyciski, logo oraz sekcje z obrazkami posiadają zdefiniowane stany `:hover` z płynnymi przejściami (`transition`), np. efekt skali (`scale`) czy zmiana filtru szarości (`grayscale`) na kolor.

3.5. Mechanizm Scroll Snap (Podstrony)

Dla podstron zaimplementowano efekt przyciągania przewijania, który dzieli treść na pełnoekranowe sekcje.

- Wykorzystano właściwości `scroll-snap-type: y mandatory` dla kontenera oraz `scroll-snap-align: start` dla sekcji. Dzięki temu użytkownik podczas przewijania automatycznie "dokuje" do kolejnego logicznego bloku treści, co buduje narracyjny charakter strony.
- Każda sekcja posiada wysokość `100vh` (viewport height), gwarantując wypełnienie całego ekranu.

3.6. Responsywność (RWD)

Strona jest w pełni responsywna i dostosowuje się do urządzeń mobilnych oraz tabletów. Wykorzystano podejście priorytetyzujące urządzenia desktopowe z punktami przerwania dla szerokości:

- 1440px
- 1200px
- 1024px/900px
- 600px

Dodatkowe zmiany w układzie mobilnym:

- **Zmiana układu Flexbox:** Wiersze (`.feature-row`, `.child-container`) zmieniają kierunek z poziomego na pionowy (`flex-direction: column`), a elementy układane są jeden pod drugim.
- **Dezaktywacja Scroll Snap:** Na urządzeniach dotykowych o mniejszej rozdzielczości (poniżej 1024px) mechanizm `scroll-snap` jest wyłączany (`scroll-snap-type: none`), a wysokość sekcji zmienia się z `100vh` na `auto`. Zapobiega to blokowaniu się przewijania na telefonach i poprawia UX.
- **Dostosowanie typografii:** Wielkości czcionek (np. nagłówków Hero) są redukowane, aby mieściły się na węższych ekranach bez przełamывania słów.
- **Nawigacja:** Pasek nawigacji na telefonach zmienia układ na kolumnowy lub "wrap", aby pomieścić wszystkie linki w obszarze dotykowym.

4. Harmonogram realizacji projektu

4.1. Etapy realizacji projektu

Proces tworzenia witryny był ciągiem mniejszych bądź większych etapów takich jak:

1. Analiza wymagań i ułożenie wstępnego schematu.
2. Wstępne stworzenie układu strony.
3. Implementacja rozwiązania przy użyciu technologii HTML5
4. Wprowadzenie kaskadowych arkuszy stylów.
5. Przeprowadzenie ponownego ułożenia witryny na nowo - reset.
6. Ukończenie strony głównej.
7. Powtórzenie kroków do pozostałych podstron zaczynając od ułożenia schematu oraz układu kończąc na implementacji CSS w celu dodania animacji, wejścia układu strony w życie i manipulacji treścią.
8. Zakończenie prac nad podstronami.
9. Przerobienie strony głównej.
10. Drobne poprawki.
11. Stworzenie dokumentacji technicznej.

4.2. Wykres Gantta

4.3. System kontroli wersji

- Podczas tworzenia projektu wykorzystano system kontroli **Git** poprzez aplikację **GitHub Desktop**. Do przechowywania kodu źródłowego wykorzystano repozytorium na platformie **GitHub**. Repozytorium dostępne jest pod adresem:

<https://github.com/Avsh11/web-technologies/tree/main/ti-project-1>

- Repozytorium zawiera pełną wersję projektu po ostatecznych poprawkach (w chwili oddania) i dostępne będzie w trybie publicznym ustalonym przez wykładowcę prowadzącego przedmiot.

5. Prezentacja warstwy użytkowej projektu

6. Instrukcja uruchomieniowa

Aby poprawnie uruchomić witrynę w celach testowych należy wykonać następujące kroki uprzednio przygotowując środowisko uruchomieniowe:

Wymagane środowisko

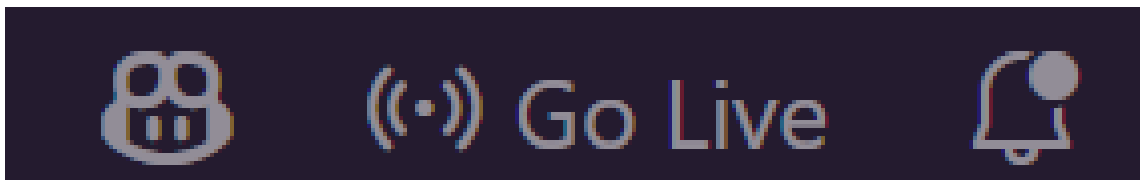
- **Proponowany system operacyjny:** Microsoft Windows / Linux / Apple MacOS
- **IDE:** Visual Studio Code
- **Rozszerzenia:** Live Server autorstwa: **Ritwick’a Dey’a**
- **Repozytorium GitHub:** <https://github.com/Avsh11/web-technologies>
- **Proponowany system kontroli wersji:** GitHub Desktop App

Krok 1. - Konfiguracja środowiska testowego

1. Pobierz edytor Visual Studio Code.
2. Pobierz aplikację GitHub Desktop.
3. Uruchom aplikację GitHub Desktop i wybierz opcję sklonowania repozytorium w zakładce **File**.
4. Sklonowane repozytorium umieszczone ma zostać w wybranym stworzonym przez użytkownika katalogu.

Krok 2. - Uruchomienie strony

1. Otwórz uprzednio skonfigurowaną aplikację Visual Studio Code.
2. Upewnij się, że poprawnie zainstalowałeś rozszerzenie Live Server.
3. Otwórz katalog, w którym sklonowałeś repozytorium.
4. Wejdź w plik `index.html`.
5. W prawym dolnym rogu **IDE** kliknij w przycisk Go Live jak na obrazku poniżej - rys. 6.1



Rys. 6.1. Przycisk Go Live rozszerzenia Live Server

7. Testowanie poprawności i stopnia optymalizacji kodu

W celu potwierdzenia iż spełnione zostały wymagania niefunkcjonalne, zostały przeprowadzone testy, które mają na celu udowodnić, że strona wraz z jej podstronami spełniają chociażby standardy **W3C**. Został też przeprowadzony test **Lighthouse** dostępny w narzędziach deweloperskich w przeglądarkach opartych na silniku **Chromium** autorstwa korporacji **Google**.

7.1. Walidacja HTML oraz CSS - W3C

Zgodnie z przyjętymi standardami kluczowym elementem weryfikacji i jakości kodu jest poddanie go testowi udostępnionemu przez **World Wide Web Consortium (W3C)**. Walidacja potwierdzać będzie poprawność semantyczną oraz składniową kodu.

7.1.1. HTML

Kod źródłowy pliku `index.html` sprawdzony został za pomocą **W3C Markup Validation Service**. Jego wynik przedstawiono na rysunku poniżej - rys. 7.1

The screenshot shows the Nu Html Checker interface. At the top, there's a blue header with the text "Nu Html Checker". Below it, a message states: "This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change". The main section is titled "Showing results for uploaded file index.html (checked with vnu 25.12.12)". Under "Checker Input", there are options to "Show" (source, outline, image report, Options...) and a "Check by" dropdown set to "file upload" with a "Choose File" button and "No file chosen" text. A note says "Uploaded files with .xhtml or .xht extensions are parsed using the XML parser." and a "Check" button is present. A green banner at the bottom of the interface reads "Document checking completed. No errors or warnings to show." Below this, it says "Used the HTML parser." and "Total execution time 4 milliseconds."

Rys. 7.1. Wynik walidacji kodu HTML5 dla pliku `index.html`

Kod źródłowy pliku `about.html` sprawdzony został za pomocą **W3C Markup Validation Service**. Jego wynik przedstawiono na rysunku poniżej - rys. 7.2

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for uploaded file about.html (checked with vnu 25.12.12)

Checker Input

Show ☐ source ☐ outline ☐ image report

Check by No file chosen

Uploaded files with .xhtml or .xht extensions are parsed using the XML parser.

Document checking completed. No errors or warnings to show.

Used the HTML parser.

Total execution time 4 milliseconds.

Rys. 7.2. Wynik walidacji kodu HTML5 dla pliku about.html

7.1.2. CSS

Kod źródłowy pliku style.css sprawdzony został za pomocą **W3C Markup Validation Service**. Jego wynik przedstawiono na rysunku poniżej - rys. 7.3

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area (checked with vnu 25.12.12)

Checker Input

Show ☒ source ☐ outline ☐ image report

Check by ☒ CSS

```
@import url('https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..900&display=swap');
@import url('https://fonts.googleapis.com/css2?
family=kanit:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
/* Zmienne zeby nie powtarzac na pale kodu + szybsza zmiana palety kolorow np */
:root {
  --font-primary: "Kanit", sans-serif;
  --font-secondary: "Montserrat", sans-serif;
  /* Kolory z palet z coolors, nazwy palet w css podstron */
  --text-color-main: #2F4F3;
  --text-color-accent: #C11121F;
  --color-main: #0A0908;
  --color-border-dim: #222222;
  --color-border-light: #444444;
  --text-color-mute: #888888;
}
```

Document checking completed. No errors or warnings to show.

Source

```
1. @import url('https://fonts.googleapis.com/css2?family=Montserrat:ital,wght@0,100..900;1,100..900&display=swap');
2. @import url('https://fonts.googleapis.com/css2?
family=kanit:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
3. /* Zmienne zeby nie powtarzac na pale kodu + szybsza zmiana palety kolorow np */
4. :root {
5.   --font-primary: "Kanit", sans-serif;
6.   --font-secondary: "Montserrat", sans-serif;
```

Rys. 7.3. Wynik walidacji kodu CSS dla pliku style.css

Kod źródłowy pliku subsite.css sprawdzony został za pomocą **W3C Markup Validation Service**. Jego wynik przedstawiono na rysunku poniżej - rys. 7.4

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area (checked with vnu 25.12.12)

Checker Input

Show ☒ source ☐ outline ☐ image report

Check by ☒ CSS

```
/* Z Google fonts wziete */
@import url('https://fonts.googleapis.com/css?family=Montserrat:ital,wght@0,100..900;1,100..900&display=swap');
@import url('https://fonts.googleapis.com/css?family=Kanit:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
/* Zmienne niektore zeby nie pisac tego samego w niektorych miejscach + paleta kolorow od razu do strony. */
:root {
  /* Czcionki - Google Fonts */
  --font-primary: "kanit", sans-serif;
  --font-secondary: "Montserrat", sans-serif;
  /* Paleta coolors do tekstow: Gothic Romance, Fiery Ocean */
  --text-color-main: #F2F4F3;
  --text-color-accent: #C1121F;
  /* Gothic Romance */
  --color-main: #0A0908;
}
```

Document checking completed. No errors or warnings to show.

Source

```
1. /* Z Google fonts wziete */
2. @import url('https://fonts.googleapis.com/css?family=Montserrat:ital,wght@0,100..900;1,100..900&display=swap');
3. @import url('https://fonts.googleapis.com/css?family=Kanit:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900;1,100;1,200;1,300;1,400;1,500;1,600;1,700;1,800;1,900&display=swap');
4. /* Zmienne niektore zeby nie pisac tego samego w niektorych miejscach + paleta kolorow od razu do strony. */
5. :root {
```

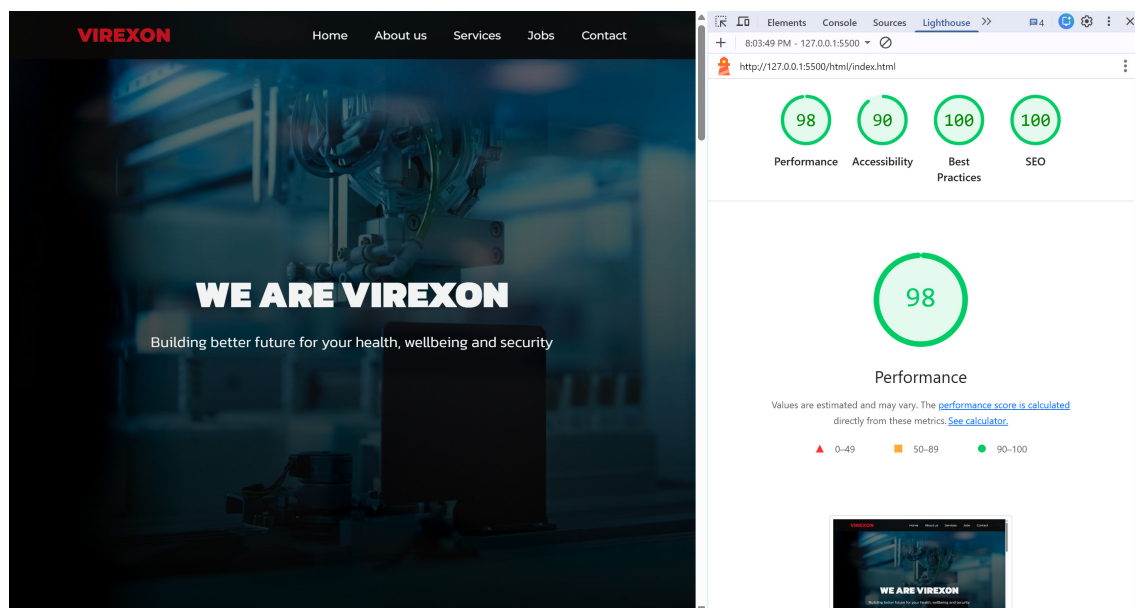
Rys. 7.4. Wynik walidacji kodu CSS dla pliku `subsite.css`

7.2. Audyt wydajności - Lighthouse

Po podstawowej walidacji wykonanej dzięki narzędziu **World Wide Web Consortium (W3C)** dodatkowo wykonano również audyt wydajności w narzędziu deweloperskim **Lighthouse** autorstwa korporacji **Google**. Wyniki testów zostaną przedstawione w poniższych podsekcjach.

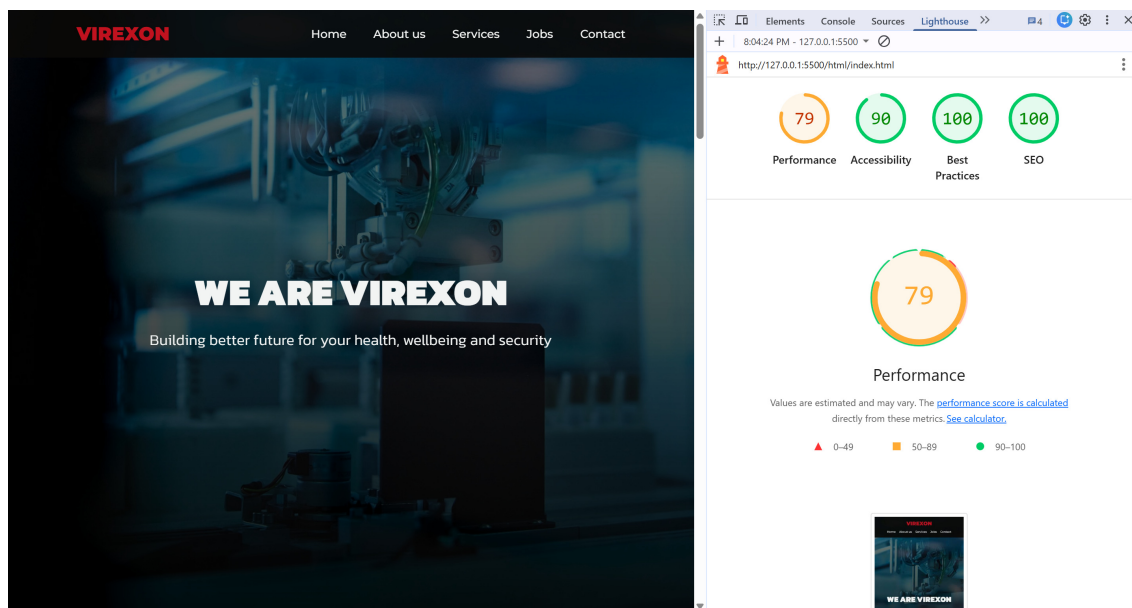
7.2.1. Strona główna

Strona główna testowana w trybie **Desktop** - rys. 7.5



Rys. 7.5. Wynik testu strony głównej - tryb Desktop

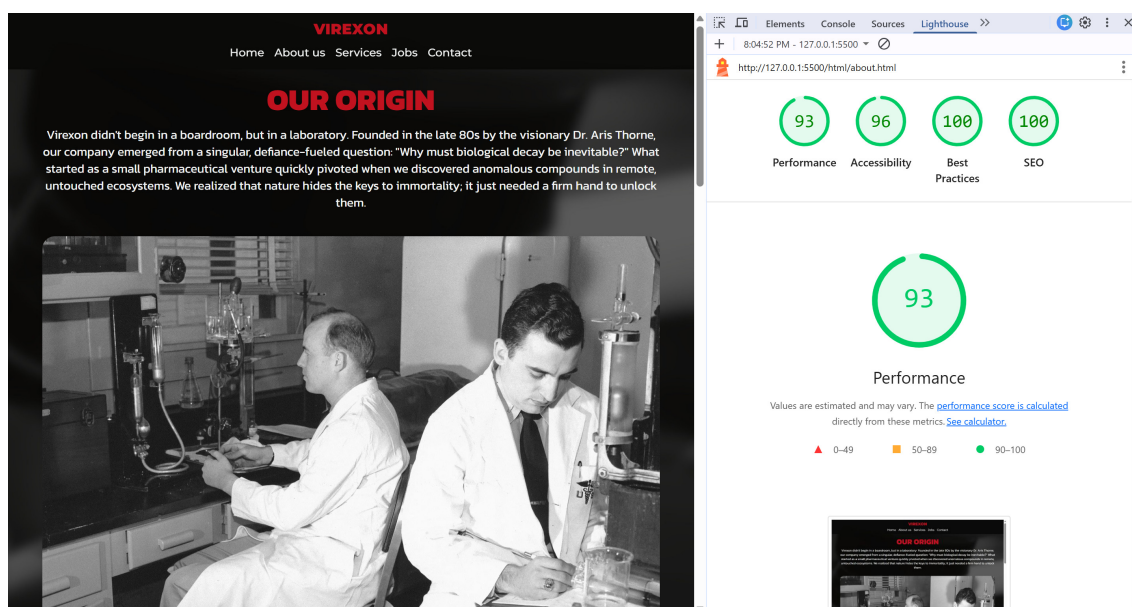
Strona główna testowana w trybie **Mobile** - rys. 7.6



Rys. 7.6. Wynik testu strony głównej - tryb Mobile

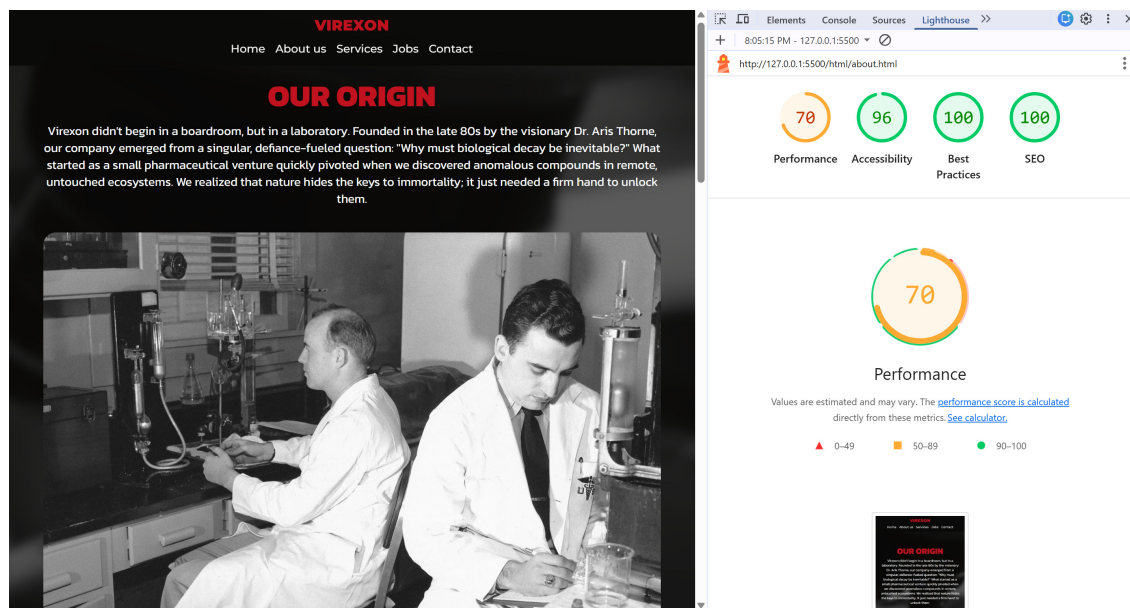
7.2.2. Podstrona

Podstrona (`about.html`) testowana w trybie Desktop - rys. 7.7



Rys. 7.7. Wynik testu podstrony - tryb Desktop

Podstrona (`about.html`) testowana w trybie Mobile - rys. 7.8



Rys. 7.8. Wynik testu podstrony - tryb Mobile

8. Podsumowanie

Celem projektu było stworzenie responsywnej i wizualnie estetycznej witryny dla korporacji oferującej wszelakie usługi w sektorze prywatnym. Umożliwić ma ona użytkownikowi łatwą nawigację przez zrozumiale rozmieszczone treści. Dzięki zastosowaniu kaskadowych arkuszy stylów i języka znaczników HTML5 udało się stworzyć witrynę spełniającą powyższe kryteria.

Mimo pomyślnego zrealizowania większości najważniejszych funkcjonalności wedle etapów rozwoju witryna wciąż posiada miejsce na wiele innych równie ważnych funkcjonalności o które można w przyszłości rozbudować ową witrynę. Oto niektóre z nich:

1. Umożliwić działanie formularza poprzez skrypty oraz implementacje języka PHP wraz z jakąś bazą danych.
2. Dodanie funkcjonalności stworzenia konta oraz logowania.
3. Dodanie większej ilości interaktywnych elementów w tym animacji.

Bibliografia

- [1] ChatGPT, 2025. <https://chatgpt.com/>.
- [2] Coolors, 2025. <https://coolors.co/>.
- [3] Freepik, 2025. <https://www.freepik.com/>.
- [4] Google Fonts, 2025. <https://fonts.google.com/>.
- [5] Pexels, 2025. <https://www.pexels.com/>.
- [6] Stack Overflow, 2025. <https://stackoverflow.com/>.
- [7] Unsplash, 2025. <https://unsplash.com/>.
- [8] W3schools, 2025. <https://www.w3schools.com/>.

Spis rysunków

6.1	Przycisk Go Live rozszerzenia Live Server	13
7.1	Wynik walidacji kodu HTML5 dla pliku index.html	14
7.2	Wynik walidacji kodu HTML5 dla pliku about.html	15
7.3	Wynik walidacji kodu CSS dla pliku style.css	15
7.4	Wynik walidacji kodu CSS dla pliku subsite.css	16
7.5	Wynik testu strony głównej - tryb Desktop	16
7.6	Wynik testu strony głównej - tryb Mobile	17
7.7	Wynik testu podstrony - tryb Desktop	17
7.8	Wynik testu podstrony - tryb Mobile	18

Załącznik nr 2 do Zarządzenia nr 228/2021 Rektora Uniwersytetu Rzeszowskiego z dnia 1 grudnia 2021 roku w sprawie ustalenia procedury antyplagiatowej w Uniwersytecie Rzeszowskim

OŚWIADCZENIE STUDENTA O SAMODZIELNOŚCI PRACY

..... Krystian Filipek
Imię (imiona) i nazwisko studenta

Wydział Nauk Ścisłych i Technicznych

..... Informatyka
Nazwa kierunku

..... 134907
Numer albumu

1. Oświadczam, że moja praca projektowa pt.: Magazyn ze sprzętem wojskowym "C.A.S"
 - 1) została przygotowana przeze mnie samodzielnie*,
 - 2) nie narusza praw autorskich w rozumieniu ustawy z dnia 4 lutego 1994 roku o prawie autorskim i prawach pokrewnych (t.j. Dz.U. z 2021 r., poz. 1062) oraz dóbr osobistych chronionych prawem cywilnym,
 - 3) nie zawiera danych i informacji, które uzyskałem/am w sposób niedozwolony,
 - 4) nie była podstawą otrzymania oceny z innego przedmiotu na uczelni wyższej ani mnie, ani innej osobie.
2. Jednocześnie ~~wyrażam zgodę~~/nie wyrażam zgody** na udostępnienie mojej pracy projektowej do celów naukowo-badawczych z poszanowaniem przepisów ustawy o prawie autorskim i prawach pokrewnych.

Rzeszów, 28.06.2025
(miejscowość, data)

Filipek Krystian
(czytelny podpis studenta)

* Uwzględniając merytoryczny wkład prowadzącego przedmiot

** – niepotrzebne skreślić