



Великотърновски университет „Св.св. Кирил и Методий“

Факултет „Математика и информатика“

Дипломна работа

**Тема: „Проектиране и разработване на уеб приложение в
училищна среда“**

Дипломант:

Александър Йорданов Георгиев

ФН: № 2009010806

Специалност: „ПОМИ“ IV курс

Научен ръководител:

гл.ас. д-р Силвия Върбанова

**Велико Търново
2024**

ЗАДАНИЕ НА ДИПЛОМНА РАБОТА

Дипломант: Александър Йорданов Георгиев

Специалност: Педагогика на обучението по математика и информатика

Курс: IV Факултетен № 2009010806

ОКС бакалавър/ магистър

Тема на дипломна работа:

„Проектиране и разработване на уеб приложение в училищна среда“

Изследователски проблеми: Изследване и анализиране на подобни приложения в уеб технологиите и уеб базираните приложения.

Цели на дипломната работа: Изграждане на уеб приложение, което да отговаря на целите и нуждите в училищна среда.

Задачи на дипломната работа:

- 1) Изграждане на уеб приложение
- 2) Вграждане на съвременни функционалности
- 3) Анализ на подобни приложения

Изискване за разработване на определени резултати: Изграждане на цялостно приложение със съвременен дизайн и полезни функционалности, както и responsive за всички устройства.

Препоръчителен обем на разработка: 50-60 страници.

Краен срок за предаване на дипломния проект: 25.08.2024

Дипломант:
(подпис)

Научен ръководител:.....
(гл. ас. д-р С. Върбанова)

Дата на възлагането:
гр. Велико Търново

ДИПЛОМНАТА РАБОТА Е ПРЕДАДЕНА В СРОК И ОТГОВАРЯ НА
ИЗИСКВАНИЯТА ЗА ДОПУСКАНЕ ДО ЗАЩИТА

Дата на предаването:
гр. Велико Търново

Научен ръководител:.....
(гл.ас.д-рС. Върбанова)

Съдържание

Увод	4
Глава 1: Съвременни уеб технологии за изграждането на уеб приложение	6
1.2 Използвани Технологии	7
1.3 Responsive Web Design, media queries	12
1.4 Bootstrap	13
1.5 JavaScript/frameworks	14
1.6 React, Single Page Application	15
Глава 2: Анализ на подобни приложения	16
2.1 Уеб сайт на СУ "Васил Левски"	16
2.2 Уеб сайт на СУ „Йордан Йовков“	23
2.3 Извод.....	27
Глава 3: Изграждане на уеб приложение за целите на образователния процес/образование.....	29
3.1 Мотивация за създаването на приложение/уеб сайт в рамките на образователния процес	29
3.2 Анализ на кодовата част от педагогическа гледна точка	30
3.3 Преглед на приложението от учебна гледна точка	38
3.4 Бъдещо развитие	49
Заключение	49
Източници	51

Увод

Технологиите се развиват изключително бързо и много хора започнаха да използват Интернет за различни неща, било то за следене на новини, онлайн пазаруване или използването на социални мрежи. Всеки един потребител можеше да се докосне до света на интернет с помощта на настолен компютър, но в днешно време вече всички притежават лаптоп, таблет, телефон или друго смарт устройство, което е свързано със световната мрежа. Чрез тези устройства се намират много по-лесно информация, където, и когато е необходимо, защото самите те са малки и компактни за ползване - навън и в дома.

Разработването на дипломната работа е една стъпка от създаването на уеб сайт, в който да са съобразени всички тези удобства за потребителя.

Целта е да се изгради приложение, в което да се създаде добро преживяване на потребителя, когато го посещава в реална работна среда. Като влезе в разработвания проект да намира по-лесно актуалните новини, както и да дава обратна връзка.

За да се разработи правилно приложението, трябва да се мине през няколко етапа:

- Планиране на уеб сайта;
- Анализ на подобни приложения;
- Изграждане на уеб приложение за целите на образователния процес/образование;
- Бъдещо развитие;

Дипломната работа се състои от три глави, като се базира на етапите, които са посочени отгоре.

- Глава 1 – „Съвременни уеб технологии за изграждането на уеб приложение“;
- Глава 2 – „Анализ на подобни приложения“;
- Глава 3 – „Изграждане на уеб приложение за целите на образователния процес/образование“;

В първа глава се разглеждат няколко основни въпроса като: „Какво е уеб сайт, стилизирането на уеб сайт, хостване?“, „Как работи сайтът, спрямо браузъра?“, „Статични уеб сайтове, динамични уеб сайтове“. След това се разглеждат различните технологии, чрез които се разработва един уеб сайт, като се споменават най-важните му приложения. Описани са какви са предимствата на самите технологии и с какво те подобряват уеб сайта.

Във втора глава се сравняват няколко работещи приложения, като се описват визуално, функционално, разглеждат се предимствата, недостатъците, приликите и разликите и накрая се прави извод, в който се описва какво трябва да представлява един съвременен уеб сайт.

В трета глава се описва процесът на разработка на самия сайт, като в едната част има и потребителската гледна точка върху разработения проект.

Глава 1: Съвременни уеб технологии за изграждането на уеб приложение

1.1 Видове Уеб сайтове

Уеб сайтът може да бъде описан като съвкупност от свързани помежду си страници, достъпни чрез общ **URL адрес**. Всяка страница се свързва с началната директория на сайта чрез домейн, започващ със символа „/“. Също така, уеб сайтът се разглежда като платформа за предоставяне на информация или различни услуги, насочени към потребителите.

Структурата и дизайнът на уеб сайта следва да бъдат лесни за използване и разбираеми за потребителите. Важно е да е налично добре организирано меню, което да улеснява навигацията и използването на приложението, като по този начин се подобрява потребителското изживяване.

Процесът на качване на уеб сайт включва свързване със сървър, което му позволява да бъде достъпен чрез браузър. Тази процедура осигурява непрекъсната достъпност до уеб сайта за потребителите, което е ключов аспект от неговото функциониране.

Когато потребителят влиза в уеб приложението, браузърът изпраща заявка към сървъра, на който е хостван сайтът. В отговор, сървърът предоставя необходимите данни, които включват **HTML** за структурата на страницата, **CSS** за нейното стилизиране и **JavaScript (JS)** за функционалността. Браузърът обработва тези данни и ги визуализира, така че потребителят да вижда само крайния резултат, представен като текст, изображения, бутони и други елементи на уеб страницата.

Статичните уеб сайтове - със своята ефективност са идеални за представяне на проста информация, където съдържанието остава относително постоянно и няма нужда от сложни функции или взаимодействия.

Динамичните уеб сайтове - за разлика от статичните, позволяват на потребителите да взаимодействат с уеб сайта по много по-интуитивен и персонализиран начин. Информацията може лесно да се актуализира в реално време, без да е необходимо цялото приложение да бъде презаредено, което води до значително подобрена потребителска ангажираност и удовлетворение. Динамичните уеб сайтове често включват функции като формуляри за контакт, системи за резервация, инструменти за търсене и социални медийни интеграции. Тези функции не са възможни при статичните уеб сайтове, тъй като изискват обработка на въведени от потребителя данни в реално време и динамична визуализация на резултатите.

1.2 Използвани Технологии

За изграждането на един уеб сайт се използват основните технологии: **HMTL** и **CSS**. В началото, когато уеб се появява, **HTML** и **CSS** са били единственият начин, по който може да се създаде сайт, но с времето тези технологии се подобряват до **HTML5** и **CSS3**.

➤ HTML5:

В света на уеб разработката **HTML (Hyper Text Markup Language)** функционира като основен градивен елемент, използвайки набор от **HTML** тагове за формиране на структурата на даден уеб сайт.

Структурата на кода започва с **<html>** тага, който обгражда цялото съдържание на една уеб страница. В него се съдържат други основни тагове за структуриране на описателната и изпълнимата част.

Описателната част се задава с първия таг **<head>**. Той се използва за добавянето на мета данни в уеб сайта, като потребителят не ги вижда директно, а всъщност са много важни, за да може приложението да функционира.

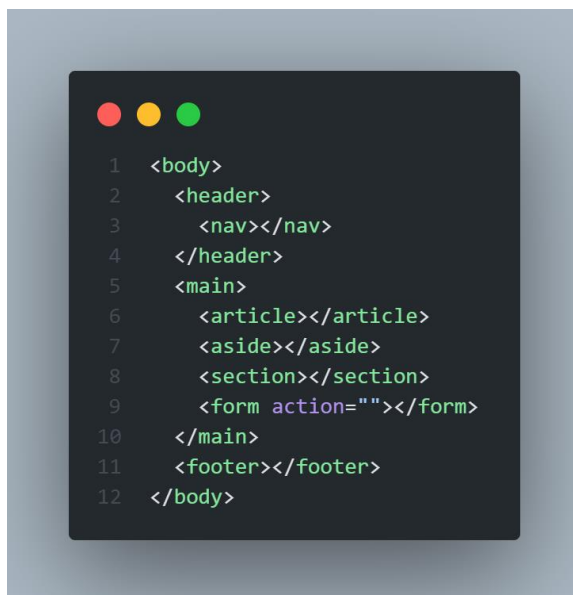
Изпълнимата част се задава с тага **<body>**. С **body** се създава визуалната част на страницата. Иползват се различни тагове като: **<h1>** за заглавие, **<p>** за създаване на параграф, **<a>** за хипервръзка, като тук се използва атрибут **href** за определянето на **URL** към хипервръзката и ****, с който създаваме изображения.

A screenshot of a code editor window with a dark background and light-colored text. The code is HTML markup for the body of a document. It includes a header, a paragraph, a link, and an image placeholder. The code is as follows:

```
1 <body>
2   <h1>Header1</h1>
3   <p>Paragraph</p>
4   <a href="#">Link</a>
5   
6 </body>
```

Фигура 1 Пример за html елементи

В **HTML5** имаме и нови семантични тагове, с които лесно се ориентираме в страницата. Примерно: **<header>**, **<nav>**, **<main>**, **<article>** **<aside>**, **<section>**, **<footer>**. В някои уеб сайтове имаме и така наречения **<form>**, който служи за аутентикация на потребителя, но може да се използва и като обратна връзка от потребителя към собственика на уеб сайта.

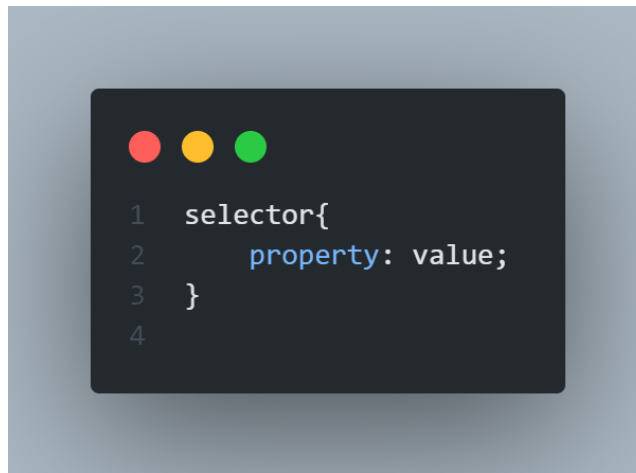
A code editor window with a dark background and light green text. It displays a snippet of HTML5 code using semantic tags. The code is numbered from 1 to 12 on the left side. The tags used are <body>, <header>, <nav>, </header>, <main>, <article>, </article>, <aside>, </aside>, <section>, </section>, <form>, </main>, <footer>, and </body>. The <form> tag has an attribute action="" highlighted in purple.

```
1 <body>
2   <header>
3     <nav></nav>
4   </header>
5   <main>
6     <article></article>
7     <aside></aside>
8     <section></section>
9     <form action=""></form>
10  </main>
11  <footer></footer>
12 </body>
```

Фигура 2 Пример за семантични тагове

➤ **CSS3:**

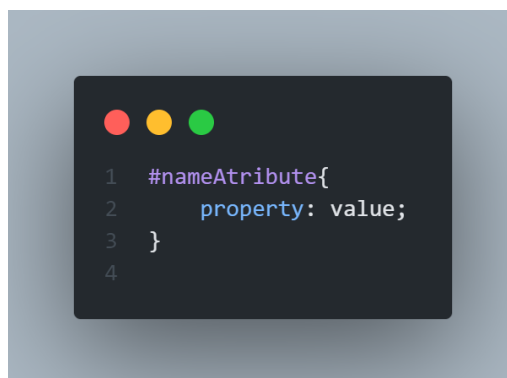
CSS (Cascading Style Sheets) се използва за стилизирането на уеб страницата с помощта на различни свойства. Със **CSS**, можем да променяме цветове, да стилизираме текст, да определяме разположението на елементите в сайта и др.



Фигура 3 Селектор в CSS

В **CSS** имаме:

- **Селектор (selector)**, който дефинира елемента, върху който ще се приложи стила. Избирането на подходящия селектор става по един от следните начини:
 - **Елементи**: 'p', 'div', 'a'...;
 - **id (идентификатор) или class(клас)**;



Фигура 4 Пример за синтаксис на „id“



Фигура 5 Пример за синтаксис на „class“

Разликата между „id“ и „class“, е че „id“ е уникален и се използва за стилизиране на един елемент, докато „class“ може да бъде приложен повече от веднъж и се използва за групиране на елементи с еднакви **CSS** стойности.

➤ **Свойствата (property) в CSS** - функционират като основни инструменти, като определят външния вид и начина на визуализация на елементите в уеб страницата. Всяко едно от тях определя различни свойства на елемента. Примери могат да бъдат:

- **Background** – с това свойство се определя какъв цвят или изображение ще е фона на страницата.;
- **Color** – с това свойство се определя цветът на текста в страницата.;
- **Margin** и **padding** – с **margin**, се определя външното отстояние на елементите, а с **padding** се определя вътрешното отстояние от елементите.;

➤ **Стойност (value)**, която определя специфична стойност на свойството в **CSS**. Ето няколко примера:

- **Background** и **color** имат стойности като: имена на цветове (**red, green, blue**), шестнадесетични кодове

започващи с „#“, които отговарят на различни нюанси на основните цветове. При определени свойства на **background** има различни стойности. Примерно при **background-image** стойността е **URL** на изображение, като **URL**-а може да е за **URL** от друг сайт в интернет и **URL** от локалните файлове на сайта.;

- **Margin** и **Padding** – стойностите за тези две свойства могат да бъдат: **%**, **px**, **em**, **rem**.;

1.3 Responsive Web Design, media queries

Появата на мобилните устройства създава различните размери на екраните и от съществено значение е уеб страниците и приложенията да бъдат разработени с помощта на подход за гъвкав дизайн, който отговаря на специфичните изисквания на всяко устройство. Тази трудност се преодолява чрез **responsive design**.

Responsive design представлява метод, с който може да се създаде дизайн на уеб сайт. Той се променя в зависимост от размера на екрана, като остава стилизиран и подреден. Методът помага уеб сайтът да постигне по-голяма използваемост като достига до повече потребители със своя гъвкав дизайн за всяко едно устройство като телефон, таблет, компютър...

Media Query е свойство в **CSS**, като се стилизират определени елементи в страницата по индивидуален начин. Свойството е основна част от **Responsive design** и се използва за промяна на дизайна на уеб сайта спрямо множество устройства.

Например при определена разделителна способност на екрана всеки елемент от страницата се оформя по специфичен начин.

Media query се вгражда в **CSS** чрез **@media**.



Фигура 6 Пример за оформление на екраните

1.4 Bootstrap

Bootstrap е, **open source framework**, разработен от **Twitter**. В тази библиотека има готови компоненти, които са лесно приложими в проект. **Bootstrap** най-вече се използва от **front-end** програмисти, като с минимален **HTML** и **CSS** създават красиви и гъвкави елементи. Библиотеката има готови форми, навигационни менюта, бутони и много разнообразие от различни стилизации. Програмистите се възползват и от опцията за готов метод на **responsive design**. Само с поставяне на специфични класове върху съответните елементи, включват готови **JavaScript** компоненти.



Фигура 7 Bootstrap response

1.5 JavaScript/frameworks

JavaScript е от високо ниво програмен език, който се използва главно в web програмирането. Скриптовият език се прилага за валидация на форми и създаване на анимации. Усвояването му позволява манипулиране на елементите в **DOM** дървото, създаване на променливи функции (кодова част, която може да се изпълни при необходимост), лесна обработка на масиви-обекти с помощта на методи (функции, с които се манипулират данни). С програмния език може да се създават **събития (event)**, с които да се създадат динамични визуализации. **JavaScript** има и различни **библиотеки** и **framework**. Пример за библиотека е **React**, с която може да се

създават съвременни уеб приложения. **React** използва компоненти, с които може да има лесна и гъвкава поддръжка, с лесната актуализация на **DOM**.

Framework е съвкупност от инструменти и технологии използвани за създаване на уеб сайтове и уеб приложения. Използването на **web framework** улеснява разработването на проекта със своите готови компоненти, стилове и функционалности. Пример за **framework** може да е **React**.

1.6 React, Single Page Application

React е библиотека, която е разработена от **Facebook** и в последствие е опакована и достъпна за програмистите. Библиотеката се използва за подобряването на приложението като бързо актуализира и рендерира компонентите на виртуалния **DOM**, съпоставящ се с реалния **DOM**. **React** най-вече се използва за **SPA(single page application)**, където още при стартирането на приложението се зареждат всички компоненти, и рендерирането позволява приложението да работи по-ефективно като нямаме забавяне.

Single Page Application (SPA) е приложение в web, което дава възможност при зареждане апликацията да визуализира всички компоненти, и с помощта на **JavaScript** да се зареждат по специфичен ред. Предимството на **SPA** е бързото зареждане на компоненти и гладката и бърза работа в самия уеб сайт.

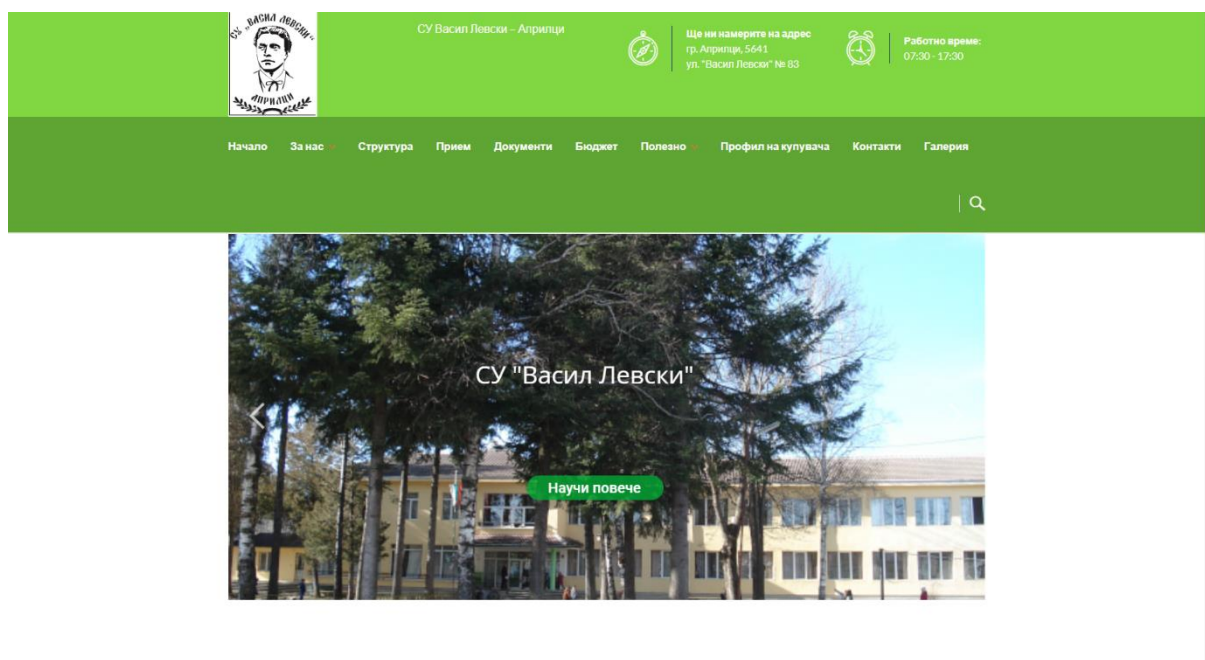
Глава 2: Анализ на подобни приложения

2.1 Уеб сайт на СУ "Васил Левски"

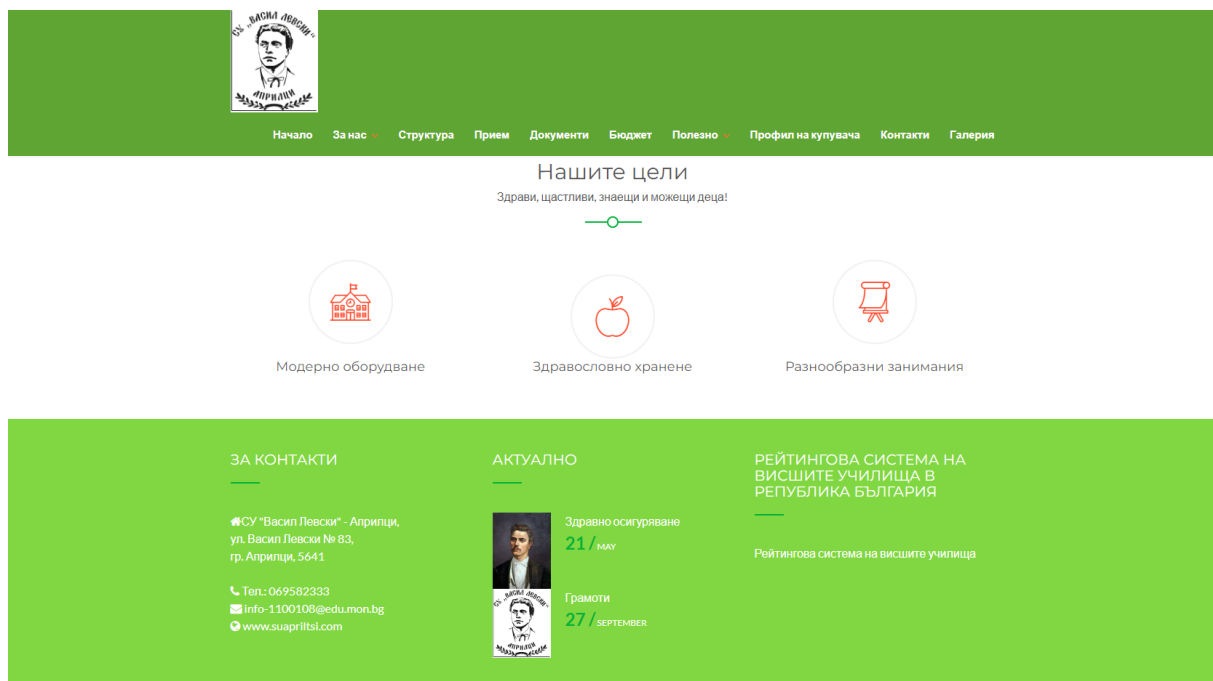
Преглед на десктоп версия на <https://suapriltsi.com/>

➤ Начална страница:

Началната страница на Средно училище "Васил Левски" е добре организирана и предоставя лесен достъп до ключови секции като: **новини**, **информация** за училището и **контакти**. Менюто е ясно и съдържа връзки към всички основни раздели. Дизайнът е изчистен, но могат да се добавят динамични изображения или интерактивни банери, които биха доусъвършенствали страницата.



Фигура 8 Начална страница 1 част



Фигура 9 Начална страница 2 част

➤ About Us:

Страницата "За нас" съдържа информация за историята на училището, неговата мисия и визия, както и данни за патрона му. Текстовото съдържание е добре структурирано и ясно, но може да се подобри, като се добавят снимки на преподавателите. Включването на кратко видео, представящо училището ще предостави по-богато изживяване за посетителите.

История

[Home](#) > История

СУ „Васил Левски“ – вчера, днес и утре

Дълъг е житейският път на училището в Априлци. Вече повече от 200 години то е непресъхващ извор на знания, култура и родолюбие. Има славна история и с чест носи името на своя патрон Васил Левски.

Училището отваря врати през 1812 г. В края на Руско-турската война в Ново село идват двама руски монаси – Геронти и Доротей. Под тяхно ръководство е построена първата училищна сграда – една голяма стая, в която учат 40-50 ученици, момчета.

Това е едно от най-старите килийни училища. В него се изучават наутилциата, псалтирът, а редом с тях и четирите аритметични действия.

През 1850г. новоселци построяват голямо за времето си училище, намиращо се редом с построената преди това църква. Училището вече е светско и в тази сграда има възможност да се учат 150 деца.

През периода 1875-1873г. в Ново село учителстват будни и прогресивни дейци на просветната мисъл, които дават своя принос не само на образователно ниво, но те са хората, запалили родолюбивата искра.

Когато постъпва като учител Никола Дабев, учениците вече са 300, а образователната институция е на нов етап. Въвеждат се нови предмети – катахизис, пеене, евангелие, а духът на учители и ученици е устремен към свободата.

Училището е огнище на светлина за новоселци. Затова след Освобождението, през 1883г., е построена нова сграда, а след тях още две- през 1912 и 1943г. Всяка от тях има своя история и е пазена в свидетелите спомени на учителите в нея.

За високи резултати в УВП през 1960 г. училището е награждено с орден „Кирил и Методий“- Първа степен. Затова МНП отпуска 20 000 лева за започване на строежа на нова сграда.

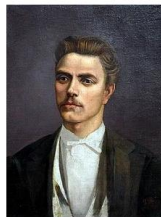
Днешното светло и просторно училище води началото си от 1966 г. и е построено, като негов директор е Пенка Мирчева- амбициозна и борбена жена.

Много реформи търпи образованието, но стъпило на здрава основа, училището в Априлци успява на всички бури.

Фигура 10 Страница на „История“

Патрон

[Home](#) > Патрон



Васил Иванов Кунчев, известен като Васил Левски,
е български национален герой.

Той е идеолог и организатор на българската национална революция, основател на Бътрешната революционна организация (БРО). Известен е и като Апостола на свободата, заради организирането и разработването на революционна мрежа за освобождаване на България от османско владичество. Пътува по страната и създава частни революционни комитети, които да подготвят обща революция. Неговата мечта е чиста и свята република, в която всички да имат равни права, независимо от своята народност и вероизповедание.

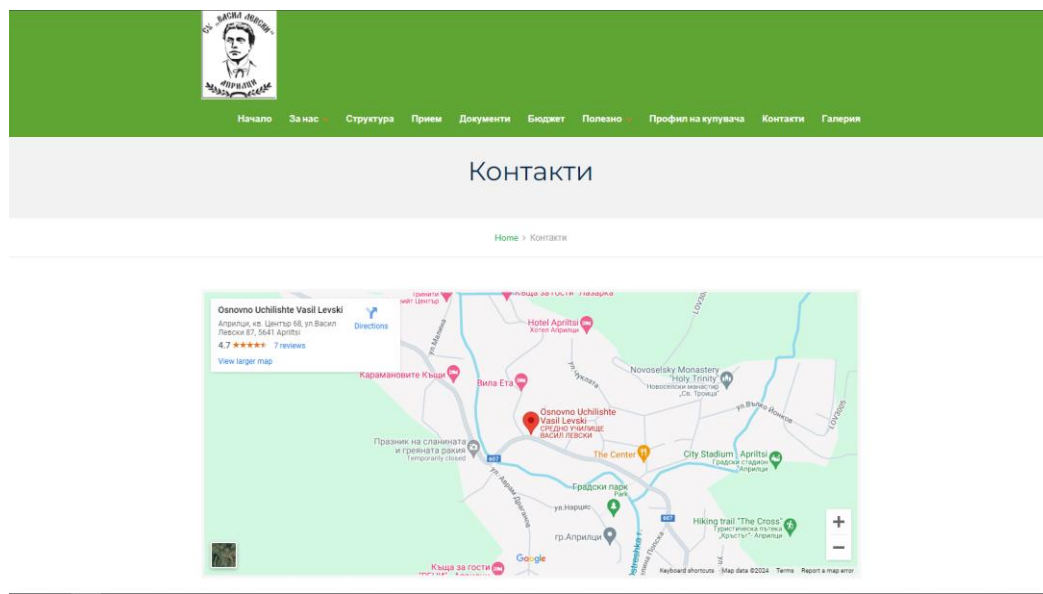
Васил Иванов Кунчев е роден на 18 юли 1837 г. в Карлово, в семейството на Иван Кунчев Иванов и Гина Василева Караиванова. Родителите му имат пет деца – Христо, Васил, Петър, Яна и Марийка.

Васил Левски първоначално учи една година в килийното училище в Карлово (1845). Продължава учението си във взаимно училище в Карлово (1846 – 1849) и същевременно учи занаята кафтанджийство при Стоян Грамът. През 1851 г. след дълго боледуване умира баща му, както и по-

Фигура 11 Страница „Патрон“

➤ Contact Page:

Страницата за контакти съдържа основните данни за връзка с училището: адрес, телефонен номер, имейл и карта с местоположение. Може да се добавят връзки към социалните мрежи или форма за обратна връзка. Това би подобрило комуникацията от потребителя.



Фигура 12 Страница „Контакти“ 1 част

Три имена*

Email *

Тема

Съобщение *

ИЗПРАТИ

Адрес
гр. Априлци,
опл. помеч,
ул. "Васил Левски" №83.

Телефон и email
Директор: Йонета Колевска
Заведващ административна
служба:
Тел: 06958 2333
info-1100108@bedumon.bg
suxlevski@gmail.com

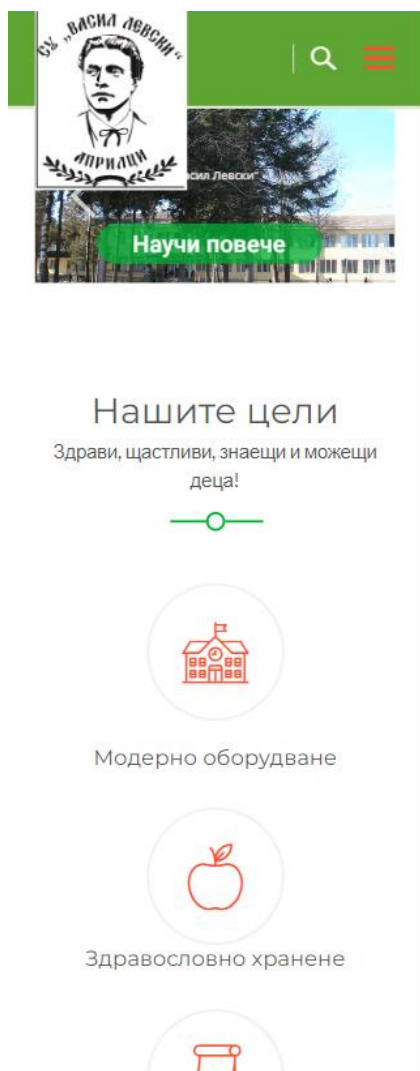
Приемно време -
Директор
Сряда,
08:30 - 10:30

Фигура 13 Страница „Контакти“ 2 част

Преглед на мобилната версия:

➤ Начална страница:

В началната страницата се забелязва как логото на сайта не е добре позиционирано, менюто придобива стил хамбургер, всички елементи отиват едно под друго. За да се подобри, трябва да се премести логото, така че да не закрива други части от страницата. При отварянето на менюто се закрива търсачката и част от линковете.



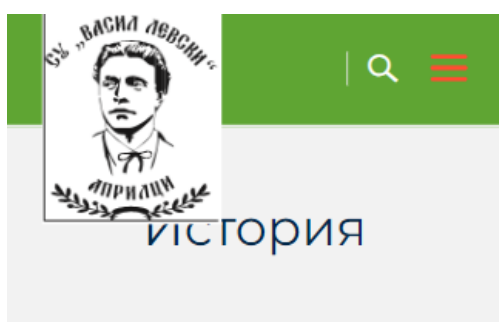
Фигура 14 Начална страница част 1



Фигура 15 Начална страница част 2

➤ История, Патрон:

В страниците **История** и **Патрон**, текстът е добре структуриран, като шрифтът е оразмерен и подравнен за по-приятна четимост от потребителя.



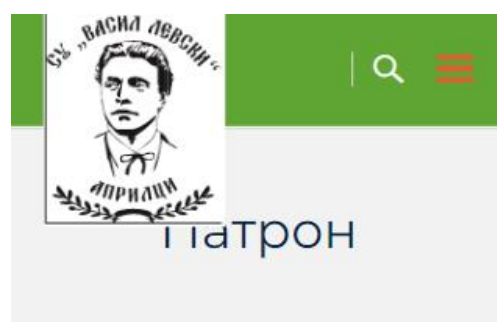
СУ „Васил Левски“ – вчера , днес и утре”

Дълъг е житейският път на училището в Априлци . Вече повече от 200 години то е непресъхващ извор на знания, култура и родолюбие. Има славна история и с чест носи името на своя патрон Васил Левски .

Училището отваря врати през 1812 г. В края на Руско-турската война в Ново село идват двама руски монаси – Геронти и Доротей. Под тяхно ръководство е построена първата училищна сграда – една голяма стая , в която учат 40-50 ученици , момчета.

Това е едно от най-старите килийни училища . В него се изучават наустилищата , псалтирът , а редом с тях и четирите аритметични действия.

Фигура 16 Мобилна страница „Патрон“



Васил Иванов Кунчев, известен като
Васил Левски,

е български национален герой.

Той е идеолог и организатор на българската национална революция, основател на Вътрешната революционна организация (ВРО).

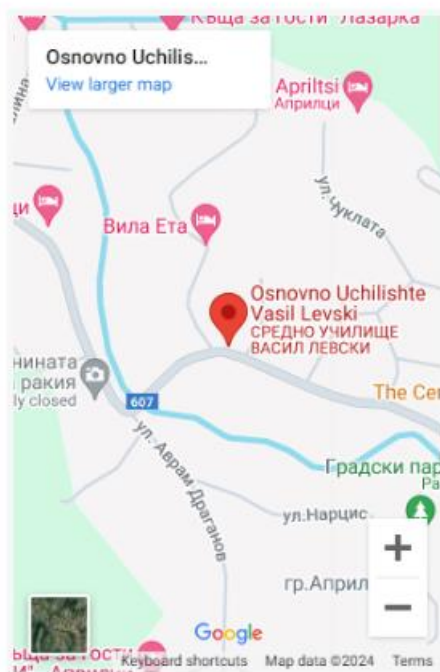
Фигура 17 Мобилна страница „История“

➤ **Contact Page:**

В страница **Контакти** се визуализира с красив и подреден дизайн, като всеки един елемент се подрежда един под друг, полетата на формата намаляват размера си и картата се мащабира спрямо размера на екрана.



[Home](#) > Контакти



Фигура 18 Мобилна страница „Контакти“

Предимства:

- Приятни за потребителя цветове
- Лесен и разбираем дизайн;
- Възможност за мобилна версия;
- Търсачка;
- Съществуващ формуляр;

Недостатъци:

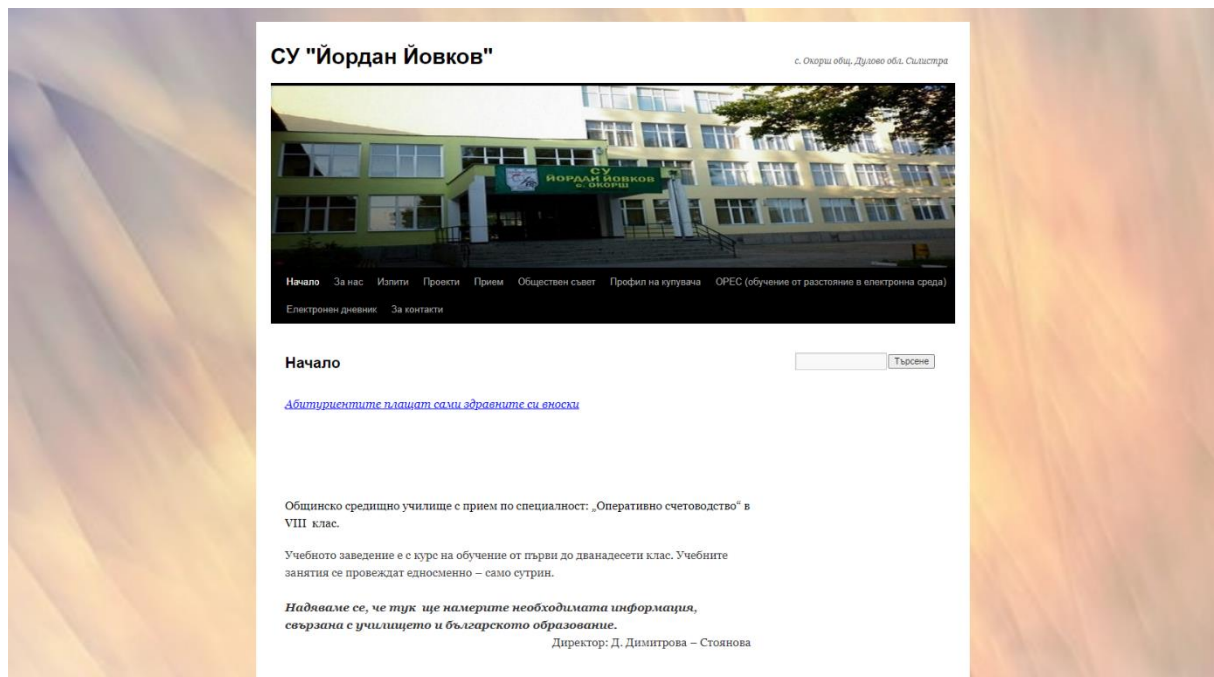
- Бавно зареждане между страниците;
- Проблем със страницата за „Галерия“;
- Недоразвит дизайн на навигационното меню;

2.2 Уеб сайт на СУ „Йордан Йовков“

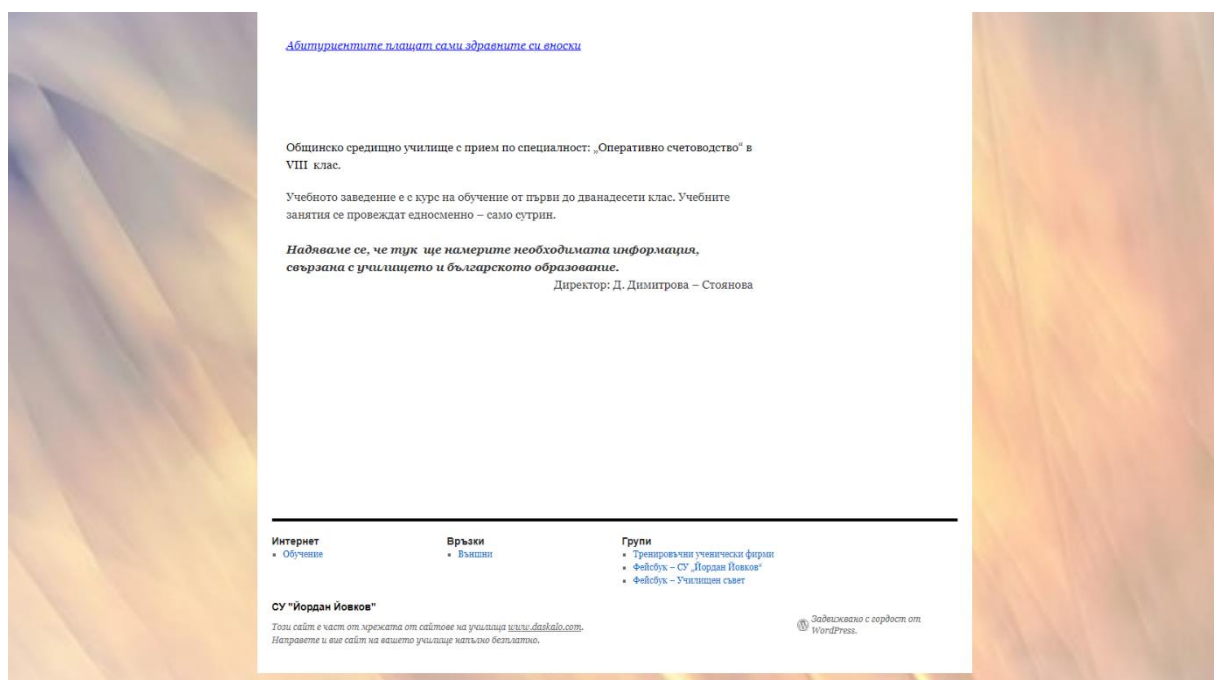
Преглед на десктоп версия на <http://suokorsh.org/>

➤ Начална страница (Home):

Началната страница на сайта предоставя основна информация за училището, включително целите и ключовите дейности. Забелязва се търсачка, която подпомага за по-лесното намиране на новини или събития в уеб сайта. Визуалният дизайн е приятен и потребителят лесно се ориентира. Уеб сайтът може да се подобри като се добавят динамични елементи.



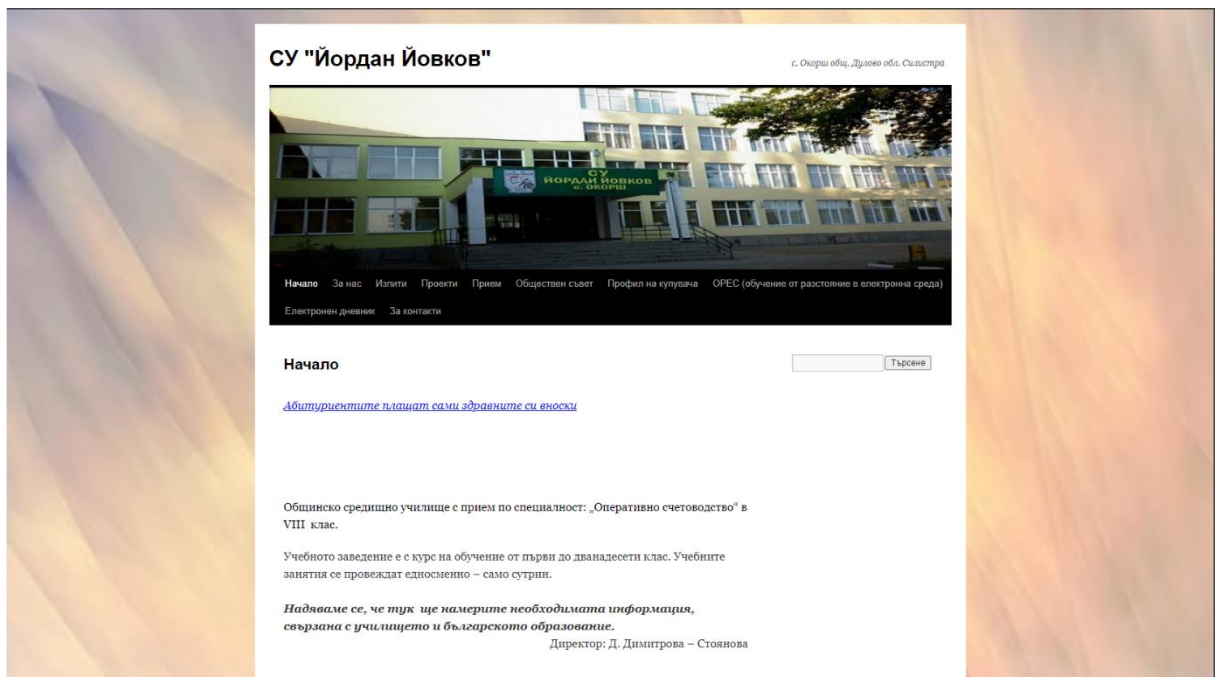
Фигура 19 Начална страница 1 част



Фигура 20 Начална страница 2 част

➤ Раздела "За нас":

В раздела "За нас" се представя основната информация за историята на училището, целите, химна на училището. Всяка една страница е лесно достъпна, дизайнът е удобен за използване. Страниците могат да се подобрят, като се създадат елементи, които да подпомогнат за стилизирането, като се създадат различни секции в някои страници, за да може потребителят да намира по-лесно нужната информация.

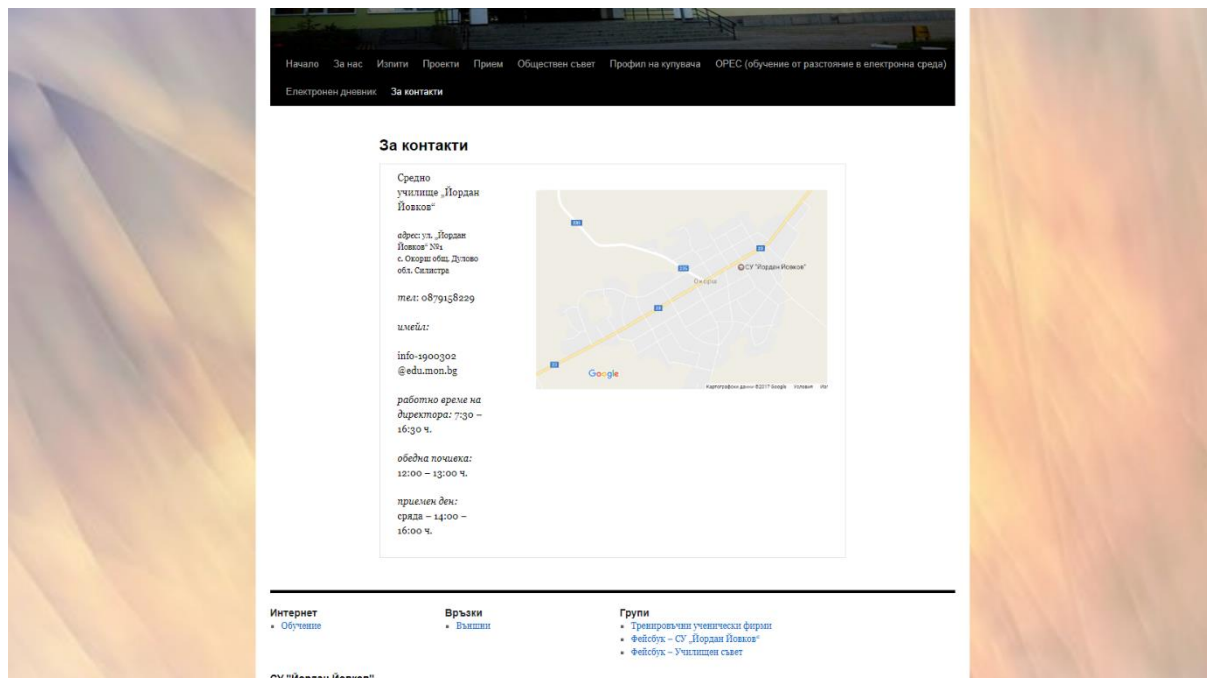


Фигура 21 Страница „За нас“

➤ Страница "За контакти":

Страницата за контакти предоставя основна информация за начина, по който потребителите могат да се свържат с училището, включително имейл, адрес, работно време, обедна почивка,

приемен ден. В тази страница може да се добави връзка към социалните мрежи на училището и форма за обратна връзка, чрез която потребителят да може текстово да зададе своите въпроси.



Фигура 22 Страница „Контакти“

Предимства:

- Приятни за потребителя цветовете;
- Добра подредба на менюто;

Недостатъци:

- Бавно зареждане между страниците;
- Недоразработен дизайн;
- Липса на формуляр;
- Не е налична мобилна версия на сайта;

2.3 Извод

След анализа на двата уеб сайта, става ясно, че и двата имат добре организирана структура със стилизиран и приятен дизайн. Менютата са удобни за потребителя и осигуряват лесна навигация между страниците за по-добро потребителско изживяване. Въпреки това, първият уеб сайт предлага мобилна версия, това разширява начините, по които потребителят може да използва самото приложение или уеб сайт, докато вторият не разполага с такава.

Когато се създава уеб приложение на подобна тема, от голямо значение е внимателно да се планира структурата на уеб сайта, като се избере подходящ дизайн и всяка функционалност трябва да работи правилно. Важно е също уеб сайтът да е достъпен за различни по размер екрани, тъй като използването на смартфони и таблети продължава да нараства.

Таблица 1

СУ "Васил Левски"	СУ „Йордан Йовков“
<p style="text-align: center;">Предимства</p> <ul style="list-style-type: none"> ✓ Приятни за потребителя цветове; ✓ Красив и интуитивен дизайн; ✓ Възможност за мобилна версия; ✓ Функционална търсачка; ✓ Форма за обратна връзка; 	<p style="text-align: center;">Предимства</p> <ul style="list-style-type: none"> ✓ Приятни за потребителя цветове; ✓ Организирано меню;
<p style="text-align: center;">Недостатъци</p> <ul style="list-style-type: none"> ✓ Проблеми със скоростта на зареждане между страниците; ✓ Проблем със Галерията; ✓ Недоразвит дизайн на навигационното меню; 	<p style="text-align: center;">Недостатъци</p> <ul style="list-style-type: none"> ✓ Недоразработен дизайн; ✓ Проблеми със скоростта на зареждане между страниците; ✓ Липса на формуляр; ✓ Не е налична мобилна версия;

Глава 3: Изграждане на уеб приложение за целите на образователния процес/образование

3.1 Мотивация за създаването на приложение/уеб сайт в рамките на образователния процес

В училище освен изучаването на програмни езици, е важно и прилагането им на практика. Затова трябва да се направи проект, с който учениците могат да видят практически как се използва наученото. Важно е да се обърне внимание на повече технологии и да се види каква роля играят те в създаването на приложение или уеб сайт.

С помощта на технологии като **HTML**, **CSS**, **Bootstrap**, **JavaScript** може да се създаде приложение. Комбинацията от тези технологии показва на учениците как се създава уеб сайт, как да го направят динамичен и с красив дизайн. По този начин се придобиват практически умения за решаване на проблеми и представа за работата на програмиста.

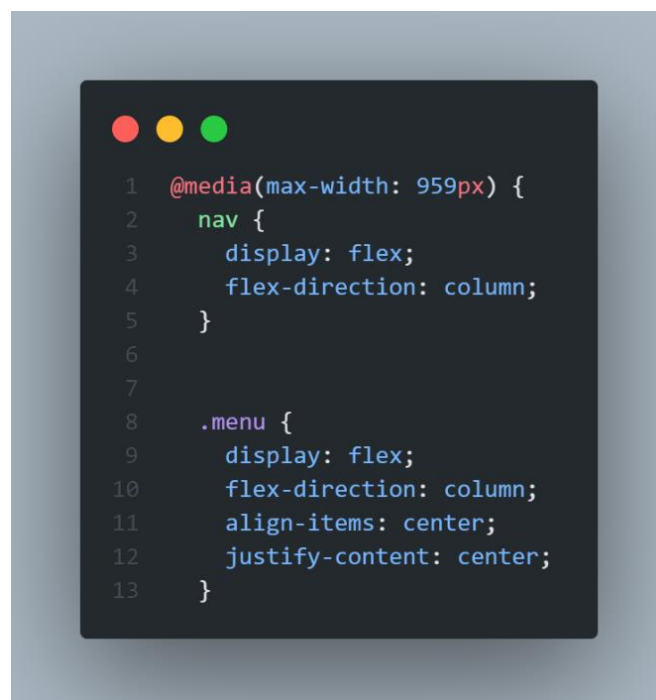
Мотивацията за създаването на приложение, е че учениците развиват ключови и практически умения в програмирането и уеб разработката. Те се научават да създават интерактивни и визуални елементи, както и придобиват важни знания и опит в управлението на тяхната функционалност.

➤ CSS:

Основната цел на кода е да се създаде адаптивно уеб оформление, което да е съвместимо към различни размери на екрана. Кодът е разделен на: навигационно меню, новини, контактна форма, учители, галерия, footer, адаптивен дизайн.

➤ Навигационно Меню:

Използва **flex** за наредбата на линковете, като така лесно се манипулира с **media query** визуализира на различните размери екрани. Така се създава специфичен дизайн за мобилна версия, който се показва при определени размери на екрана.



Фигура 24 CSS стилизация на навигационното меню

➤ Секция Новини:

На страницата с новини се използва **CSS Grid (grid-template-areas)** за динамично подреждане на новинарските полета, като структурата се пренарежда спрямо размерите на екрана. С помощта на псевдоелемента **::after** изображенията се затъмняват, което осигурява по-добър контраст и прави текста по-четим, без да се губи качеството на визията.



Фигура 25 CSS стилизация на клас „Новини“

➤ Секцията "Учители":

Визуализира информация за преподаватели, като се създават профилни карти, наредени с **flexbox layout method**, които включват снимка, име и бутон за допълнителна информация. Всеки бутон активира модален прозорец, показващ биографията на учителя, използвайки **Bootstrap** библиотеката. Той съдържа заглавие, изображение и автобиография на учителя.

```
1 <div class="profile-box">
2   
3   <h1>Иван Иванов</h1>
4
5   <!-- Button trigger modal -->
6   <button type="button" class="btn btn-primary" data-bs-toggle="modal" data-bs-target="#staticBackdrop">
7     Повече Информация
8   </button>
9
10  <!-- Modal -->
11  <div class="modal fade" id="staticBackdrop" data-bs-backdrop="static" data-bs-keyboard="false" tabindex="-1"
12    aria-labelledby="staticBackdropLabel" aria-hidden="true">
13    <div class="modal-dialog">
14      <div class="modal-content">
15        <div class="modal-header">
16          <h5 class="modal-title" id="staticBackdropLabel">Иван Иванов</h5>
17        </div>
18        <div class="modal-body">
19          
20          <p style="text-align: justify;">
21            Иван Иванов е директор на СОУ "Светлина" от 2015 година, с над 20 години опит в образованието.
22            Завършил е магистратура по педагогика в Софийския университет "Св. Климент Охридски" и е
23            специализирал управление на образованието. Работил е като учител по история и заместник-директор по
24            учебната дейност, като под негово ръководство училището значително подобрява академичните си
25            резултати. Г-н Иванов е посветен на индивидуалния подход към всеки ученик и активно работи за
26            създаване на позитивна учебна среда. В свободното си време се интересува от история, литература и
27            пътешествия.</p>
28          </div>
29          <div class="modal-footer">
30            <button type="button" class="btn btn-secondary" data-bs-dismiss="modal">Close</button>
31          </div>
32        </div>
33      </div>
34    </div>
35  </div>
```

Фигура 26 HTML кодова част на профилни карти

➤ Контактна Форма:

За добрия дизайн на полетата се използва **flexbox**, чрез който се подравнява формата, а гъвкавият дизайн подпомага за лесното подреждане при всички размери екрани, както и при библиотеката **Bootstrap**.

```
1 <div class="contact-box1">
2   <div class="full-name-div">
3     <label class="label-name" for="fullName-input">Full Name</label>
4     <input class="form-control" id="fName-input" type="text" placeholder="Full Name">
5   </div>
6   <div class="email-div">
7     <label class="label-name" for="email-input">Email Address</label>
8     <input class="form-control" id="email-input" type="email" placeholder="Email">
9   </div>
10  <div class="subject-div">
11    <label class="label-name" for="subject-input">Subject</label>
12    <input class="form-control" id="subject-input" type="text" placeholder="Subject">
13  </div>
14  <div class="message-div">
15    <label class="label-name" for="message">Message</label>
16    <textarea name="message" class="form-control textarea-size" id="message"
17      placeholder="Message"></textarea>
18    <input type="submit" value="Send Message" class="btn btn-primary">
19  </div>
20 </div>
```

Фигура 27 HTML кодова част на страницата „Контакти“



Фигура 28 CSS стилизация на формата от страница „Контакти“

➤ Галерия:

Галерията представлява множество изображения, които са наредени чрез **flexbox** и **bootstrap grid**.



Фигура 29 CSS стилизация на страница „Галерия“

➤ **JavaScript:**

JavaScript се използва за управление на навигационното меню, динамично отваряне и затваряне на подменюта и показване на изображения в модален прозорец.

- **Структура на кодовата част:**

Основните файлове в кодовата част включват **router.js**, **main.js** и необходимите **HTML файлове** за свързване на навигационното меню и галерията. Структурата може да бъде разделена на три основни компонента:

- **Навигация (Navigation):**

Тази част от кода управлява събитията върху навигацията и актуализира **URL адреса**, като същевременно филтрира съдържанието чрез функцията **Router**. Така симулира така наречените **SPA(single page application)** приложения. За по-добра функционалност на началната страница се зарежда.

```

1  import { router } from './router.js';
2
3  const navigationElement = document.querySelector(".menu");
4
5  navigationElement.addEventListener('click', (e) => {
6      e.preventDefault();
7      // Click on 'A'?
8      if (e.target.tagName == 'A') {
9
10         // Get element's href
11         let url = new URL(e.target.href);
12
13         // Import to router function pathname
14         router(url.pathname);
15     }
16 });
17
18 //? View a home page after loading website
19 document.addEventListener('DOMContentLoaded', () => {
20     router('/');
21 });

```

Фигура 30 JS кодова част на менюто

```

1  import { renderHome } from './page/home.js';
2  import { renderContact } from './page/contact.js';
3  import { renderTeachers } from './page/teachers.js';
4  import { renderNews } from './page/news.js';
5  import { renderGallery } from './page/gallery.js';
6
7  const routes = {
8      '/': renderHome,
9      '/contact': renderContact,
10     '/news': renderNews,
11     '/gallery': renderGallery,
12     '/teachers': renderTeachers,
13 }
14
15 export function router(path){
16     hiddenContent();
17
18     const renderer = routes[path];
19     renderer();
20 }
21
22 export function hiddenContent(){
23     const section = document.querySelectorAll(".view-section");
24     section.forEach((v) => v.style.display = "none");
25 }

```

Фигура 31 JS код на router.js файл

- **Мобилно меню (Mobile Menu):**

Тази част от кода управлява показването и скриването на мобилното меню, въз основа размера на екрана. По този начин, менюто става гъвкаво и ефективно на всички устройства.



```
1  const toggle = document.querySelector(".toggle");
2  const menu = document.querySelector(".menu");
3
4  // Функция за показване/скриване на мобилното меню
5  function toggleMenu() {
6    if (menu.classList.contains("active")) {
7      menu.classList.remove('active');
8      toggle.querySelector("a").innerHTML = "<i class='fas fa-bars'></i>";
9    } else {
10     menu.classList.add("active");
11     toggle.querySelector("a").innerHTML = "<i class='fas fa-times'></i>";
12   }
13 }
```

Фигура 32 JS кодова част за мобилно меню

3.3 Преглед на приложението от учебна гледна точка

- **Десктоп версия**

- **Начална страница**

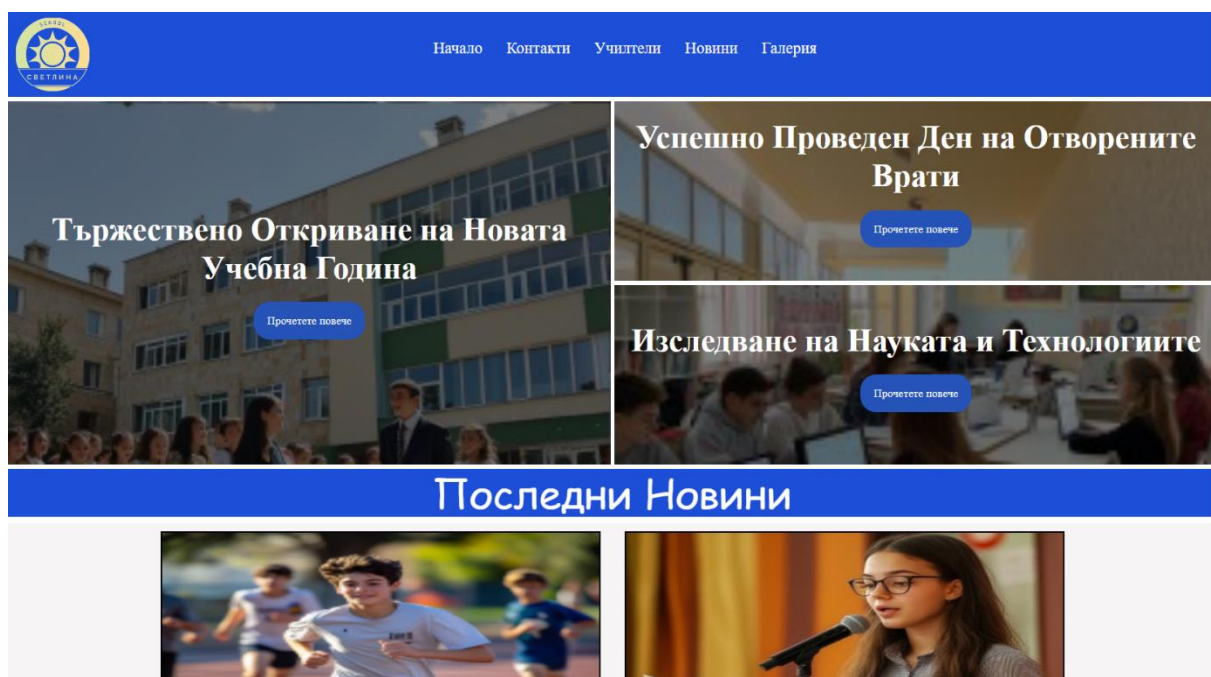
В началната страница се открива много необходими компоненти, които учениците трябва да знаят за разработването на уеб сайт/ приложение. Започва с меню, което включва лого и линкове към другите страници. Тук се изучава как се създава меню чрез **css** свойствата: **color**, **text-decoration**, **cursor**, **flex** и др.

Следва секцията с новините, като тук учениците изучават как се работи с **html** елементи като: **div**, **h1**, **button** и **css** свойства като: **grid**, **flex**, **z-index**, **background**.

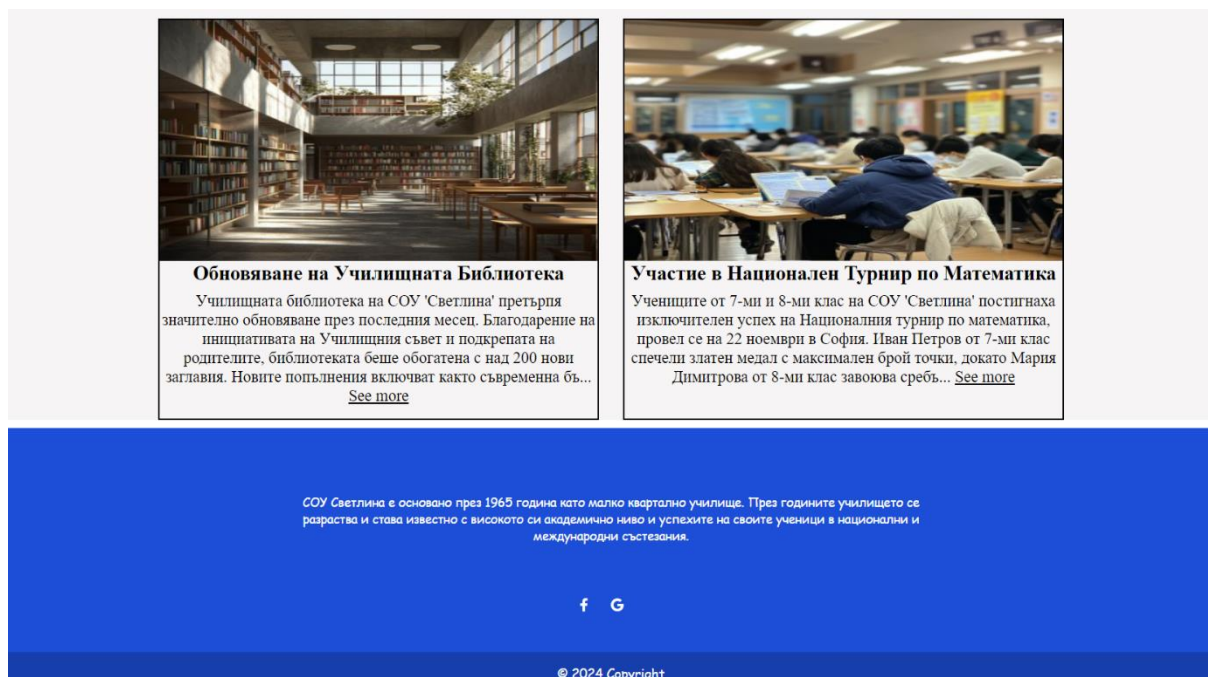
Под новините се намира лента, която озаглавява следващата секция - „**Последни Новини**“, където се изучава какво представлява **html** елемента **hr**, и как се стилизира чрез **css** свойства.

В секцията за „**Последни новини**“ съществуват други под секции, които са наредени с **flexbox**, като тук учениците изучават допълнителни свойства на **flex** като: **flex-wrap**, **justify-content**.

Най-отдолу на страницата се намира **footer** частта, в която учениците ще придобият умения за вграждане на **bootstrap components**. Заедно с това те ще могат да подобряват тези компоненти с познанията си в **css**.



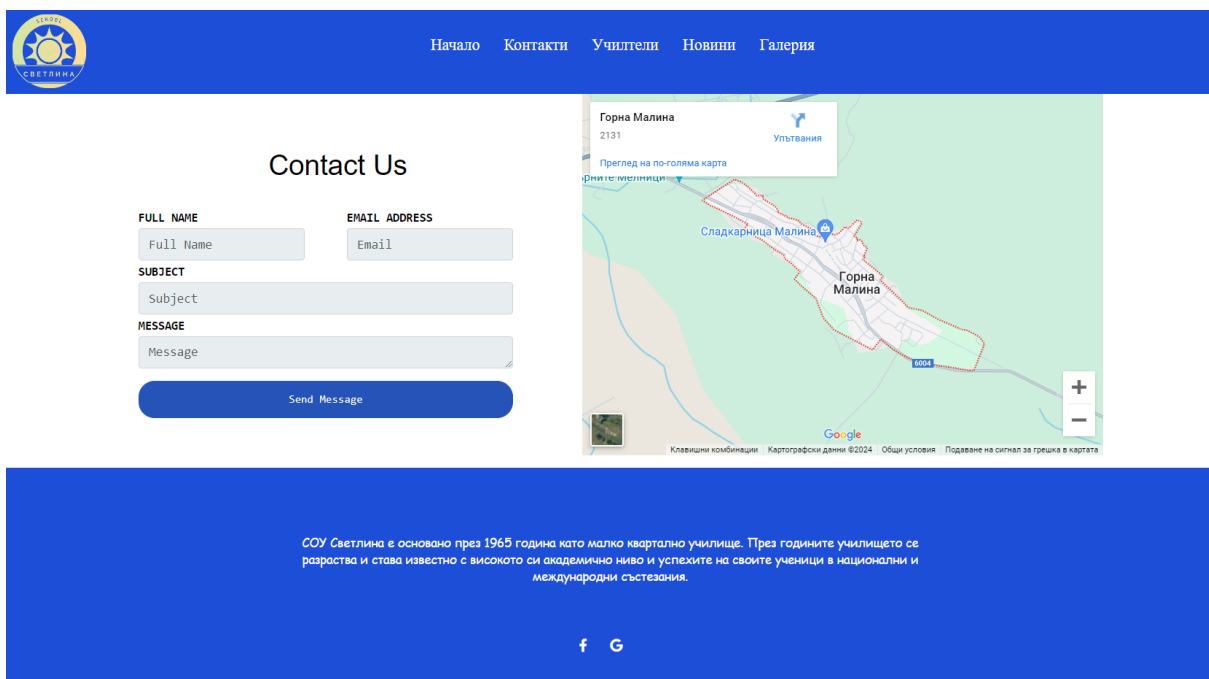
Фигура 32 Начална страница част 1



Фигура 33 Начална страница 2 част

➤ Контакти

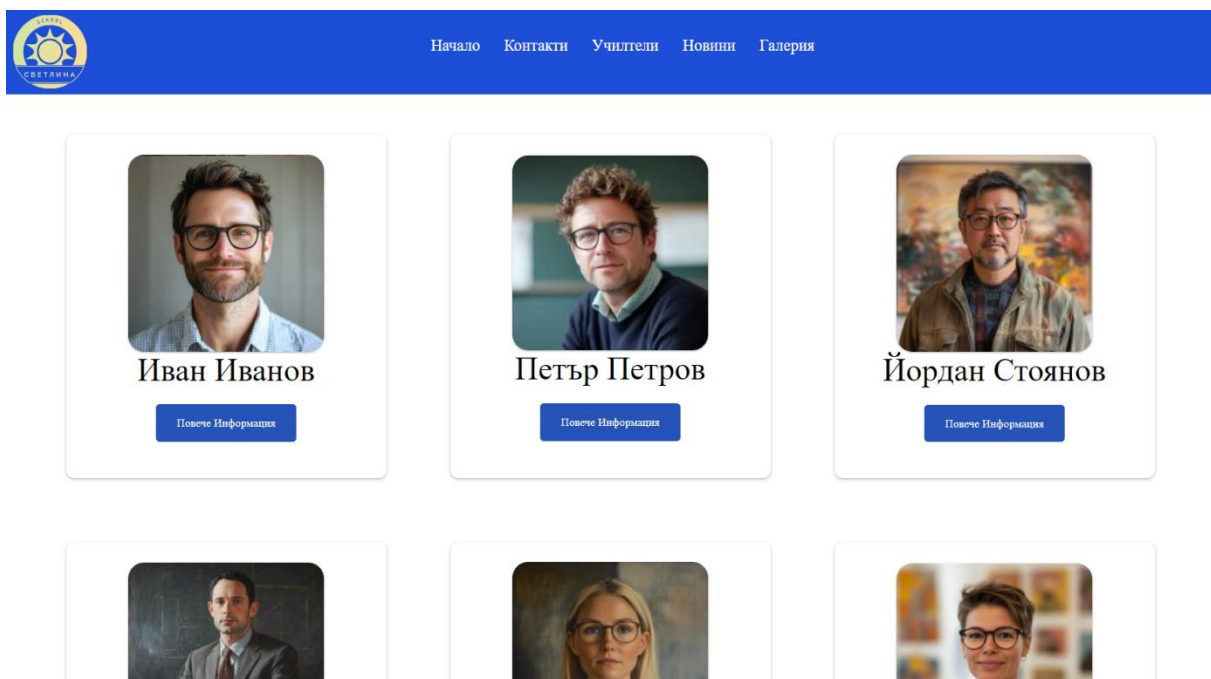
В страницата контакти учениците се сблъскват с контактните форми, където ще научат какво са **input**, **label**, **textarea** и как могат да интегрират карта. Ще научат как се стилизират тези **html** елементи с **bootstrap** и **css**.



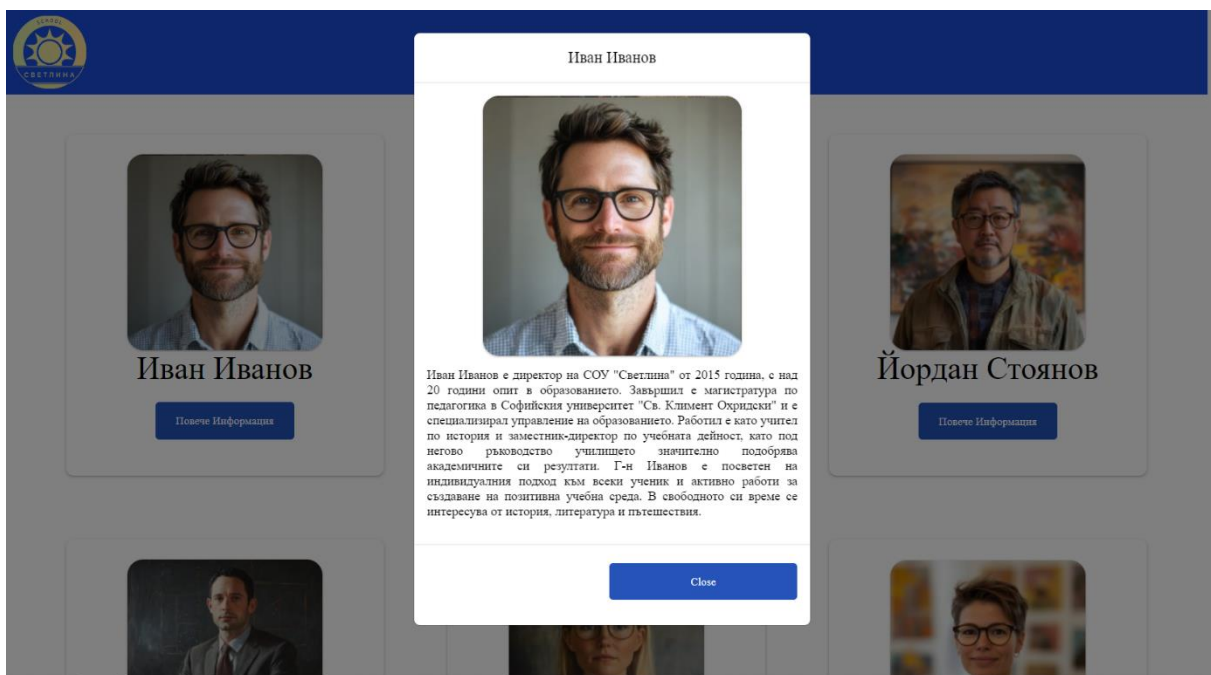
Фигура 34 Страница на „Контакти“

➤ Учители

В страницата за учители, се разглежда отново вграждането на **bootstrap components**, като тук един компонент се произползва за всяка една секция на учител. Уменията, които придобиват учениците са работа с **Bootstrap**, **JavaScript** и **CSS**.



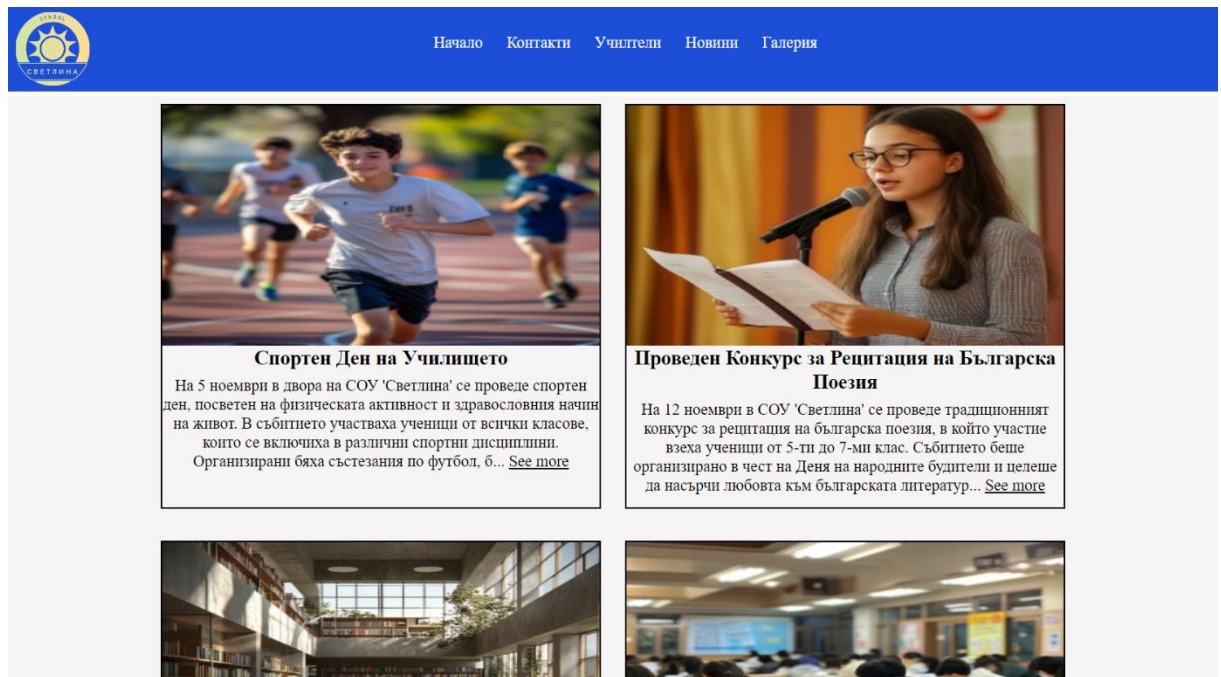
Фигура 35 Страница на „Учители“



Фигура 36 Профилна карта от страница на „Учители“

➤ Новини

В страницата за новини се разглежда подобни секции за новини, каквито са описани в **„Начална страница“**, където се намират всички новини от уеб сайта.



Фигура 37 Страница на „Новини“

➤ Галерия

В страницата на галерия, се изучава **html** елементът **img**. Учениците тук прилагат специфичното нареждане на изображенията с **flex** и **flex-wrap**, като се разглежда и ново свойство на **css** **“gap”**.

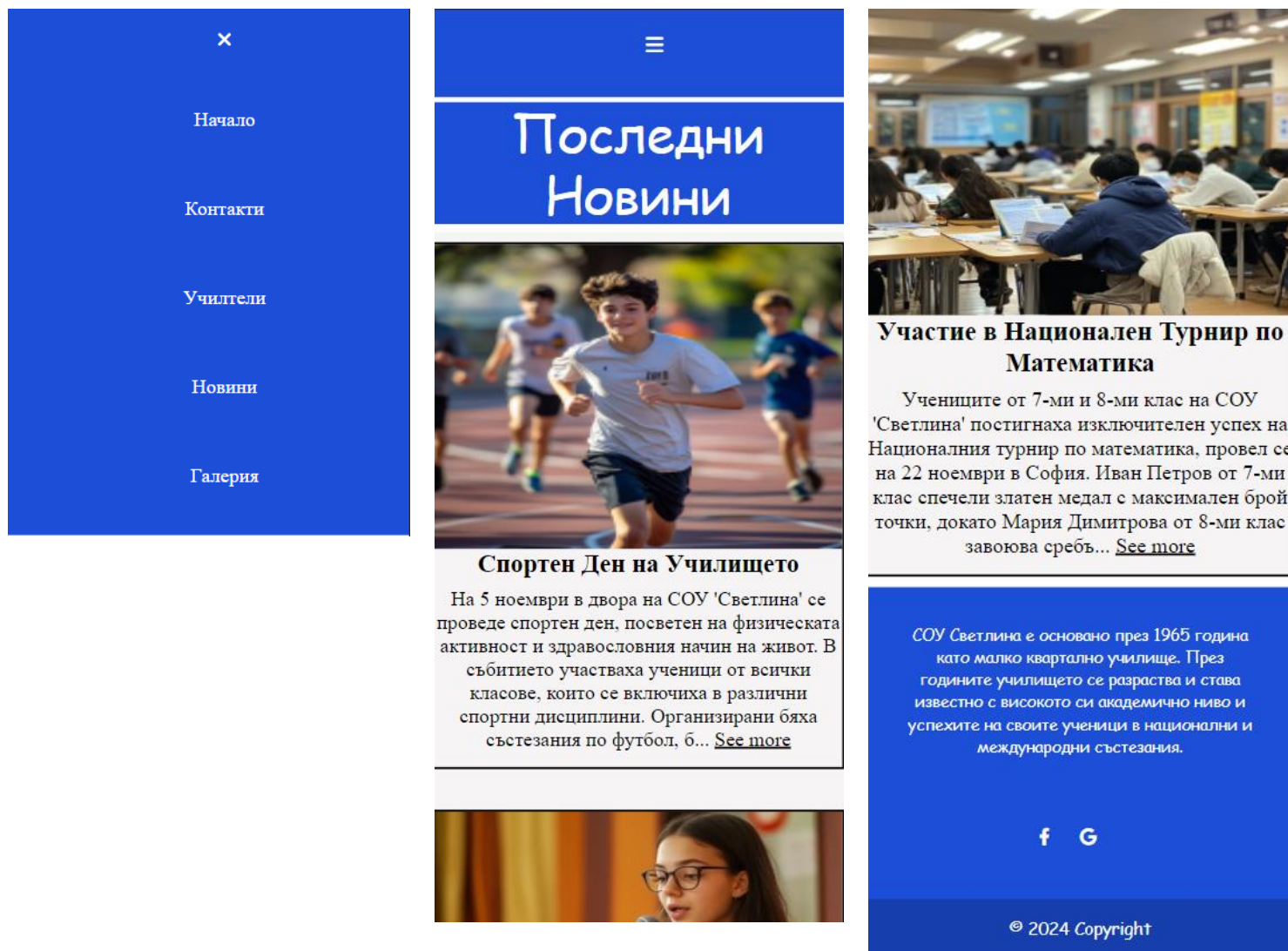


Фигура 37 Страница на „Галерия“

- **Мобилна версия**

- **Начална страница**

В началната страница учениците ще научат как да променят дизайна на навигационното меню с помощта на **Responsive Design** и **JavaScript** и как да скриват и показват елементи в зависимост размера на екрана. Секцията за произволни новини се скрива и секцията за последни новини се подрежда едно под друго.



Фигура 38 Мобилна начална страница

➤ Контакти

В страницата за контакти учениците ще разгледат как формата може да бъде реорганизирана и картата мащабирана, спрямо екраните на различни устройства.

Contact Us

FULL NAME

Full Name

EMAIL ADDRESS

Email

SUBJECT

Subject

MESSAGE

Message

Send Message

Горна Малина

Преглед на по-голяма карта

Сладкарница Малина

Горна Малина

6004

+

-

Google

Клавищни комбинации Картографски данни ©2024 Общи условия

Фигура 39 Мобилна страница на „Контакти“

На страниците за „Учители“, „Новини“ и „Галерия“ учениците ще видят как разгледаните принципи на **flexbox** се прилагат за създаване на адаптивен дизайн.



Иван Иванов

[Повече Информация](#)



Петър Петров

[Повече Информация](#)

Иван Иванов



Иван Иванов е директор на СОУ "Светлина" от 2015 година, с над 20 години опит в образованието. Завършил е магистратура по педагогика в Софийския университет "Св. Климент Охридски" и е специализирал управление на образованието. Работил е като учител по история и заместник-директор по учебната дейност, като под негово ръководство училището значително подобрява академичните си резултати. Г-н Иванов е посветен на индивидуалния подход към всеки ученик и активно работи за създаване на позитивна учебна среда. В свободното си време се интересува от история, литература и пътешествия.

[Close](#)



Фигура 40 Мобилна страница на „Учители“



Спортен Ден на Училището

На 5 ноември в двора на СОУ 'Светлина' се проведе спортен ден, посветен на физическата активност и здравословния начин на живот. В събитието участваха ученици от всички класове, които се включиха в различни спортни дисциплини. Организирани бяха състезания по футбол, б... [See more](#)



Фигура 41 Мобилна страница на „Новини“

3.4 Бъдещо развитие

Разработването на уеб приложението, което се разглежда в „3.3 Преглед на приложението от учебна гледна точка“ помага на учениците да придобият практически умения в уеб програмирането и дизайна. След като усвоят основите на технологиите **HTML**, **CSS** и **JavaScript**, ще прилагат знанията си на практика. Учениците могат да развият проекта, и когато е готов да го добавят в своето портфолио, което ще бъде изключително ценно при кандидатстването им в различните университети в страната и чужбина. Освен това придобитите знания могат да бъдат прилагани и в разработването на лични проекти.

Този проект развива техническите им умения и способностите за решаване на проблеми, които ще бъдат полезни във всяка една сфера на програмирането. Учениците могат да надградят уеб сайта, като добавят нови функционалности и като оптимизират скоростта при зареждане на отделните страници, по този начин показват своя прогрес, знания и умения в уеб приложенията.

Заклучение

В процеса на разработване на дипломната работа се прилагат следните етапи: **планиране на уеб сайт, анализ на подобни приложения, изграждане на уеб приложение за целите на образователния процес/образование, бъдещо развитие.**

Дипломната работа представя процеса на разработване на уеб сайт в училищна среда, като е насочена върху съвременните уеб технологии и тяхното приложение в изграждането на функционален и привлекателен проект.

Анализирани са подобни уеб сайтове и са изведени ключовите изисквания за създаване на по-ефективен новинарски сайт, който предлага удобно потребителско преживяване и адаптивност за различни устройства. Също така предоставя възможност на учениците да придобият практически умения и знания в областта на уеб разработката. Те могат да развият критично мислене, умения за работа в екип и креативност, което го прави отлична основа за тяхното бъдещо развитие в сферата на технологиите и дигиталния свят. Ключово за уеб приложението, което е описано в трета глава, е че предлага различни нива на трудност, в зависимост от нивото на учениците, така преподавателят може да избира различни варианти на трудност.

Източници

1. Денис Колишниченко, Денис. JavaScript и jQuery: Практическо програмиране. София: Издателство "Book Store", [2012].
2. Денис Колишниченко. *HTML 5 и CSS 3: Практическо програмиране за начинаещи*. София: Издателство "Book Store", [2012].
3. Денис Колишниченко. Адаптивен уеб дизайн с Bootstrap. София: Издателство "Book Store", [2012].
4. <http://suokorsh.org/>
5. <https://directiveconsulting.com/resources/glossary/responsive-design/>
6. <https://getbootstrap.com/docs/5.0/getting-started/introduction/>
7. <https://landforce.armf.bg/>
8. <https://medium.com/@maryam.salman70/responsive-web-design-creating-a-seamless-user-experience-across-devices-a15b25d431b7>
9. <https://react.dev/>
10. https://softuni.bg/?utm_source=GO&utm_medium=cpc&utm_campaign=S-SU-OurBrand-ProgrammingBasics&utm_term=softuni&utm_content=SoftUni&gad_source=1&gclid=CjwKCAjw65-zBhBkEiwAjrQRM099uvZ8OmCPI_lvAX-AdOARyFC2j4F2KNZZVic6UWx-zCH11dWulBoCe0MQAvD_BwE
11. <https://suapriltsi.com/>
12. <https://www.linkedin.com/pulse/whats-new-bootstrap-4-husain-s-/>

Актуални към дата 25 август 2024 година