

Универзитет “Св. Кирил и Методиј” – Скопје
Факултет за информатички науки и компјутерско инженерство



Семинарски труд по предметот
Основи на Веб дизајн
на тема:

*“Изработка на single page web site за
студентски активности со Angular JS слично
со ФИНКИ “*

Ментор:

Проф. Иван Чорбев

Изработиле:

*Стефан Куновски 151048 и
Димитри Дојчиновски 151062*

Содржина:

Апстракт	3 стр.
Вовед	3 стр.
Материјал и методи за формирање на бизнис моделот	7 стр.
Формирање на single page web сајт	16 стр.
Резултати	17 стр.
Дискусија и Заклучок	20 стр.
Литература	21 стр.

Абстракт

Веб дизајнот е релативно нова индустрија која благодарение на интернетот стана популарна во последните 10 години како дигитален медиум кој стана значаен дел од животот на современиот човек. Веб интеракцијата се користи за информирање, комуникација, маркетинг и продажба, социјален живот итн.

Поради се поголемото значење на веб дизајнот во сите области на современото живеење оваа индустрија е во постојан пораст, а евидентна е потребата од веб дизајнери кои би ги користеле најновите ИТ технологии во создавањето на атрактивни и динамични веб страни.

Вовед

HTML (HyperText Markup Language) е јазикот кој се користи за создавање на веб документи, т.е дефинирање на структурата, содржината и функцијата на еден HTML документ. За разлика од процедуралните програмски јазици како Ц, Ц++ или Јава, HTML е означивачки јазик кој го одредува форматот на текстот што се прикажува во веб пребарувачите како што се Mozilla Firefox, Google Chrome и др. Така, HTML нас ни служи за одредување на наслови, текстови, слики, хипертекст врски, а исто така овозможува тие исти елементи во помала или поголема мерка стилски или графички да се модифицираат.

HTML е јазик базиран врз SMGL (Standard Generalized Markup Language), општо прифатен стандарден јазик за селектирање. Првичните верзии на HTML (HTML+ во 1994 и HTML 2,0 во 1995) ,создадени врз трудовите на Tim Berners-Lee од 1991 година, и интернетот не требало да станат визуелна средина, но со појавувањето на визуелните интернет пребарувачи, интернетот го направило своето, меѓутоа како последица на брзиот и неконтролиран развој, се појавија проблеми околу меѓусебната неподдржаност на различните пребарувачи на различни платформи (вакви проблеми се присутни и денес но стануваат се помалку нагласени), особено Mosaic Netscape и Microsoft Internet Explorer, кои не дочекаа стандарди за интернет програмирање. Во 1996, ново создадената W3C става значаен удел во развојот на веб програмирањето и ги објавува своите препорачани стандарди HTML 3.2. Тоа е слика на сите HTML елементи кои се често користени во тоа време и вклучува многу презентационски екстензии на HTML-от кои беа како резултат на Netscape/IE расправијата и отсуството од стилски страници. HTML 4,0 (1998) и HTML 4,01 (малата ревизија објавена во 1999) имаа цел да го стават HTML-от назад на вистинскиот пат со нагласување на поделбата на структура и презентација и подобрување пристапноста. Сите прашања околу презентацијата(видливиот дел на една веб страница) се предадени на новоформираните Каскадни стилски страници (Cascading Style Sheets- CSS).

Речиси во исто време кога HTML 4.01 беше во развоја, луѓето од W3C станаа свесни дека еден ограничен јазик за означување нема да биде доволен за опишување на сите видови информации (хемиска нотација, математички равенки, презентација на мултимедија, финансиски информации итн.) кои можат да се споделат преку Web. Нивното решение било: XML (eXtensible Markup Language), метајазик за создавање на јазика за означување. XML е фактички поедноставување на SGML, бидејќи SGML се докажа дека може да биде покомплексен од тоа што на интернетот му е потребно.

W3C имаше визија за XML базиран Web со многу јазици за означивање работејќи заедно, дури и во рамките на еден документ. Се разбира, да се овозможи ова треба секој документ претпазливо да се пишат со јазиците за маркирање и строго да се почитува XML синтаксата, но тука има потенцијал да дојдте до конфузија. Нивниот прв чекор беше да се препработи HTML според правилата на XML, за да може да биде компактибилен на другите платформи. Резултатот е XHTML 1.0, кој речиси е идентичен на HTML 4.01, кои ги споделуваат истите елементи и атрибути, но со построги барања за тоа како маркирањето да се направи.

HTML 4.01 заедно со XHTML 1.0, неговиот построг XML базиран брат, станаа камен-темелник на Web стандардите. Тие сеуште се најтемелни и постојано подржани стандарди (иако HTML 5 брзо ги замена овие стандарди). Но, W3C не запре тука. Со визија за XML базиран Web, тие започнаа да работат на XHTML 2.0, уште еден обид да направат работите да работат подобро од HTML 4.01 што работеле. Проблемот беше што не е компатибилен со стаите стандарди и интернет пребарувачи. Пишувањето и процесите за одобрување се влечеше со години со никаква имплементација на пребарувач. Без имплементација на пребарувач, HTML 2.0 заглави.

Во меѓувреме, во 2004 година, членовите на Apple, Mozilla и Opera формираа групација наречена Web Hypertext Application Technology Working Group (WHATWG, whatwg.org). Целта на WHATWG е за продолжување на развојот на HTML за да се задоволат новите потреби на начин кој е во согласност со набљудување на real-world практики и однесувањето на интернет пребарувачите. Нивните првобитни документи, Web Applications 1.0 и Web Forms 1.0, се фклопија во HTML 5 како нов стандард под водство на Ian Hickson во 2014 година.

HTML користи специјални делови од програмски јазик наречени тагови со цел да му пренеси на пребарувачот како една веб страна треба да личи. Таговите можат да бидат во пар (отворачки и затворачки) или празни (самозатворачки). Побитни тагови во HTML се:

- `<!DOCTYPE>` Таг за тип на документ
- `<head>...</head>` таг кој содржи информации за документот
- `<body>...</body>` таг за дефинирање на телото на документот
- `<style>...</style>` таг за стилизирање на некој елемент или цел HTML код
- `<script>...</script>` таг за воведување на скрипта
- `<div>...</div>` див таг кој додава нов сегмент во веб страницата
- `<p>...</p>` таг за параграф
- `<a>...` таг за активен линк
- `` таг за внесвање на слика
- `...` `<i>...</i>` `<s>...</s>` `<u>...</u>` тагови за болд, италик, прецртан текст и подвлечен текст
- `<h1>...</h1>` ... `<h6>...</h6>` тагови за големина на фонтот на наслово
- `</br>` `</hr>` тагови за нов ред и хоризонтална линија (соодветно)
- `<table>...</table>` таг за креирање на табела
- `<tr>...</tr>` `<td>...</td>` тагови за ред во табела и таг за келија од табелата
- `<!-- ... -->` таг за поставување на коментар
- `<class>` таг кој класифицира повеќе елементи со иста идентификација
- `<id>` таг кој именува уникатен елемент во HTML кодот
- `<title>` таг кој го одредува името на документот во веб пребарувачот

HTML документите се зачувуваат со .html екстензија на крајот од името на документот.

Каскадни стил страници (Cascading Style Sheets- CSS) е стандард на W3C за дефинирање на надворешниот изглед на документот напишан во HTML, се однесува на начинот на кој документот се прикажува на корисникот, без разлика дали на екран на компјутер, екран на мобилен телефон, печатена хартија или страницата да е прочитана од screen reader. CSS е посебен јазик со своја синтакса. Таа синтакса има голема моќ во тоа како да се дизајнира една веб страна. Кога се користи во својот полн потенцијал, CSS е стабилна и моќна алатка за дизајнирање. CSS се додаваат на документот на два начини:

- Екстерна стил страница е посебен, текст документ кој содржи голем број на правила за стил, тој се именува со .css суфикс, при што овај документ може да се поврзе со повеќе HTML документи. Ова е најможен и префериран метод за прикачување на стил страница на содржина.
- Вметната стил страница односно со користење на style елементот вметнат помеѓу “<” и “>” и се вметнува во head делот на HTML документот.

```
<head>
  <title>Required document title here</title>
  <style>
    /* style rules go here */
  </style>
</head>
```

Ако во HTML документот таговите имаа главна улога, тука селекторите се главни за модифицирање на CSS-от. Постојат многу различни видови на CSS селектори кои ќе ни овозможат да влијаеме на одредени елементи од HTML документот.

- *{} универзален селектор, ги селектира сите елементи во документот
- h1,h2,h3 {} тип селектор, ги обележува сите тагови од истиот тип
- .class-selector{} селектор на класа
- #id-selector{} селектор на id таг
- Li>a{} селектор на a елемент кое е дете на li
- P a {}селектор на a елемент кој е внатрешен елемент на P
- h1+p{} селектор на првиот p елемент во h1
- h1~p{} селектор ако има два p елементи кои се роднини на h1 елемент

JavaScript е програмски јазик кој додава интерактивноста и сопствени однесување на нашите веб страници. Тоа е client-side јазик за скриптирање, што значи работи на машината на корисникот, а не на сервер, како и другите веб програмирање јазици како што се PHP и Руби. JavaScript е воведена во 1995 година како начин за да додадете програми во веб-страници во веб пребарувачот Netscape Navigator. Јазикот оттогаш беше усвоен од страна на сите други големи веб шребарувачи. Важно е да се напомене дека Javascript нема никаква

поврзаност со програмскиот јазик Јава. Сличното име беше донесено после спогодба помеѓу девелоперите на JavaScript и Sun microsystems кои се девелопери на програмскиот јазик Јава. Кога JavaScript бил воведен, јазикот Јава е изложен на голем маркетинг и се стекнување со голема популарност.

По нејзиното донесување надвор од Netscape, документи со стандарди беа веднаж донесени со цел да се опише начинот на кој јазикот Javascript треба да работи за да биде сигурен дека различните делови од софтверот кој тврди дека поддржува JavaScript всушност дека се зборуваме за ист јазик. Ова се нарекува EC-MAScript стандард (ECMA Меѓународната организација што го направи за стандардизација). Во пракса, термините ECMAScript и Javascript може да се користат наизменично, тие се две имиња за ист јазик.

Во дигиталниот свет веб страниците станаа значајна алатка за интеракција преку која големите и мали компании, групи, поединци и непрофитни организации комуницираат со своите целни групи на World Wide Web. Целта на оваа семинарска работа е да прикаже начинот и принципите на формирање на single page web site за студентски активности со Angular JS слично со ФИНКИ.

Во семинарскиот труд по предметот “Основи на Веб Дизајн” е опфатена постапката на формирање на Веб страница за студентски активности со користење на HTML и каскадни стилови. Притоа се проследени сите потребни чекори за формирање на страната со користење на WebStorm како моќна платформа за кодирање и асистенција во користењето на HTML, CSS како и AngularJS

Цел на трудот е да се прикажат дел од можностите во користењето на горенаведените јазици во дадената платформа, при што се користени ресурси (слики и текстови) од постоечката веб страна на Финки.

Методологија

<head> сегментот содржи мета таг со цел да се пишат кириличните знаци, линк за CSS документот и <script> таг за кои JavaScript библиотеки се користат.

```
<head>
  <meta charset="UTF-8">
  <link rel="stylesheet" type="text/css" href="style.css">
  <title>HOME</title>
  <script
src="//ajax.googleapis.com/ajax/libs/angularjs/1.3.3/angular.min.js"></script
>
  <script src="//cdnjs.cloudflare.com/ajax/libs/angular.js/1.3.3/angular-
route.min.js"></script>
</head>
```

Телото на страницата односно <body> делот се состои од 7 елементи:

1. Навигациска лента- содржи 4 <a> линк тагови кои ги нагласува 4-те главни делови на веб страната и нуди брз пристап до нив со кликување на насловите.

```
<ul id="nav1">
  <a ng-click="scrollTo('eden')"><li>Студентски активности</li></a>
  <a ng-click="scrollTo('dva')"><li>Настани</li></a>
  <a ng-click="scrollTo('skola')"><li>Летна школа</li></a>
  <a ng-click="scrollTo('cetiri')"><li>Контакт</li></a>
</ul>
```

2. Лого и мото на организацијата

```
<div id="nesto">
  
  "Да изградиме нова ИТ иднина"
</div>
```

3.Студентски активности – во овај дел се прикажуваат најсвежите новости и активности поврзани со студентскиот живот на студентите од факултетот ФИНКИ. 3-те новости ротираат на секои 5 секунди и им се прикажува краток опис на активността.

```
<div id="eden">
  <div id="naslov">
    СТУДЕНТСКИ АКТИВНОСТИ
  </div>
  <div id="slika1">
    

    <h2 id="n1"><span>TEDx настан на ФИНКИ <br/>"Дали технологијата
може да ги реши најголемите човекови предизвици?"</span></h2>
    <h2 id="n2"><span>Студенти од ФИНКИ победници на Codefest 2016
<br/> освоено прво место на овогодинашниот 32-часовен маратон во програмирање
Codefest Marathon под мотото "Challenge Your Limits"</span></h2>
    <h2 id="n3"><span>Студенти на ФИНКИ победници на NASA Space App
Challenge</span></h2>

    
  </div>
</div>
<div id="dva">
```

4.Настани-овај дел содржи 3 послени настани организирани од факултетот каде со поместување на курсерот над сликата на еден настан излегува краток опис и при тоа може со кликување на него да се оди на линкот на оригиналната страница за повеќе детали за настанот.


```

<div id="q">НАСТАНИ</div>
<a
href="http://finki.ukim.mk/mk/content/%D0%B1%D0%B5%D0%B7%D0%B1%D0%B5%D0%B4%D0
%BD%D0%BE%D1%81%D0%BD%D0%BE-
%D1%82%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%B0%D1%9A%D0%B5-%D0%BD%D0%B0-
%D1%84%D0%B8%D0%BD%D0%BA%D0%B8" target="_blank"></a>
<a
href="http://finki.ukim.mk/mk/content/%D0%BE%D1%82%D0%B2%D0%BE%D1%80%D0%B5%D0
%BD-%D0%B4%D0%B5%D0%BD-%D0%BD%D0%B0-%D1%84%D0%B8%D0%BD%D0%BA%D0%B8"
target="_blank"></a>
<a href="http://finki.ukim.mk/mk/content/%D0%B4%D0%B5%D0%BD-
%D0%BD%D0%B0-%D0%B5%D0%B2%D1%80%D0%BE%D0%BF%D0%B0" target="_blank"></a>
<a
href="http://finki.ukim.mk/mk/content/%D0%B1%D0%B5%D0%B7%D0%B1%D0%B5%D0%B4%D0
%BD%D0%BE%D1%81%D0%BD%D0%BE-
%D1%82%D0%B5%D1%81%D1%82%D0%B8%D1%80%D0%B0%D1%9A%D0%B5-%D0%BD%D0%B0-
%D1%84%D0%B8%D0%BD%D0%BA%D0%B8" target="_blank"><h1 id="p1">
Безбедносно тестирање на ФИНКИ<br><br>Почитувани студенти, во
изминатите неколку дена беше изведена безбедносна проверка за свеста и
отпорноста на студентите на ФИНКИ за phishing напади (или таканаречено мрежно
рибарчење).
</h1></a>
<a
href="http://finki.ukim.mk/mk/content/%D0%BE%D1%82%D0%B2%D0%BE%D1%80%D0%B5%D0
%BD-%D0%B4%D0%B5%D0%BD-%D0%BD%D0%B0-%D1%84%D0%B8%D0%BD%D0%BA%D0%B8"
target="_blank"><h1 id="p2">
Отворени денови на ФИНКИ<br><br>На 07 мај 2016 година (сабота) со
почеток во 11 часот, Факултетот за информатички науки и компјутерско
инженерство организира Отворен ден, на кој идните студенти ќе имаат уште една
можност директно да се запознаат со можностите за студирање на Факултетот.
</h1></a>
<a href="http://finki.ukim.mk/mk/content/%D0%B4%D0%B5%D0%BD-
%D0%BD%D0%B0-%D0%B5%D0%B2%D1%80%D0%BE%D0%BF%D0%B0" target="_blank"><h1
id="p3">
Ден на Европа<br><br>По повод денот на Европа, 9ти мај,
Европската унија на институти за култура (EUNIC), со поддршка на
Секретаријатот за европски прашања и Делегацијата на ЕУ, организира настан
"Младите и мобилноста", во понеделник, 9ти мај, 2016 год., од 11 до 14h часот
во ЕУ-МК инфо центарот (Плоштад Македонија).
</h1></a>
</div>

```

5. Летна школа- е делот на веб страната каде се нудат курсеви од факултетот во веб дизајн, каде со селектирање на насовите може да се виде краток опис на курсот и на крајот има формулар да се пополни дококу некој е заинтересиран да присуствува на некој од курсевите.

```

<h1 id="skola">ЛЕТНА ШКОЛА ЗА ПРОГРАМИРАЊЕ</h1>
<ul id="nav2">
<a href="#/"><li>HTML и CSS</li></a>
<a href="#/javas"><li>JavaScript</li></a>

```

```

        <a href="#/php"><li>PHP со MySQL</li></a>
        <a href="#/prijava"><li>Пријава</li></a>
    </ul>
    
    <div ng-view id="pri"></div>

```

6. Контакт-општи информации за контакт како маил, фејсбук страна и линк до други страници каде може да се воспостави контакт со факултетот како и мапа за каде се наоѓа факултетот.

```

<div id="cetiri">
    <div id="contact">
        Контакт:<br>
        contact@finki.ukim.mk
    </div>
    <a href="http://finki.ukim.mk/" target="_blank">
        
    </a>
    <a href="https://www.facebook.com/FINKI.ukim.mk/" target="_blank">
        
    </a>
    <a href="https://twitter.com/FINKIedu" target="_blank">
        
    </a>
    <a href="https://www.linkedin.com/company/faculty-of-computer-
science-and-engineering---skopje" target="_blank">
        
    </a>
    <a href="https://www.youtube.com/user/FINKIedu" target="_blank">
        
    </a>
    <a href="https://www.flickr.com/photos/finki-cse/" target="_blank">
        
    </a>
    <a ng-click="scrollTo('nav1')">
        
    </a>

    <div id="map">
        <script
src='https://maps.googleapis.com/maps/api/js?key=AIzaSyBIZxGn97jRp30h_THlngIb
KV5zqbGX7Cs&v=3.exp'></script>
        <div style='overflow:hidden;height:250px;width:400px;'><div
id='gmap_canvas' style='height:250px;width:400px;'></div><div><small><a
href="http://embedgooglemaps.com">embed google maps
</a></small></div><div><small><a href="https://termsfusegenerator.net">terms
of use generator</a></small></div>
        <style>#gmap_canvas img{max-
width:none!important;background:none!important}</style></div>
        <script type='text/javascript'>function init_map(){var myOptions
= {zoom:15,center:new
google.maps.LatLng(42.0047685,21.409779999999995),mapTypeId:
google.maps.MapTypeId.ROADMAP};map = new
google.maps.Map(document.getElementById('gmap_canvas'), myOptions);marker =
new google.maps.Marker({map: map,position: new

```

```

google.maps.LatLng(42.0047685,21.409779999999955));infowindow = new
google.maps.InfoWindow({content:'<strong>ФИНКИ</strong><br>Pyреp Бoшкoвиќ 16,
Skopje, Macedonia<br>'});google.maps.event.addListener(marker, 'click',
function(){infowindow.open(map,marker);});infowindow.open(map,marker);}google
.maps.event.addDomListener(window, 'load', init_map);</script>
</div>
</div>

```

8. Футер –информации за тоа кои ја изработија страницата

```

<div id="foot">
    Стефан Куновски и Димитри Дојчиновски <br>
    &copy; ФИНИКИ 2016
</div>

</div>

<script src="js.js" defer></script>
</body>
</html>

```

Single page веб страните се одлично решение за одредени проекти кога целата содржина е вметната на една страна, а се особено корисни кога се работи за пристап од мобилни уреди. За изготвувањето на веб страната беше неопходно познавање на основите на веб дизајнот, алатките и платформите за работа.

CSS делот е напишан благодарение на стекнатото знаење од предметот Основи на Веб дизајн и особена благодарност до асистентите кои ни имаат кажано одредени трикови при изработка на каков било веб проект, повеќето од синтаксата е користена од предавањата и аудиториските предавања, за одредени проблеми ние се послужиравме со пребарување на решенија онлајн на страниците w3c.com и stacoverflow.com.

```

<body> тагот т.е целата страница има background: white;, div тагот на
насловот, настаните и контакт има нијанса на сиво background-color: #fbfbf9;,
div тагот на Студентски активности има background-color: lightgray;
body{
    background: white;
    margin: 0;
    font-family: "Helvetica";
}
#eden{
    width: 100%;
    padding-bottom: 3%;
    background-color: lightgray;
}
#dva{
    padding-bottom: 2.5%;
    background-color: #fbfbf9;
}

```

```

#nesto{
  padding-top: 2%;
  background-color: #fbfbf9;
  color: red;
  font-size: 40px;
  text-align: center;
  padding-bottom: 2%;
  font-style: italic;
}
#cetiri{
  background-color: #fbfbf9;
  padding-bottom: 2.5%;
}

```

:hover е селектор кој има улога кога со курсеорт од глуфцето ќе се помини над некој елемент кој содржи :hover селектор да се селектира тој елемент. Во нашиот пример навигационата лента има некоја потемна нијанса на сива боја (#333) и кога ќе поминиме над неа таа ќе се селектира и ќе има црна боја.

```

#nav1 li:hover{
  background-color: black;
}

```

Не само за обојување може да послужи да се зголеми елементот кога ќе се помини над него со курсерот.

```

#fb{
  position: absolute;
  width: 2%;
  margin-left: 24%;
  margin-top: 1.5%;
}
#fb:hover{
  margin-top: 1%;
  width: 3%;
  margin-left: 23.5%;
}

```

Viewport вредностите ги користиме за пропорционалност на фонтот со страната односно за контролирање на големината на фонтот според големината на страната.

Vh 1/100 -viewport height

vw 1/100-viewport width

vmax 1/100 -ја зема поголемата вредност

vmin 1/100 -ја зема помалата вредност

во зависност од тоа дали страната е посирока или подолга

```
#p1, #p2, #p3{
  display: none;
  margin-top: -26%;
  position: absolute;
  width: 22.5%;
  left: 8%;
  color: white;
  font: bold 1.25vmax/1.5vmax Helvetica, Sans-Serif;
  letter-spacing: -1px;
  background: rgb(0, 0, 0); /* fallback color */
  background: rgba(0, 80, 255, 0.5);
  text-align: center;
}
```

Arrow to up page е често користен за брзо враќање во горниот дел на страната бидејќи `position:fixed` бидејќи со тоа тој се движи паралелно со `slide` линијата на страницата.

```
#arrow{
  position: fixed;
  top: 80%;
  right: 0;
  width: 2.5%;
  cursor: pointer;
}
```

JSS

```
var app=angular.module('single-page-app',['ngRoute']);
app.config(function($routeProvider) {
  $routeProvider
    .when('/', {
      templateUrl: 'home.html'
    })
    .when('/javas', {
      templateUrl: 'javas.html'
    })
    .when('/php', {
      templateUrl: 'php.html'
    })
    .when('/prijava', {
      templateUrl: 'prijava.html'
    })
    .when('/pusteno', {
      templateUrl: 'pusteno.html',
      controller: 'ctrl'
    })
  });
```

Ова е една од главните функции во AngularJS и се користи за лоадирање на темплејти. Принципот на работење е сличен со линкови, меѓутоа за разлика од

линковите не се отвора нова страна, туку на самата страна само доаѓа до промена на темплејт. Промената на темплејтите се прави преку навиџацискиот бар во сегментот за Летна школа, и притоа `<div ng-view>` го ладира соодветниот темплејт, кој сам по себе е HTML документ.

```
app.controller('cfgController',function($scope, $location, $anchorScroll){

    $scope.scrollTo = function(id) {
        var old = $location.hash();
        $location.hash(id);
        $anchorScroll();
        $location.hash(old);
    }
});
```

Поради начинот на кој што работат темплејтите во AngularJS, се појавува проблем при примена на анчор линковите. Затоа е применета оваа функција, која ја има истата функционалност. Се користи за брзо шетање низ страната преку навиџацискиот бар на врвот од страната, и стрелката од десната страна. Исто така специфично е `.location(old)`, која го задржува истиот темплејт кој веќе беше ладирани на страната.

```
document.getElementById("ss1").onmouseover = function() {mouseOver1()};
document.getElementById("ss1").onmouseout = function() {mouseOut1()};
document.getElementById("p1").onmouseover = function() {mouseOver1()};
document.getElementById("p1").onmouseout = function() {mouseOut1()};

function mouseOver1() {
    document.getElementById("p1").style.display = "block";
}

function mouseOut1() {
    document.getElementById("p1").style.display = "none";
}
```

Оваа функција е искористена за сликите во сегментот Настани. Начинот на работа е едноставен, кога со курсорот ќе се оди на некоја слика се појавува текст кој ја опишува таа слика. Важно е да се напомени дека е применета и кај текстовите што се појавуваат, бидејќи во случај кога ќе се појави текстот тој се наоѓа над сликата и веќе курсорот не се смета дека е над сликата што предизвикува текстот постојано да се прикажува и исчезнува.

```
function setheight() {
    var divHeight = document.getElementById('ss1').offsetHeight;
    document.getElementById('p1').style.height = (0.68 * divHeight) + 'px';
    document.getElementById('p2').style.height = (0.68 * divHeight) + 'px';
    document.getElementById('p3').style.height = (0.68 * divHeight) + 'px';
}
```

```
setInterval(setheight,1);
```

Оваа функција е искористена за динамичко поставување на висина на текст кутиите кои се појавуваат над сликите. Бидејќи големините на сликите се пропорционални со страната, потребно беше величините на текстовите да бидат пропорционални со сликите. Затоа оваа функција ја пресметува должината на сликата, и според тоа ја поставува должината на текстот.

```
function displayNextImage1() {
    x = (x === images.length - 1) ? 0 : x + 1;
    document.getElementById("s1").src = images[x];
}

function startTimer() {
    mt1 = setInterval(displayNextImage1, f);
    mt2 = setInterval(displayNextImage2, f);
    mt3 = setInterval(displayNextImage3, f);
}

function display1() {
    temp1=document.getElementById("s1").src;
    document.getElementById("s1").src = document.getElementById("s2").src;
    document.getElementById("s2").src = temp1;
    clearInterval(mt1);
    clearInterval(mt2);
    clearInterval(mt3);
    startTimer();
}
```

Овие функции се применети во сегментот за Студентски активности. Функцијата displayNextImage автоматски ги ротира сликите со помош на функцијата startTimer која со помош на setInterval работи на секои 5s. Функцијата display е искористена за on-click на двете странични слики, така што при кликување на една од тие слики, таа слика доаѓа на местото на главната слика, а исто така и се ресетира тајмерот така што следната слика ќе дојде за 5s.

```
function text(){
    if(document.getElementById("s1").src == sss1)
    {
        document.getElementById("n1").style.display = "inline";
        document.getElementById("n2").style.display = "none";
        document.getElementById("n3").style.display = "none";
    }else{
        if(document.getElementById("s1").src == sss2)
        {
            document.getElementById("n1").style.display = "none";
            document.getElementById("n2").style.display = "inline";
        }
    }
}
```

```

        document.getElementById("n3").style.display = "none";
    }
    else
    {
        document.getElementById("n1").style.display = "none";
        document.getElementById("n2").style.display = "none";
        document.getElementById("n3").style.display = "inline";
    }
}
}

```

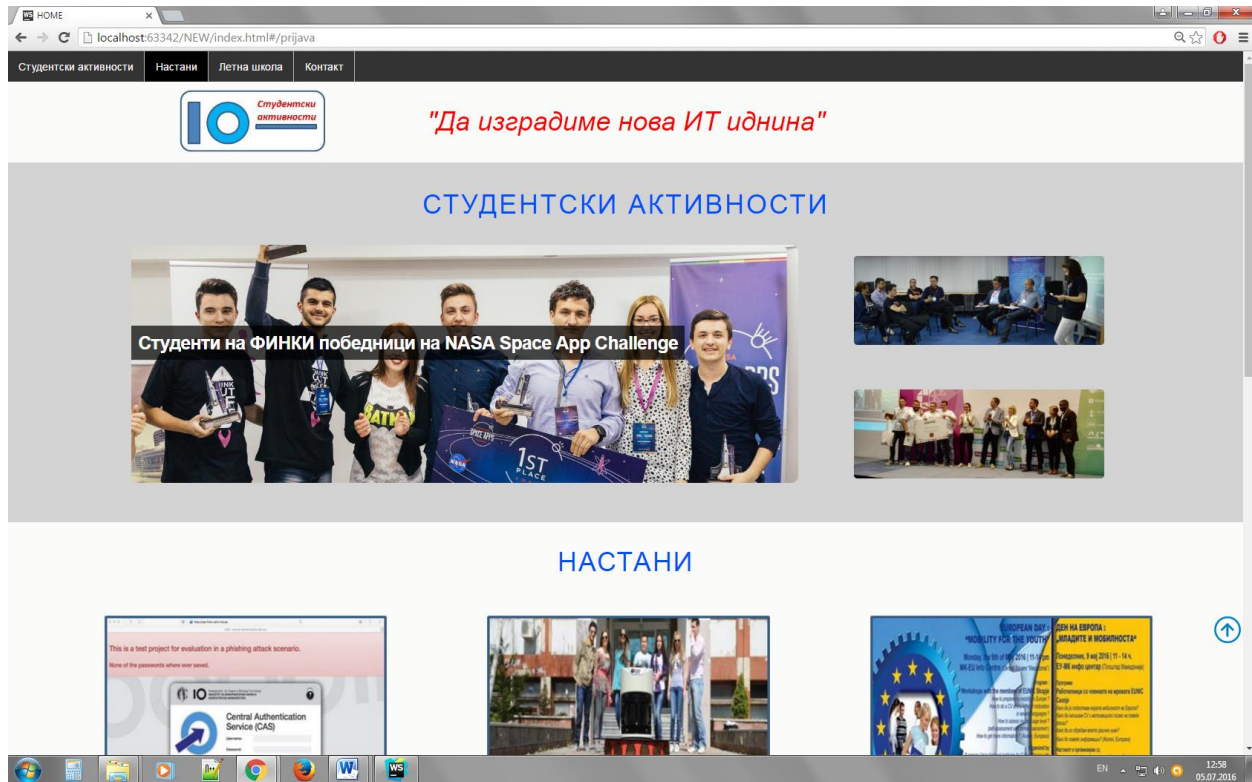
Оваа функција е искористена за да проверува која слика е на главната позиција и според тоа одлучува кој е соодветниот текст што треба да се прикажи.

Формирање на Веб страната (решение на проблемот)

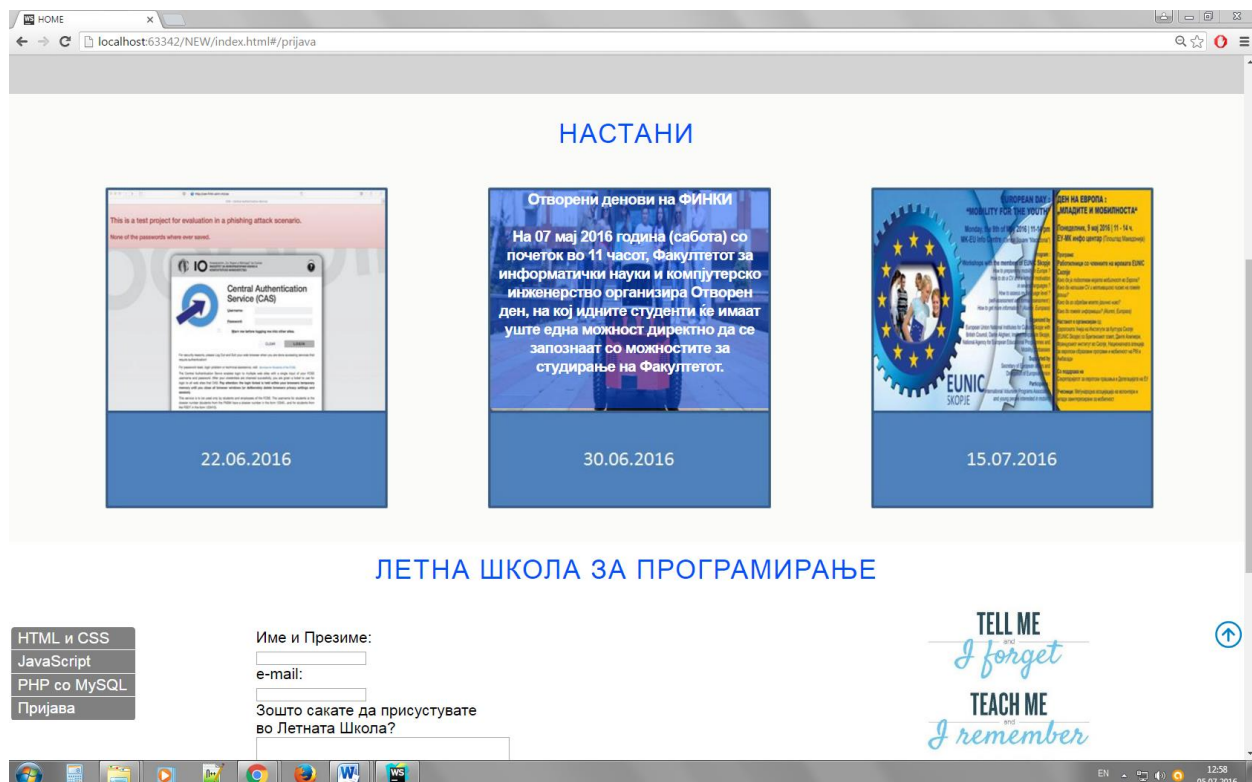
Основната идеја при формирање на **single page** веб страна е да целата содржина е опфатена на една страна при што корисникот со користење на навигациски алатки ќе се движи низ истата. Страната е замислена како информативен панел каде студентите на ФИНКИ ќе можат да ги следат актуелните проекти, курсеви и активности во кои можат активно да учествуваат и да ги развиваат своите знаења.

Содржината на страната опфаќа неколку секции каде се посебно поставени студентските активности, најновите настани, актуелностите и контакт информациите. Со хедерот корисникот на страната може со нивно кликување да дојде до брз пристап до делот кој го посакува да го разгледува, делот Студентски активности содржи три новости кои со помош на AngularJS ротираат и се менуваат на секои 5 секунди, левиот дел од овај сегмент ја зголемува сликата и излегува краток опис на навеста, а додека другите две новости чекаат 5 секунди да се ротираат и да се направи истото и за нив, но немора да се чека да се види новоста, со кликање на сликата може побрзо да се отвори новоста. Настани сегментот е едноставен преглед на три последни настани организирани од факултетот, каде е прикажан датум и промо слика од настанот, и со носење на курсерот над таа слика излегува краток опис на настанот. Летната школа се фактички курсеви организирани од факултет на различни полиња од веб дизајн и веб програмирање, заинтересираниот може да ги разгледа 3те курсеви со селектирање на имињата на курсевите при што се појавуваат додатни информации за нив и на крајот ако е одлучен може да пополни формулар за пријавување. Контакт сегментот содржи општи информации за контакт со факултетот маил, фејсбук страна и др како и мапа за каде се наоѓа факултетот. За брзо навигирање низ страницата е додадено и arrow to top, што значи со негово прицкање се враќа горе во насловниот сегмент.

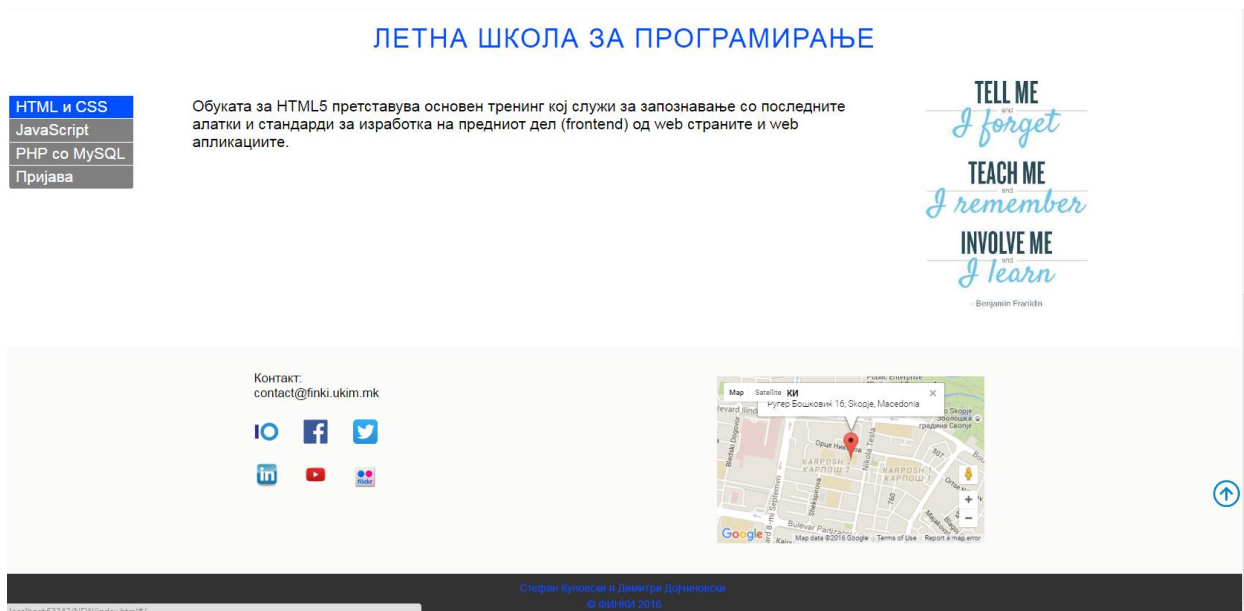
Резултати



Слика 1



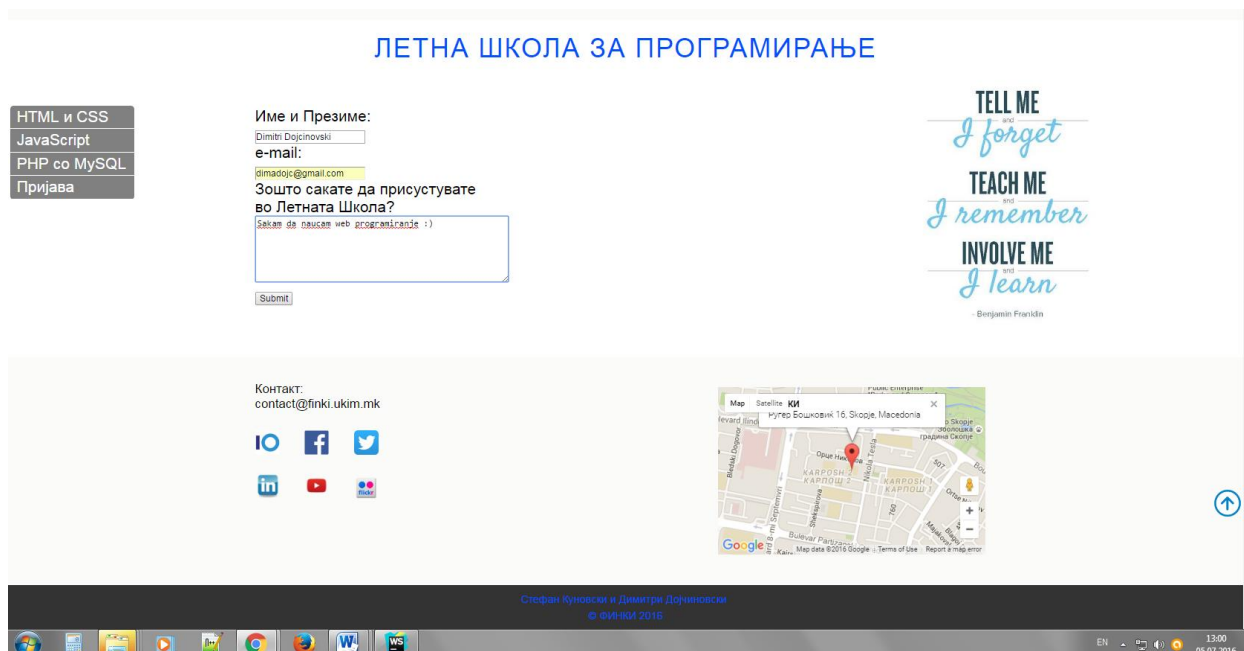
Слика 2



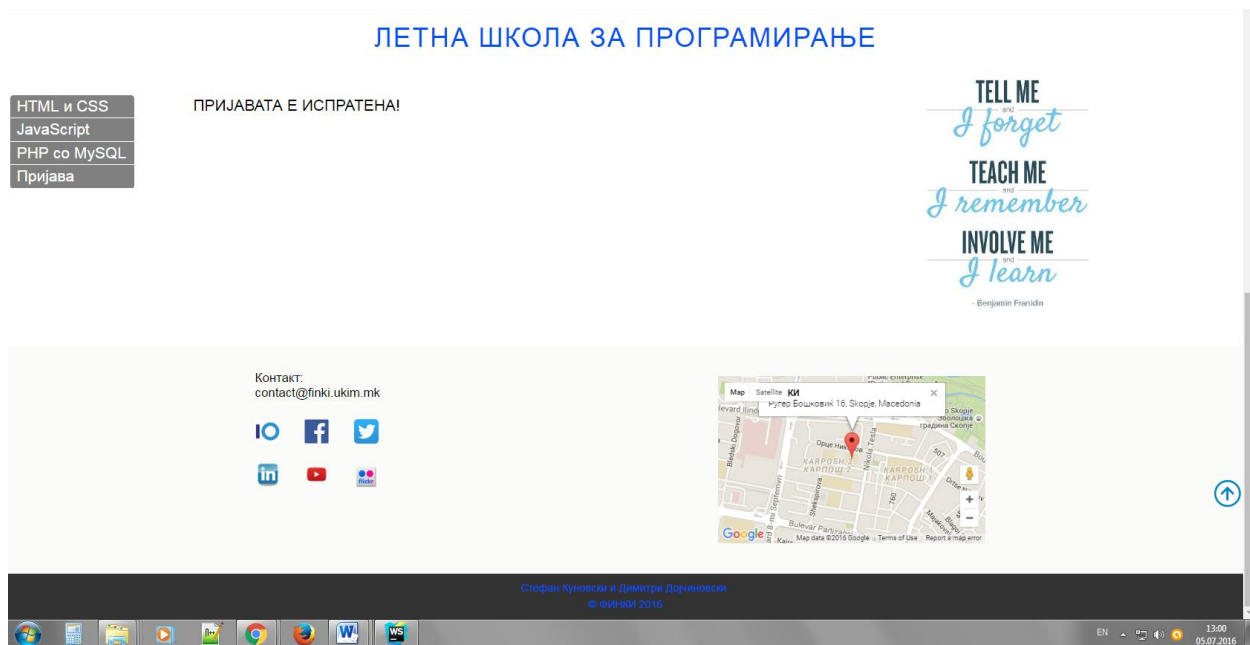
Слика 3

На слика 1 се гледа горниот дел на страната навигационата лента, насловот и Студентки Активности <div> таг каде применуваме AngularJS. На слика 2 е прикажан Настан div таг каде имаме примена на :hover селекторот.

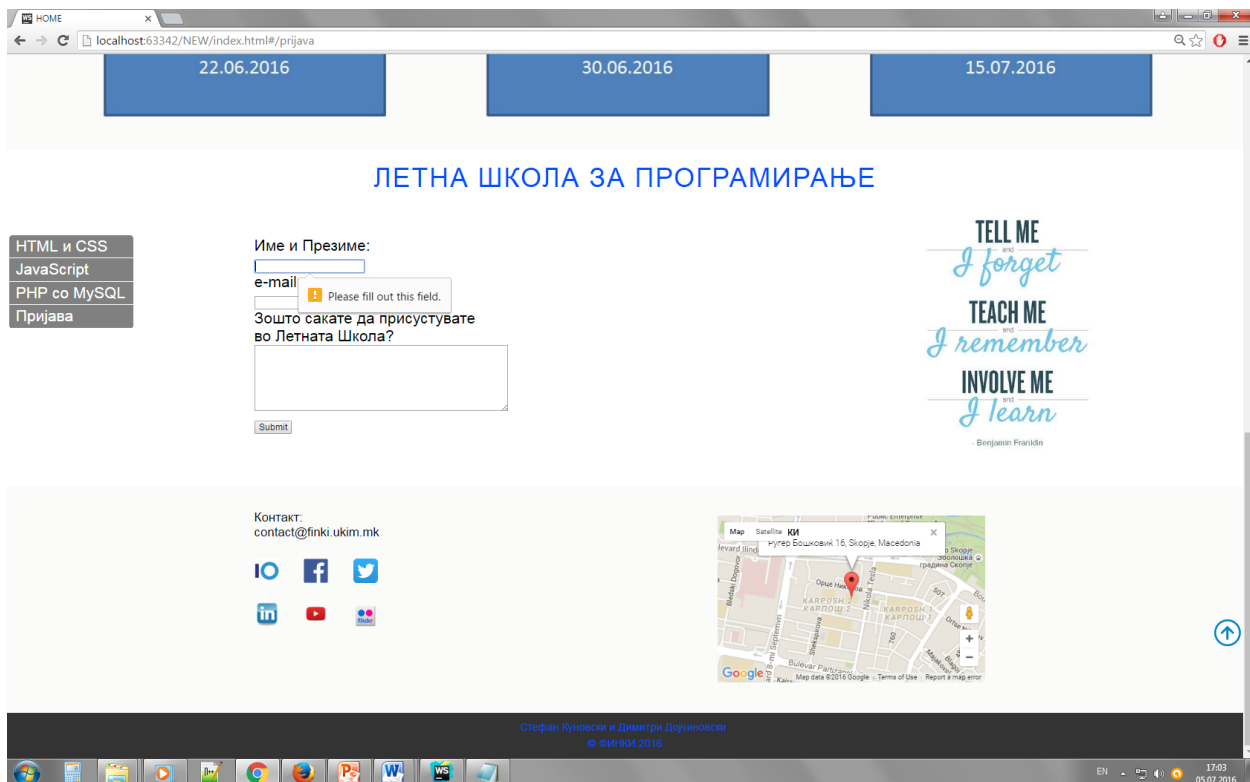
На сликите 3,4 и 5 е прикажан Летна школа и контакт елементите на веб страницата. На слика 4 и 5 е прикажана состојбата кога после пополнување на формуларот за учество во летната школа се појавува информација дека пријавата е испратена, доколку фали некоја информација формуларот нема да се испрати и ќе излеи облаче на местото каде фали информација(слика 6).



Слика 4



Слика 5



Слика 6

Дискусија и заклучок

Single-page веб страните денес имаат голема популарност т.е тие се во тренд бидејќи му помагаат на корисникот на страната да работи на една, удобна веб површина. Но секогаш немаат практична примена, како на пример страницата <http://www.porsche.com.mk/> каде се потреби повеќе од една страница за да се опише што нуди една фирма како нив. Едноставно кажано во Single-page страниците не треба да содржи премногу елементи (тоа е исто како во вреќа што крепи 5кг материјал се става 10кг материјал) бидејќи една од карактеристиките на Single-page страниците е едноставноста

Заклучок: Single-page дизајнот може да биде опција за изработка на многу страници, но тие не се секогаш совршен примерок за изработка на мала веб страна, но за сигурност вреди да се помисли да се реши некој проект со помош на Single-page дизајн.

Литература:

- Digital Business Models - Gjorgji Mancheski
- <http://www.promodel.com/library/>
- Jennifer Niederst Robbins, Learning Web Design A beginner's guide to HTML,CSS,JavaScript, and web graphics,4th edition,Part II, HTML Markup for structure,pp.182, 2012
- Jennifer Niederst Robbins, Learning Web Design A beginner's guide to HTML,CSS,JavaScript, and web graphics,4th edition,Part III, CSS for Presentation, pp.207-224, 2012
- P.J. Deitel, H.M. Deitel, Internet & World Wide Web How to program,4th edition,Chapter 5 CSS , pp.160,2008
- P.J. Deitel, H.M. Deitel, Internet & World Wide Web How to program,4th edition,Chapter 6 JavaScript: Introduction to Scripting, pp.198,2008