**ВИСОКА ТЕХНИЧКА ШКОЛА СТРУКОВНИХ СТУДИЈА**

**НОВИ САД**

**ЗАВРШНИ РАД**

**Студијски програм - Информационе технологије**

**Тема:HTML, CSS и Javascript у креирању веб сајтова и апликација**

Професор ментор: Ученик:

Проф.др. Жељко Еремић Душан Лучић IT72/19

Нови Сад, август 2023. год.

САДРЖАЈ

[1. Увод 3](#_Toc144040805)

[2. Технологије израде 4](#_Toc144040806)

[2.1. HTML 4](#_Toc144040807)

[2.2. CSS 5](#_Toc144040808)

[*2.3.* JavaScript 6](#_Toc144040809)

[3. Конкретна имплементација на веб страници 8](#_Toc144040810)

[3.1. Структурирање HTML-а 8](#_Toc144040811)

[3.1.1. Таг <html> 9](#_Toc144040812)

[3.1.2. Таг <head> 9](#_Toc144040813)

[3.1.3. Таг <body> 10](#_Toc144040814)

[3.1.4. Футер 16](#_Toc144040815)

[3.2. Уређивање текста CSS 18](#_Toc144040816)

[3.2.1. Наслови и поднаслови 18](#_Toc144040817)

[3.2.2. Топ Мени 18](#_Toc144040818)

[3.2.3. Logo 19](#_Toc144040819)

[3.2.4. Nav Bar 19](#_Toc144040820)

[3.2.5. Top News 20](#_Toc144040821)

[3.2.6. Tab News 21](#_Toc144040822)

[3.2.7. Footer 21](#_Toc144040823)

[4. БМИ – индекс телесне масе 23](#_Toc144040824)

[4.1. Фунционалност програма БМИ 24](#_Toc144040825)

[4.1.1. Класа Container 24](#_Toc144040826)

[4.1.2. Функција calculate() 25](#_Toc144040827)

[4.1.3. Приказ резултата 26](#_Toc144040828)

[5. Закључак 27](#_Toc144040829)

[6. Литература 28](#_Toc144040830)

# Увод

Живимо у ери дигитализације, сведоци смо револуције у начину на који се информације стварају, складиште, преносе и размењују. Дигитализација је постала кључни покретач напретка у готово свим сферама живота, а један од најистакнутијих аспеката ове трансформације је брзина размене информација. Управо захваљујући дигитализацији, данас смо у могућности да веома брзо приступимо и поделимо информације, што има огроман утицај на наше друштво, економију и свакодневни живот.

Дигитализација представља процес конверзије аналогних података и информација у дигитални облик, који се може манипулисати, складиштити и преносити путем електронских уређаја и мрежа. У ери дигитализације, физичке копије књига, новина и докумената постепено уступају место електронским верзијама које су доступне путем рачунара, телефона и других дигиталних уређаја. Један од најважнијих аспеката дигитализације, као што је већ наведено, је брзина размене информација. Раније, пре дигиталног доба, пренос информација био је ограничен физичким средствима као што су писма, телеграми или телефонски позиви. Ови процеси су често захтевали време и физичку доставу, што је успоравало комуникацију и размену информација на глобалном нивоу.

Интернет је кључни покретач брзе размене информација у дигиталном добу. Данас, људи широм света могу тренутно делити идеје, документе, фотографије и видео снимке путем платформи за друштвене мреже, е-поште, апликација за размену порука и других дигиталних канала комуникације. Информације којима су били потребни дани или недеље да стигну на одредиште, сада се могу пренети у неколико секунди или минута.

Брзина размене информација има огроман утицај на пословни свет. У економији која се све више ослања на дигиталне технологије, брза размена података омогућава компанијама да доносе одлуке на основу реалних временских информација. Уз помоћ брзе размене информација, компаније могу пратити тржишне трендове, одговарати на потребе купаца и реаговати на промене у реалном времену.

Комуникација са пријатељима и породицом, управљање финансијама, приступање образовним ресурсима, проналажење информација и забава - све то постаје доступно са само неколико кликова. Друштвене мреже омогућавају људима да остану повезани и обавештавају се о дешавањима у реалном времену. Путовања су постала једноставнија, са могућношћу тренутне резервације летова, хотела и возова.

Када свим наведеним разлозима додамо и чињеницу да спорт игра веома важну улогу у нашем животу, долазимо и до разлога зашто је тема овог завршног рада израда веб-странице *Спортски портал*. Од овог портала се очекује да свим заинтересованим лицима омогући лак приступ актуелним спортским вестима, информацијама о спортистима и догађајима.

# Технологије израде

## HTML

*HTML* (Hypertext Markup Language) је стандардни језик за израду веб-страница. Користи се за дефинисање структуре и садржаја веб-страница. *HTML* је језик за означавање хипертекста (енгл. *Hyper Text Markup Language*). Представља описни језик специјално намењен опису веб-страница.

Ако говоримо о *HTML*-у што и јесте тема овог рада, онда ћемо најпре описати неке основне функције а потом и напредне функције *HTML*-а.

*HTML* (Hypertext Markup Language) је језик који се користи за креирање докумената на World Wide Web-u, тј. он одређује структуру, садржај и функцију *HTML* документа. *HTML* нам служи за одређивање логичке и физичке структуре унутар *HTML* документа, тј. за одређивање наслова, пасуса, слика, хипертекстуалних веза, а такође нам омогућава да те исте елементе распоредимо у већој или мањој мери у погледу стила или графике. *HTML* је језик из *SGML*-а (Standard Generalized Markup Language), опште прихваћени стандардни језик за означавање. *HTML* је створен да омогући неке врло једноставно структурисане интернет странице - пасусе, преломе редова и заглавља.

Помоћу *HTML*-a се једноставно могу одвојити елементи као што су наслови, параграфи, цитати и слично. *HTML* дефинише скуп заједничких елемената за веб-странице: заглавља, пасусе, листе и табеле. Такође, дефинише формате знакова, као што су подебљана слова и примери кода. Поред тога, у *HTML* стандард су уграђени елементи који детаљније описују сам документ као што су кратак опис документа, кључне речи, подаци о аутору и слично. Ови подаци су познати као мета-подаци и јасно су одвојени од садржаја документа. Ови елементи и формати означени су унутар *HTML* докумената помоћу **тагова**. Сваки таг има одређени назив и активира се из садржаја документа помоћу ознаке.

*HTML* се првобитно заснивао на *стандардном општем језику за означавање* (*SGML*), чија је једна од главних функција да описује општу структуру садржаја у документу, а не његов стваран изглед на екрану. Због свог *SGML* наслеђа, *HTML* је језик за опис структуре документа, а не за његову стварну презентацију.

У почетку је пружао углавном елементарне ствари за означавање и форматирање текста. Како је веб растао тако је расла и потреба за богатијим садржајем, те је у том смеру развијан и *HTML* стандард, а његова актуелна верзија је *HTML 5*.

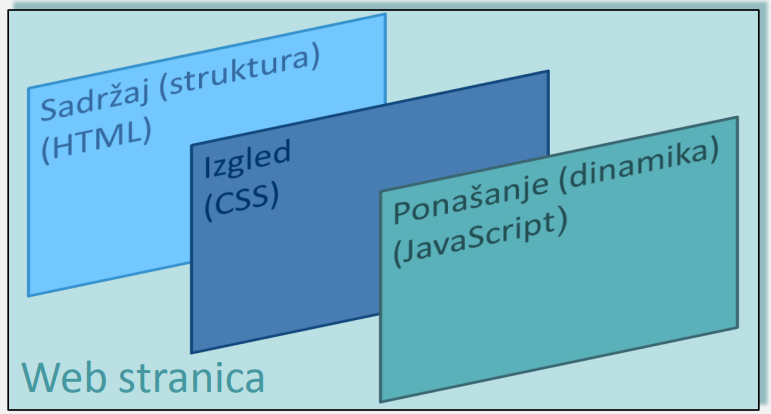
## CSS

*CSS* (Cascading Style Sheets) је језик форматирања помоћу кога се дефинише изглед (стил) веб-страна. Кад се појавио Веб једино је *HTML* дефинисао комплетан изглед, структуру и садржај веб-страница. Када су у *HTML* 3.2 додати тагови почињу потешкоће за веб-програмере, јер *HTML* документи постају већи, и врло је тешко било сналазити се у њима. Да би се упростио такав *HTML* документ од верзије *HTML* 4.0 уведен је CSS (1996 године).

Како је *HTML* постајао компликованији, давао је све више могућности за дефиницију изгледа елемената, али је истовремено постајао мање читљив и тежи за одржавање. Пошто *HTML* не говори много о приказу странице, развијен је посебан језик који се користи за описивање изгледа и геометријског распореда свих елемената на страници. Језиком *CSS* се представљају правила која дефинишу како треба да изгледају елементи на веб-страни.

*CSS* има далеко веће дизајнерске могућности од чистог *HTML* -а. Зато је данас *CSS* готово незаменљив у дизајнирању веб-страница. *CSS* синтакса се састоји од описа изгледа елемената у документу. Опис може да дефинише изглед више елемената, и више описа може да дефинише један елемент. На тај начин се описи слажу један преко другог да би дефинисали коначни изглед одређеног.

***Данашње веб-странице деле се на:***

* *HTML* део који дефинише садржај (структуру) веб-странице.
* *CSS* део који дефинише изглед веб-странице (као на пример боја слова, позадине, маргине и тако даље).
* *JavaScript* део који дефинише динамику веб-странице.

Слика 1 – Данашње веб-странице

Правила *CSS*-a су бројна и битно је пратити смернице за правилну употребу. Подразумева се коришћење **бирача** (енгл. *selector*), који садржи листу **својстава** и **вредности**, повезану са тим својствима. Сва својства која су постављена у бирачу су затворена витичастим заградама.

***CSS*** ***код се може задавати на три стандардна места:***

* Директно у *HTML* тагу, користећи аргумент стyле (инлине).
* У заглављу документа унутар тага < style>.
* У екстерној датотеци, која се линкује тагом <link>

***Инлајн стил*** се убацује директно у *HTML* таг, користећи аргумент style. Овим се нарушава намена *CSS* -а да се листа стилова примењује на цео веб-сајт.

***Уграђени стил*** се убацује директно у заглавље документа, између почетне и завршне ознаке (тага) стила (<style> ... </style>).

Да бисте у потпуности искористили предности *CSS*-а, треба да користите екстерни стил. Екстерни стил је текстуална датотека са екстензијом .css.

Стилови у екстерној датотеци имају предност у односу на преостала два начина (стилови дефинисани у заглављу и инлине стилови), а то је да помоћу екстерне датотеке можемо да утичемо на елементе више докумената (страница) и централизовано да управљамо стиловима и изгледом дела или целог сајта.

Препоручени начин дизајна модерних веб-страна је формирање одвојеног *CSS* фајла који садржи дефиниције стилова чиме се обезбеђује прегледнији изглед свих страна, јер више *HTML* фајлова може да се укључи у исти *CSS* документ.

## JavaScript

*JavaScript* је данас присутан свуда. Компаније које су развиле претраживаче сада се утркују у креирању најбржих *JavaScript* виртуелних машина чиме се отварају могућности ширења примене *JavaScript* -а у областима обраде слика, аудио и видео записа, развоја игара и модерних апликација применом одређених окружења.

У *JavaScript* -у можемо рећи да променљиве немају типове већ да вредности имају типове. То значи да сам језик разликује типове, али да програмер не мора унапред задавати типове променљивама, већ их рачунар аутоматски препознаје.

Такође, у *JavaScript* -у је дозвољено да иста променљива током извршавања програма добија вредности различитих типова. Одвајање меморије за сваку променљиву се не врши пре извршавања програма (тј. током компајлирања), већ током извршавања, сваки пут када променљивој доделимо нову вредност. Супротно од строго типизираних језика код којих важи да свака променљива унапред мора да има задат тип, језици попут *JavaScript* -а су слабо типизирани (engl. loosely typed).

*JavaScript* је програмски језик који се користи за претварање веб-страница у апикације. Користи се за управљање садржајем веб-стране и омогућава корисницима интеракцију са тим страницама без њиховог поновног учитавања.

***JavaScript може бити повезан са HTML документом на три начина:***

* коришћењем ознаке <script> унутар самог документа,
* додавањем догађаја унутар појединих *HTML* елемената и
* повезивањем са екстерном .js датотеком преко атрибута src у тагу <script>.

Слика 2 – Ток података од веб-сервера до прегледача

Данас је подршка за *JavaScript* јака у свим прегледачима, тако да се аутори прегледача више не утркују у подршци већ у брзини извођења одређених алгоритама.

Попис прегледача који подржавају стандард ECMAScript 2021:

• Microsoft Edge

• Firefox

• Safari

• Chrome

• Opera

У наредним поглављима биће описан поступак израде веб-странице *Спортски портал* применом наведених технологија.

# Конкретна имплементација на веб страници

## Структурирање HTML-а

Текст можемо форматирати и уређивати помоћу основних *HTML* елемената и стилова и њихових атрибута. Елементи *HTML* стила користе се када желите да промените графички изглед садржаног текста, а атрибути служе као помоћ неким елементима, што им помаже да раде потпуно коректно. Састоје се од имена и вредности. Неки од елемената су:

1. Пасус (<p>) или пасус који представља одређену логичку јединицу садржаја.

2. Наслов (од <h1> до <h6>) који користимо за одвајање и представљање различитих тематских целина документа. H1 је највиши ниво наслова, а h6 најнижи.

3. Нови ред (<br>) је елемент који се користи при премештању садржаја у нови ред у пасусу.

4. Немогућност нарушавања налога (<brbr>). Ако је текст превелик да стане у један ред на монитору, на дну прозора ће се појавити водоравни „клизач” и корисник ће морати да помиче да га прочита.

5.Предформатирани текст (<pre>) разликује се од обичног текста по томе што је сваки знак исте ширине. Најчешће се користи када желите да представите део рачунарског кода на веб-страници или желите да задржите размаке и нове редове без употребе посебног *HTML* знака & нбсп; или елемент <br/>

6. Фонт (<font>) је идентификатор који нам омогућава да одредимо фонт текста на нашој страници и његове атрибуте, величину и боју тог фонта.

Неки од стилова су:

1. Bold (<b>) се користи за подебљавање текста

2. Italic (<i>) се користи за искривљавање текста

3. Underline (<u>) се користи за подвлачење текста

4. Strike (<s>) користи се за прецртавање текста

5. Subscript (<sub>) се користи за стављање текста на место индекса

6. Superscript (<sup>) користи се за стављање текста на место експонената

Неки од атрибута:

1. Align је атрибут који се користи за поравнање елемената наше странице (текст, слика, табела итд.) улево, удесно или у центар, тако да су његове вредности: лево, десно, средина.

2. Color је атрибут који користимо за одређивање боје фонта. Постоје разне друге варијације овог атрибута, као што је background-color, који се користи за одређивање боје позадине.

3. Size је атрибут који користимо за промену величине фонта. *HTML* разликује 7 величина фонтова које носе вредности од 1 до 7, али такође можемо изразити величине фонтова у пикселима.

4. Face је атрибут који користимо за одређивање врсте фонта нашег текста

Листа је такође елемент *HTML* -а који се може користити за форматирање текста, односно за одређивање како ће текст бити приказан на нашој страници. Спискове смо одвојили од осталих елемената, јер ћемо о њима рећи нешто више него о осталим елементима. Листа је пасус који садржи низ елемената листе. Постоје три врсте спискова: нумерички, симболички и дефиницијски. Могуће је угнездити их, па су идеално средство за дефинисање логичке структуре, нпр. мени стабла.

У нумеричким списковима елементи су означени редним бројевима или словима. Елемент <ol> ( уређена листа) користи се за дефинисање нумеричке листе, а елемент <li> користи се за дефинисање појединачних елемената листе.Симболични елементи листе означени су круговима или неким другим симболима. Елемент <ul> ( неуређена листа ) користи се за дефинисање нумеричке листе, док се <li> користи за дефинисање појединачних елемената листе као код нумеричке листе. За листе дефиниција, сваки елемент листе састоји се од два поделемента: израз дефиниције <dt> (дефиниција појам) и саме дефиниције <dd> ( опис дефиниције ). Елемент <dl> се користи за дефинисање листе дефиниција (списак описа ).

Тагови и класе структуре не утичу на изглед странице када се она приказује – они су ту само да би помогли прегледачима.

Након што у жељеном уређивачу текста направимо нову датотеку са екстензијом .html, потребно је да дефинишемо следеће ознаке.

### Таг <html>

Први таг структуре на свакој *HTML* страници је <html>. Овај таг означава да је садржај датотеке у језику *HTML* и служи као оквир за све тагове који чине страницу. Таг <html> би требало да уследи одмах после идентификатора DOCTYPE, који се користи да дефинише верзију *HTML*-a којом је урађена страница.

### Таг <head>

У <head> тагу су смештени тагови који садрже информације о страници. На следећој слици су приказани употребљени тагови:



Слика 3 – Приказ <head> тага на веб-порталу

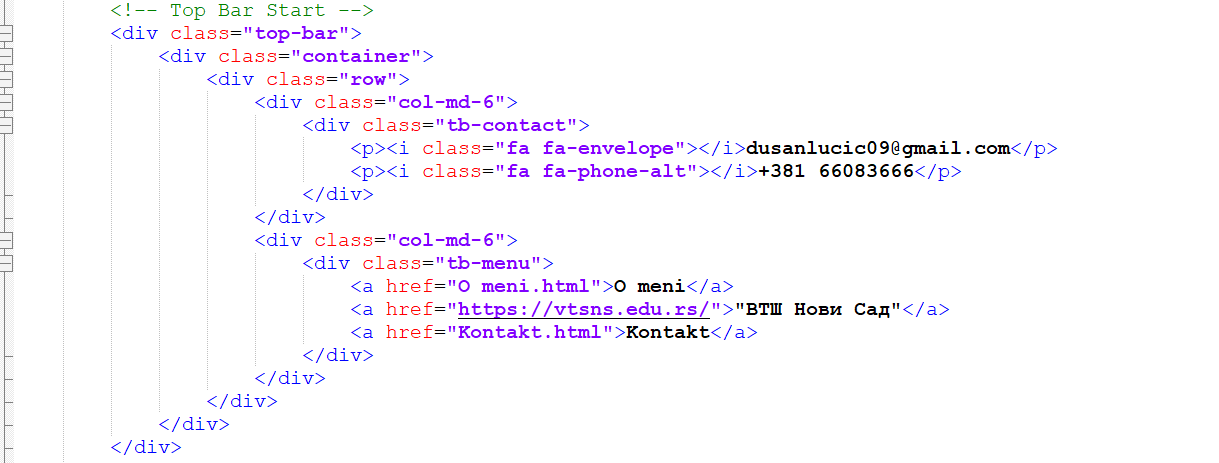
Једна од најважнијих намена <meta> ознаке је дефинисање кодне стране документа. Помоћу овог податка, прегледач може правилно да растумачи начин кодирања текста на страници. Да бисмо дефинисали кодну страну, користимо параметар charset. Ово је специфичан облик ознаке <meta> којем је прослеђена вредност utf-8, која подразумева UNICODE стандард.

За доделу наслова страници користи се таг <title>, чији ће садржај бити приказан у картици прегледача. Могуће је, на истом месту, приказати и лого странице, који се дефинише у оквиру елемента <link>, користећи атрибут href који води до сродне путање слике која представља лого; атрибут rel који дефинише тип везе између тренутног документа и оног који се повезује с њим и атрибут type који одређује тип повезаног документа.

Ознака <link> је употребљена и за повезивање *HTML* документа са датотеком style.css у којој су смештене све стилске одреднице за дату страницу, као и за укључивање библиотеке фонтова.

### Таг <body>

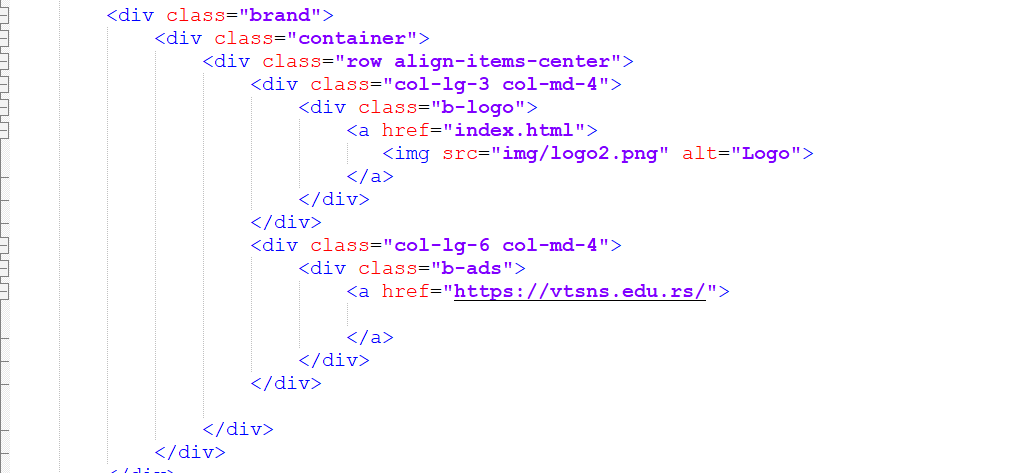
Садржај странице налази се унутар тага <body> који обухвата сав садржај портала као што су текст, линкови, слике... Овај таг је угњежден унутар тага <html>, а исто важи и за таг <head>. Сви *HTML* тагови функционишу тако што формирају појединачне угњеждене делове текста.





Слика 4 – Топ мени Спортски портал

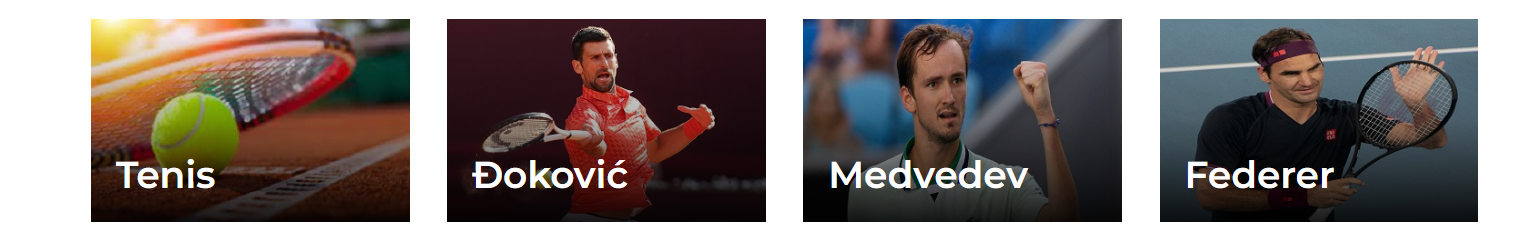
Мени је смештен у елемент <div> којем је придружена класа top-bar, дефинисана у *CSS* документу style.css. Конструкцијом top-bar-a бирају се сви тагови <a> из елемента <div> са класом top-bar. На овај начин се навигационом менију дефинишу позиција контакта, имена и презимена, броја телефона, линка о мени.





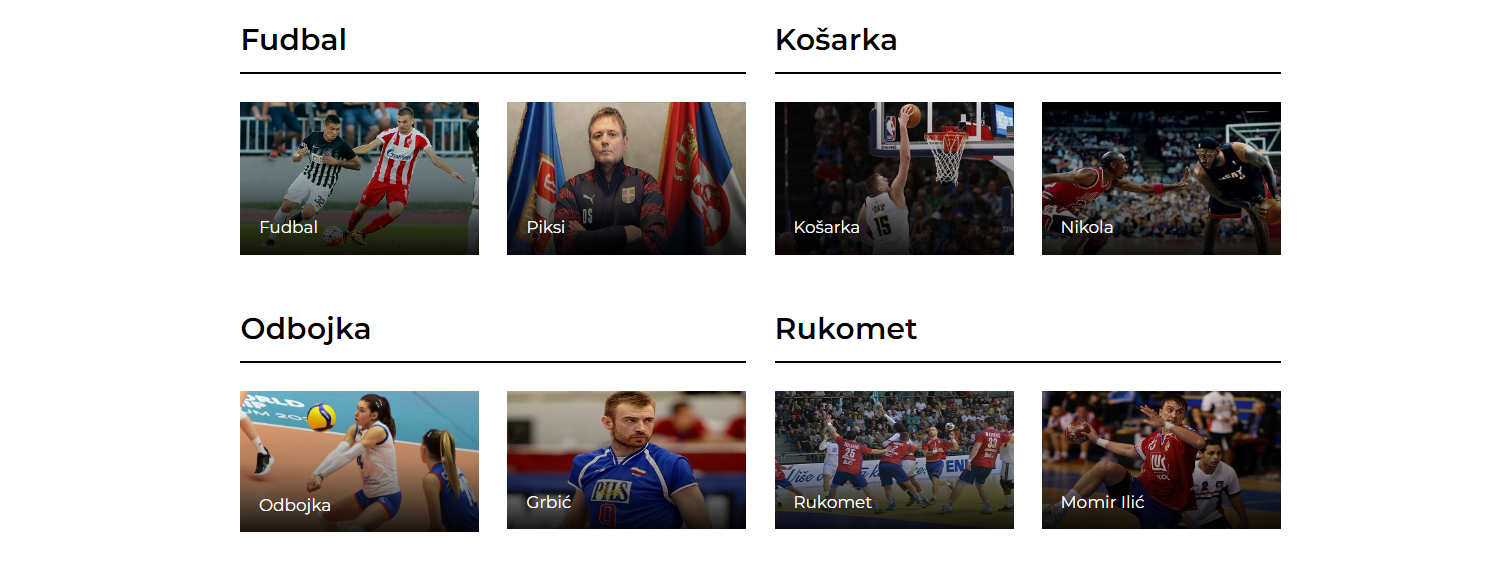
Слика 5 – Лого Спортски портал

Слика 6 – Главни мени Спортски портал



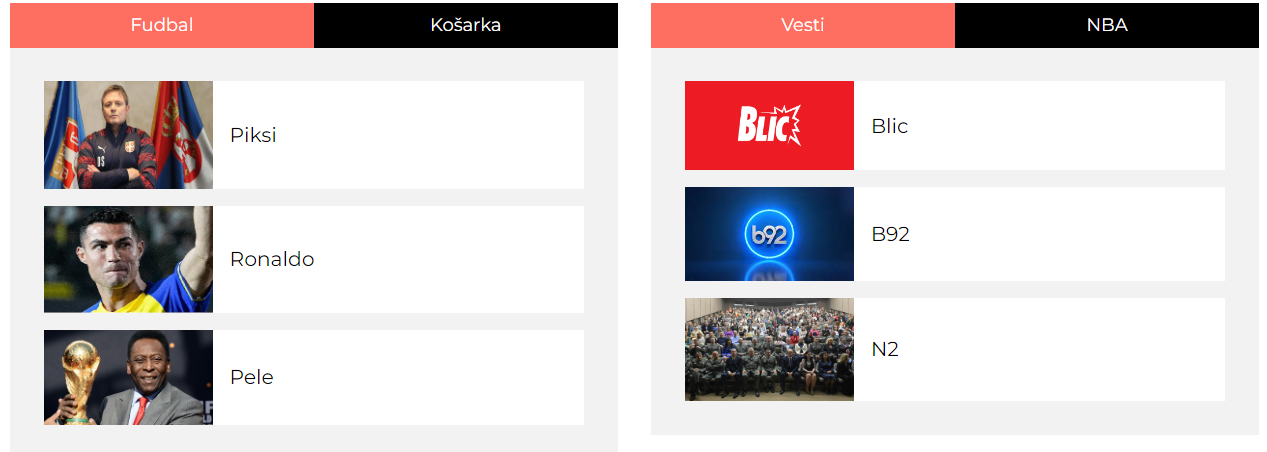
Слика 7 – *Тоp news* Спортски портал



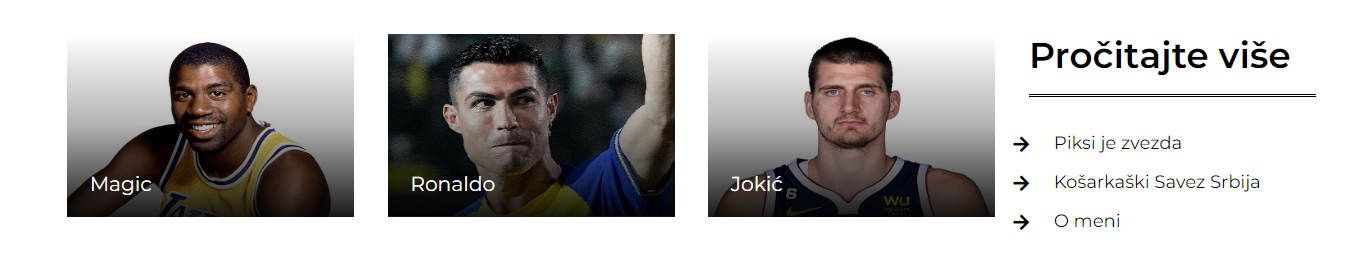


Слика 8 – *Cat news* Спортски портал





Слика 9 – *Tab news* Спортски портал



Слика 10 – *Main news* Спортски портал

### Футер

Футер садржи линкове за друштвене мреже, корисне линкове, брзе линкове, *newsletter* и тзв. *copyright* сегмент. Садржај футера стилизован је класом footer-menu, footer-bottom и посебним тагом <footer>.

Линкови за друштвене мреже су направљени по принципу употребе јавне библиотеке ***font-awesome*** са имплементацијом линкова ка тим друштвеним мрежама. Имплементиран је таг <link> унутар <head> тага на почетку странице.

*<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.10.0/css/all.min.css" rel="stylesheet">*

Слика 11 – Footer Спортски портал

Менији су креирани уз помоћ <ul> елемента, који представља неуређену листу (енгл. *unordered list*), у којој се елементи приказују без индексирања. Свака ставка у листи се представља елементом <li>, а у њему се, као у примарном менију, налази таг <a> са адресом спољне странице на коју линк води, али и текст који ће бити приказан кориснику.

*Copyright* сегмент футера је одређен одговарајућом класом col-md-6 copyright која дефинише начин приказивања (display), ширину (width), величину маргина (margin-top, margin-bottom), поравнање, величину и боју текста (text-align, font-size, color) a све то је одређено глобалном *CSS* поставком дефинисаном тагом <link> унутар <head> тага на почетку странице.

<link href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" rel="stylesheet">



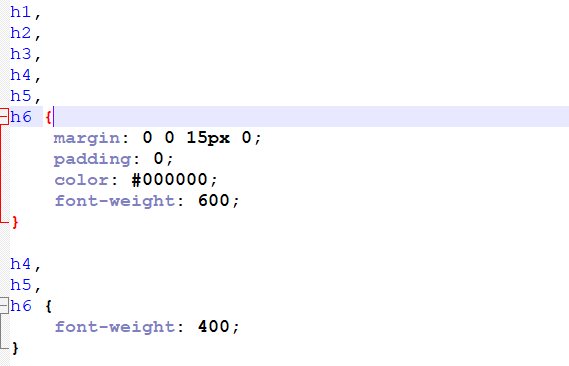




Слика 12 – *Copyright* сегмент футера Спортски портал

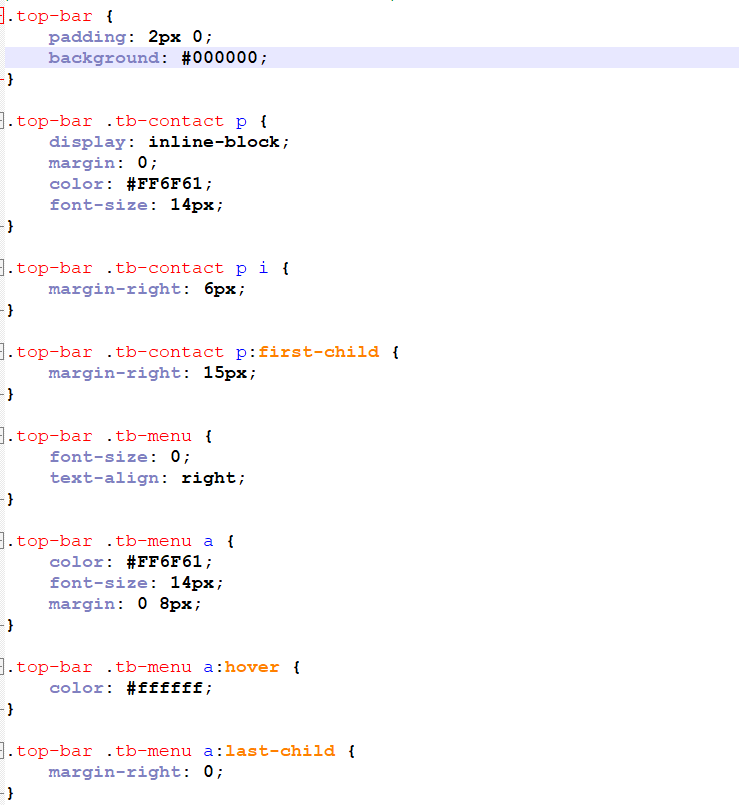
## Уређивање текста CSS

### Наслови и поднаслови

Наслови се формирају коришћењем тагова од <h1> до <h6>, у којима <h1> представља највећи, а <h6> најмањи наслов. Својства ових тагова се могу прецизније одредити у *CSS* датотеци.

Слика 13 – Промена величине наслова и поравнање текста по средини

### Топ Мени

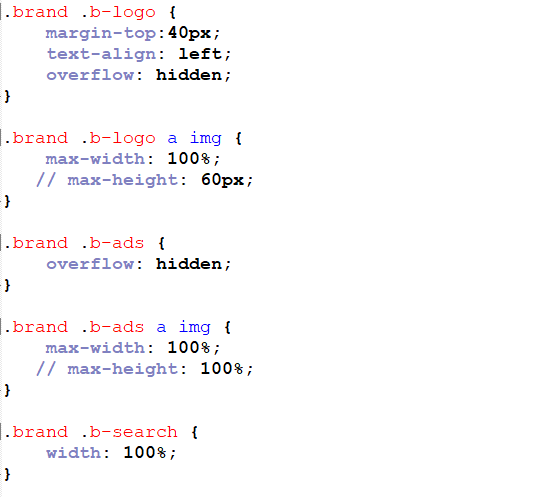
Топ мени се дефинише *CSS* селекторима приказаним на слици 14. На овај начин су дефинисане маргине и *padding*, величине фонтова и боја.

Слика 14 – Топ мени CSS

### Logo

Топ мени се дефинише *CSS* селекторима приказаним на слици 15. На овај начин су дефинисане маргине и ширина елемената. Својство overflow одређује да ли да се исече садржај или да се додају траке за померање када је садржај елемента превелик да стане у наведено подручје.

Својство overflow има следеће вредности:

* видљиво - подразумевано. Прелив није исечен. Садржај се приказује изван оквира елемента
* скривено - Преливање је исечено, а остатак садржаја ће бити невидљив
* померање - Преливање је исечено, а трака за померање је додата да бисте видели остатак садржаја
* ауто - Слично померању, али додаје траке за померање само када је то потребно

Слика 15 – Logo CSS

### Nav Bar

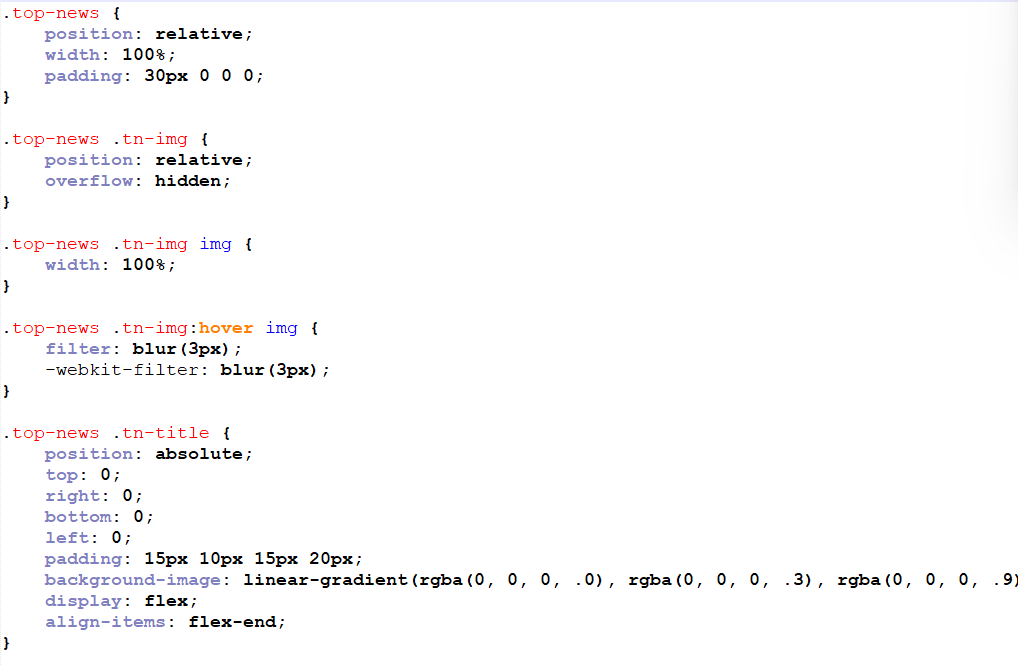
Nav Bar се дефинише *CSS* селекторима приказаним на слици 16. На овај начин су дефинисане маргине и положај елемената, *z*-индекс, *padding* и бокс подешавање. Својство положаја одређује тип методе позиционирања која се користи за елемент.

Постоји пет различитих вредности положаја:

* статични
* релативни
* фиксни
* апсолутни
* лепљив

Слика 16 – Nav Bar CSS

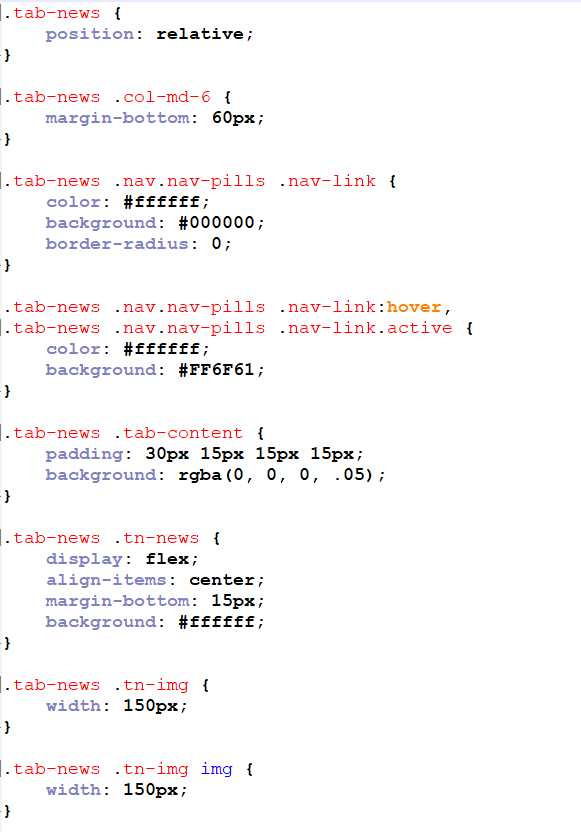
### Top News

Top News се дефинише *CSS* селекторима приказаним на слици 17. На овај начин су дефинисане маргине и положаји елемената, ширине елемената, *padding* и слика позадине.

Слика 17 – Top news CSS

### Tab News

Tab News се дефинише *CSS* селекторима приказаним на слици 18. На овај начин су дефинисане боја, позадина, маргине и положаји елемената, ширине елемената и *padding*.



Слика 18 – Таb news CSS

### Footer

Footer се дефинише *CSS* селекторима приказаним на слици 19. На овај начин су дефинисани боја позадинe, маргине и положаји елемената, padding и list - style.





Слика 19 – Footer CSS

# БМИ – индекс телесне масе

Индекс телесне масе (енг. Body Mass Index) је метода рачуњања ухрањености. Што је индекс више изнад оквира уредних вредности, то је већи ризик од обољевања од разних срчаних болести, дијабетеса и повишеног крвног притиска.

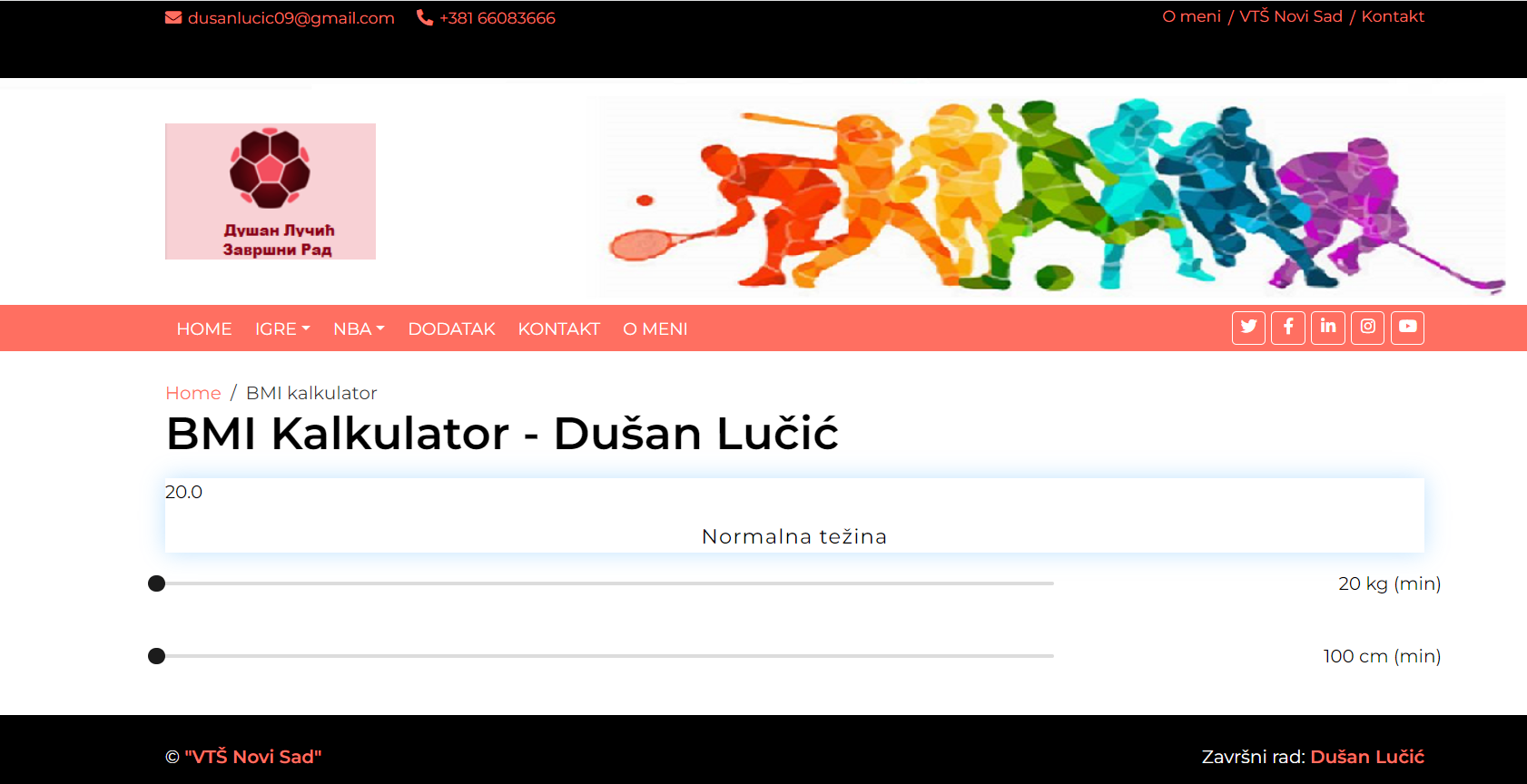
Рачуна се уз коришћење врло једноставне формуле:

Тежина (кг) : Висина² (м) = БМИ

Приказује однос масе и висине тела, међутим не узима у обзир телесну грађу, па је његова употреба ограничена. Он не може илустровати проценат масног ткива у односу на мишићну или коштану масу - што су основни критеријуми за проценат ухрањености. Појединци са великом телесном масом и високим БМИ индексом не могу се аутоматски категоризовати као гојазни. БМИ треба схватити као оквирну методу, будући да стварно здравствено стање особе треба оценити у ширем медицинском контексту.

Након разматрања појединачних елемената веб-странице, посебно ћемо се бавити анализом странице *БМИ у линку „Додатак“*, који је представљен као ставка у навигационом менију, а за циљ има да провери индекс телесне масе. Специфичност ове странице огледа се у томе што има посебну функционалност у односу на остале веб странице у Спортском порталу и израђена је у *JavaScript*-у са екстензиојм .js.

Заглавље почетне стране БМИ не разликује се значајно од заглавља већ описаних страница (види поглавље 3.3). Једина разлика је постојање два клизача: *Тежина*, којим се задају вредности тежине и клизача *Висина* којим се задају вредности висине корисника.



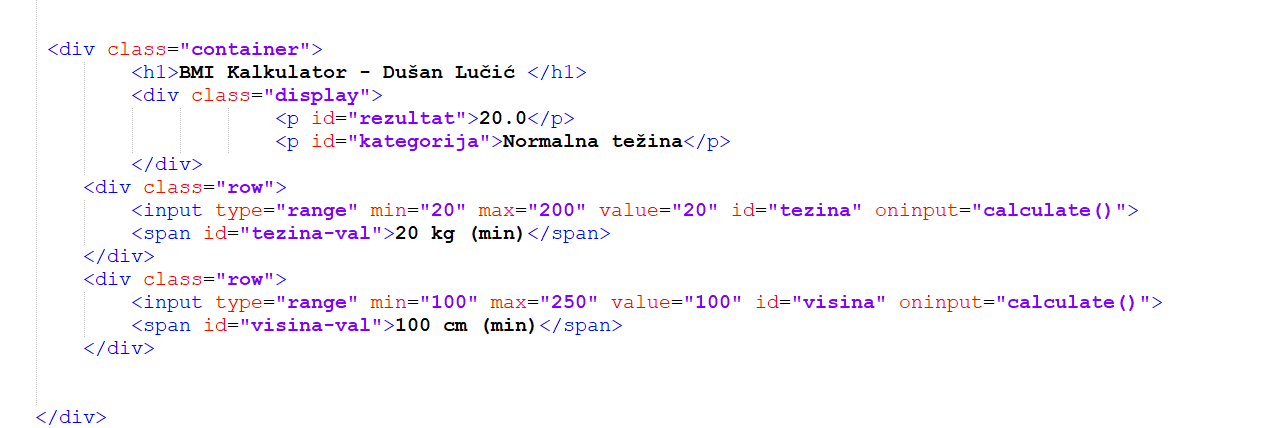
Слика 20 – БМИ

## Фунционалност програма БМИ

Кликом на дугме *Додатак*, из навигационог менија прелази се на страницу dodatak.html. На страници dodatak.html приказује се примарни навигациони мени, статусна трака, као и програм БМИ. Наведени елементи распоређени су коришћењем различитих flex својстава, преко посебних класа које су додељене одговарајућим *HTML* елементима (који су формирани на већ описане начине).

Програм урађен у *JavaScript*-у који је имплементиран у оквиру HTML тага, налази се између <script > тагова и формира приказ програма за израчунавање индекса тежине који је значајан за спортисте али и за обичне људе који желе да прате своју тежину и функционисање свог тела.

### Класа Container



Слика 21 – Класа Container

Класа *Container* садржи у себи класу *Display* која се састоји од два параграфа који имају свој идентификатор *ID*= “rezultat” i *ID*= “Normalna težina”.

Поред наведене класе, следећи одељак садржи два *div* тага са класама *row* унутар којих су дефинисани клизачи type = “range” са својим атрибутима мин. и макс, вредностима и окидачима *oninput="calculate()"* где се позива функција calculate().



Слика 22 – Приказ класе Container

### Функција calculate()

Функција calculate() налази се између тагова <script type="text/javascript"> </script> унутар којих је смештена програмска функција која је заправо главна функционалност БМИ програма. Као што је горе наведено, када корисник помери клизач активира се догађај *oninput="calculate()"* који директно позива функцију.

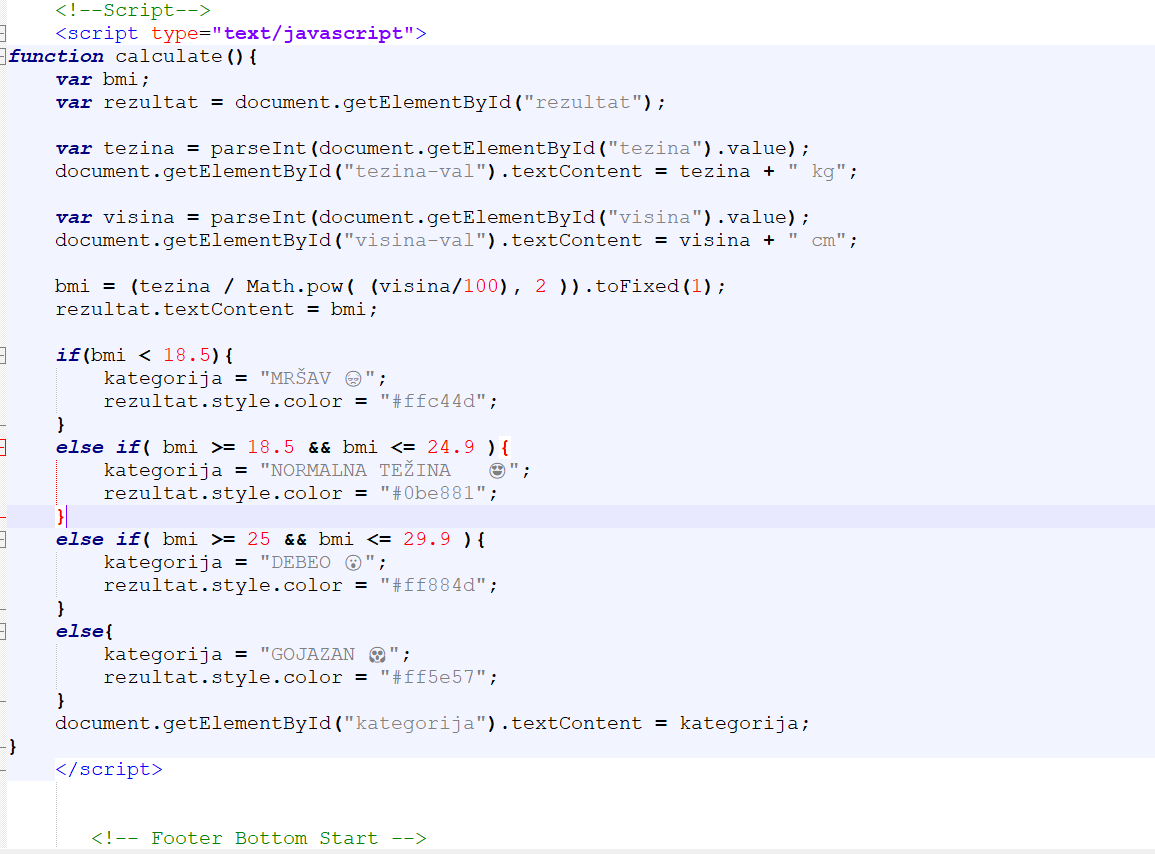
У самој функцији су дефинисане четири променљиве: *бми*, *резултат*, *тежина* и *висина*.

Променљива *rezultat* користи функцију getElementById("rezultat"). Овом функцијом се кроз *JavaScript* проналази елемент *HTML*-а који има конкретну идентификациону ознаку и примењује на њу конкретне ознаке и приказује их. Поред тога, на цео контент се додаје и стринг *" kg"* кроз следећу структуру *JavaScript*-a *document.getElementById("tezina-val").textContent = tezina + " kg".*

У променњивој *бми*  се додаје прорачун индекса тежине по следећој формули:

*bmi = (tezina / Math.pow( (visina/100), 2 )).toFixed(1);*

*rezultat.textContent = bmi;*



Слика 23 – класа Calculate()

### Приказ резултата

Кретањем клизача се рефрешује приказ у *HTML* таговима који су обележени одређеним идентификатором *ID*. Приказ резултата се мења у зависности од резултата. Резултат који је нормалан биће приказан зеленом бојом уз одређени емотикон, док ће се резултат који није у границама нормале приказати црвеном бојом и другим емотиконом.

***if(bmi < 18.5){***

***kategorija = "MRŠAV 😒";***

***rezultat.style.color = "#ffc44d";***

***}***

***else if( bmi >= 18.5 && bmi <= 24.9 ){***

***kategorija = "NORMALNA TEŽINA 😍";***

***rezultat.style.color = "#0be881";***

***}***

***else if( bmi >= 25 && bmi <= 29.9 ){***

***kategorija = "DEBEO 😮";***

***rezultat.style.color = "#ff884d";***

***}***

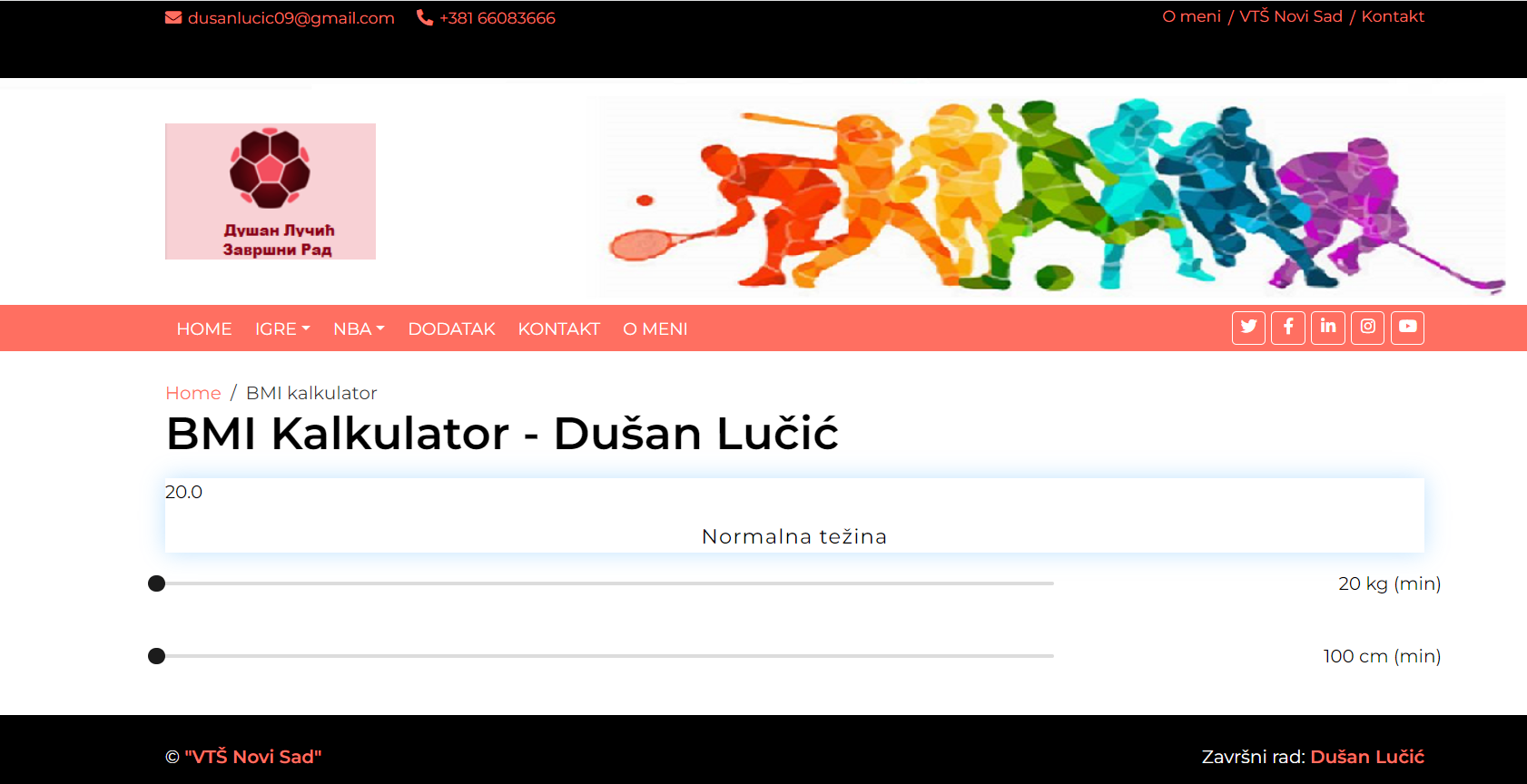
***else{***

***kategorija = "GOJAZAN 😱";***

***rezultat.style.color = "#ff5e57";***

***}***

***document.getElementById("kategorija").textContent = kategorija;***



Слика 24 – Приказ класе Calculate()

# Закључак

*HTML*, *CSS* и *JavaScript* су три кључна језика за израду веб-страница и веб-апликација. Сваки од њих има своју улогу и функционалност.

Комбинација ова три језика омогућава израду модерних, интерактивних и атрактивних веб-страница и апликација. *HTML* пружа структуру, *CSS* додаје стил, док *JavaScript* доноси функционалност. Ови језици су међусобно повезани и често се користе заједно како би се постигао жељени резултат.

Учење *HTML*-а, *CSS*-а и *JavaScript*-а омогућује вам стварање сопствених веб-пројеката, прилагођавање изгледа и понашања веб-страница као и интеракцију са корисницима. Наставак учења и пракса ових језика отвара врата многим могућностима у свету веб-развоја.

Неминовни развој нове технологије сигурно ће повећати потражњу за напреднијим и функционалнијим веб-страницама. У исто време, напредак технологије ће омогућити лакше и брже имплементирање сложенијих функција и самим тим смањити време потребно за израду веб-странице.

Технолошка револуција променила је начин на који приступамо, делимо и користимо информације. Брза размена информација пружа невероватне могућности за економски развој, иновације, комуникацију и ефикасност у свим аспектима нашег живота.

Овај завршни рад је потенцијално добар пример како може да се развије један портал о спорту конкретно али и добра основа која омогућава даљи рад на пројекту.

# Литература

1. https://www.w3schools.com/css/css\_positioning.asp
2. Array.prototype.forEach() – *JavaScript*, Mozilla
3. https://www.w3schools.com/css/css\_overflow.asp
4. [https://developer.mozilla.org/en US/docs/Web/JavaScript/Reference/Global\_Objects/Array/forEach](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/forEach) (април 2023)
5. *CSS* Grid Layout, Mozilla Developer Network, <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout> (април 2023)
6. https://www.stetoskop.info/medicinski-kalkulatori/indeks-telesne-mase
7. *CSS* Својства која дефинишу границе елемента, Петља, <https://petlja.org/kurs/488/7/6316> (април 2023)
8. *CSS* стилови, Петља, <https://petlja.org/kurs/488/6/6310> (април 2023)
9. Element: innerHTML property, Mozilla Developer Network, <https://developer.mozilla.org/en-US/docs/Web/API/Element/innerHTML> (април 2023)
10. *HTML* link tag, W3Schools, <https://www.w3schools.com/tags/tag_link.asp> (април 2023)
11. *CSS*, Википедија, <https://sr.wikipedia.org/wiki/HTML> (април 2023)
12. JSON.parse(), Mozilla Developer Network, <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/JSON/parse> (април 2023)
13. let – *JavaScript*, Mozilla Developer Network, <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/let> (април 2023)
14. Margin, Mozilla Developer Network, <https://developer.mozilla.org/en-US/docs/Web/CSS/margin> (април 2023)

Датум предаје завршног рада: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Комисија:

Председник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Испитивач \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члан \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Коментар:

Датум одбране: \_\_\_\_\_\_\_\_\_\_\_\_\_ Оцена\_\_\_\_\_\_\_\_\_\_ (\_\_\_)