



ИНСТИТУТ ЗА МАТЕМАТИКУ И ИНФОРМАТИКУ
ПРИРОДНО-МАТЕМАТИЧКИ ФАКУЛТЕТ
УНИВЕРЗИТЕТ У КРАГУЈЕВЦУ

МАСТЕР РАД

MATHML

Ментор
др Владимир Цвјетковић

Студент
Фуртула Милош, 1018/2014

Децембар 2015.

Садржај

1. Увод	4
1.1. Мотивација	4
1.2. Циљ истраживања	5
1.3. Циљ пројектног задатка	5
1.4. Терминологија.....	6
1.5. Садржај рада	6
2. О MathML-у.....	7
2.1. Употреба	7
2.2. Презентација и семантика	8
2.2.1. Презентациони MathML.....	8
2.2.2. Семантички MathML.....	10
2.3. Укључивање MathML-а у HTML/XHTML документе	11
2.4. Софтверска подршка	12
2.4.1. Веб претраживачи	12
2.4.2. Едитори.....	14
2.4.3. Подршка софтверских инжењера	15
3. MathJax	16
3.1. Инсталација и конфигурација	17
3.1.1. Конфигурација путем сервера.....	17
3.1.2. Инсталација копије MathJax-а	17
3.1.3. Конфигурациони фајлови	18
3.1.4. <i>In-line</i> конфигурисање	19
3.2. MathJax подршке и излазни формати.....	21
3.2.1. Подршка за TeX и LaTeX	21
3.2.2. Подршка за MathML.....	21
3.2.3. Подршка за AsciiMath.....	22
3.2.4. Излазни формати.....	22
3.3. Динамичко генерисање	23
3.3.1. Ред акција.....	23
3.3.2. MathML из генерисане математике.....	24

3.4. Компатибилност са веб претраживачима	25
4. Семантика математичких израза	26
4.1. Content MathML	26
4.1.1. Структура и обим израза	26
4.1.2. Strict Content MathML	28
4.1.3. Речници садржаја (Content Dictionaries - CDs)	29
4.1.4. Елементи Content MathML-а за кодирање структуре израза	29
4.1.5. Одређени оператори и константе Content MathML-а	34
5. Резултати истраживања	41
5.1. Закључци истраживања	42
6. Презентација пројектног задатка	44
6.1. Почетна (<i>home</i>) страница	45
6.1.1. Генератор израза	46
6.2. Страница за чување израза	55
6.3. Страница сачуваног израза	56
6.4. Кориснички мени	59
6.5. Моји изрази	60
6.6. Колекција	61
6.7. Претрага	61
7. Софтверско решење пројектног задатка	64
7.1. Архитектура сајта	64
7.2. База података	65
7.3. Организација кода	66
7.4. Битна софтверска решења	69
7.4.1. Формирање израза	69
7.4.2. Дефинисање Content MathML кода	70
7.4.3. Наглашавање изворних кодова бојењем	71
7.4.4. Генерисање изворних кодова	72
7.4.5. Израчунавање израза на основу Content MathML кода	73
8. Закључак	75
Додатак	76
Листа скраћеница	81
Литература	82

1. Увод

Овај мастер рад се бави проучавањем и истраживањем начина за визуелни приказ и чување семантичког значења математичких и научних израза, формула и једначина на вебу.

1.1. Мотивација

Већ дуже време, није постојао ефективан начин за приказивање математичког и научног садржаја на веб страницама. Углавном су коришћене технологије које користе конверзију. Структура веб страница се дефинише у HTML¹-у. Из тог разлога се већина решења за приказ математичког и научног садржаја базира на конверзији у HTML или директним дефинисањем HTML-а. Међу популарним решењима су:

- PDF² – документи који садрже математички и научни садржај могу се лако превести у PDF.
 - Графички приказ је високог квалитета.
 - PDF није HTML и зато се не може интегрисати на веб страницу на добар начин и уједно мора постојати подршка за приказ PDF-а у веб претраживачу.
- Конверзија TeX-а и LaTeX³-а у слике и додавање слика у HTML документе.
 - Лако се имплементира и компатибилно је са свим веб претраживачима и системима.
 - Слике успоравају читавање странице, изискују додатну организацију фајлова и директоријума и приликом промена величина не чувају квалитет.
- Превођење TeX-а и LaTeX-а у HTML – LaTeX документ се преведе у HTML помоћу неког програма или библиотеке.
 - Лако се имплементира.
 - Графички приказ је понекад чудног изгледа. Некад је потребно додатно инсталирати поједине фонтове.
- HTML – директно дефинисање HTML-а, док се користе елементи за форматирање изгледа.
 - Брз приказ докумената са доста математичког и научног садржаја. Не захтева никакву посебну инсталацију и бесплатно је за коришћење.
 - Само једноставни изрази могу бити приказани.

¹ Језик за дефинисање структуре веб страница

² Портатилни формат докумената који се користи за презентовање докумената независно од софтверских апликација, хардвера и оперативних система

³ Систем за обраду текста

Поред графичког приказа, потребно је дефинисати значење математичког израза. Размена и поновно коришћење садржаја математичког израза у апликацијама није довољно олакшана. Постоји неколико решења које омогућавају обраду семантичког значења математичког израза и међу њима највише се истиче *Wolfram Mathematica*. Поред тога што постоји као самостални програм, Mathematica је софтвер који се може користити као језгро апликација за обраду значења математичких израза. Mathematica има свој језик који представља математичку нотацију која је природна за човека. Такође, пружа велики број функционалности, разна израчунавања, као и графички приказ математичких израза. Међутим, дефинисање значења математичког израза у овом софтверу може бити двосмислено, а такође постоји проблем са лиценцирањем и коришћењем, јер није отвореног кода.

Полако се повећава број научних радова који су доступни у окружењима за виртуелно учење. Било је потребно осмислити решење за истовремено дефинисање изгледа, а и јединственог значења математичког и научног садржаја. Универзална математичка нотација мора бити бесплатна за коришћење, а приступ изворним кодовима не сме бити ограничен. Из ових разлога формиран је стандард за дефинисање изгледа и значења математичких израза, **MathML**. Треба урадити детаљан преглед MathML-а и развити апликацију која омогућава његово коришћење.

1.2. Циљ истраживања

Главни циљ мастер рада је проучавање математичког програмског језика **MathML**, и истраживање које дефинише предности и мане те технологије, њене могућности и примену. Потребно је проучити оба аспекта технологије, а то су визуелни приказ математичких и научних израза (*Presentation MathML*) и чување њиховог семантичког значења (*Content MathML*). Такође ће бити проучена библиотека **MathJax**, шта пружа, и која су њена ограничења.

1.3. Циљ пројектног задатка

Потребно је направити потпуно функционалан веб сајт који се базира на апликацији *Генератор израза*. Она представља главни део пројекта и садржи графички интерфејс, који је лак за коришћење, у сврху формирања математичких израза. То значи да треба да има скуп основних оператора, симбола и константи помоћу којих би се градили сложени математички изрази. Основне функционалности сајта су:

- могућност чувања израза за даље коришћење, и визуелни приказ израза у већини популарних веб претраживача
- преузимање изворног кода сачуваних израза за коришћење на другим веб страницама
- израчунавање израза са основним операцијама коришћењем дефинисаног семантичког садржаја

1.4. Терминологија

Преглед значења термина који су коришћени у раду:

- **математика** – било који математички или научни садржај који се појављује на веб страници, а у то спада код дефинисан у **AsciiMath**-у, **TeX**-у, **LaTeX**-у, **MathML**-у, или **OpenMath**-у
- **математички израз** – сложени математички или научни (физика, хемија, ...) израз
- **основни израз** – било који оператор, симбол или константа који служи за грађење сложених израза

1.5. Садржај рада

На почетку рада ће бити дат кратак увод у MathML, употреба и софтверска подршка. Затим ће бити описан начин рада и употреба MathJax библиотеке. MathJax библиотека спада у истраживање о презентационом MathML-у. Семантичко значење израза ће бити проучено кроз Content MathML. Након тога се могу видети резултати и закључци истраживања, који представљају увод за опис пројектног задатка. Пројектни задатак ће бити описан кроз објашњење начина рада и кроз дефиницију битних софтверских решења.

2. О MathML-y

Mathematical Markup Language (MathML) служи да омогући и олакша употребу математичког и научног садржаја на вебу. Такође служи као подршка другим апликацијама као што су рачунарски алгебарски системи⁴. Може се користити за презентацију математичких нотација са визуелним приказом високог квалитета и за апликације где семантика математичких израза игра већу улогу.

Овај језик је проширење **XML**-а и служи за описивање математичких нотација, чувајући и структуру и садржај. *Extensible Markup Language* (XML) је језик помоћу ког се описују произвољне структуре података. Служи за чување докумената у формату који је лако читљив и за човека, и за машину. Овакав тип структуре користе машине за размену и чување података. Циљ MathML-а је да интегрише математичке изразе на веб странице и друге документе. Препоручен је од стране W3C⁵ и део је HTML5⁶.

MathML 1 је објављен у Априлу 1998. године као први XML језик који је препоручен од стране W3C. Последња верзија је изашла у октобру 2010. године под именом MathML 3.

2.1. Употреба

Иако HTML има велики репертоар тагова, не обазире се на математику. Без начина да се употребе HTML тагови за приказ математичких формула, аутори су користили драстичне мере. Популарне методе укључују додавање слика у техничким документима који имају математички или научни садржај. Те слике су преузете из неких других пакета и сачуване у неком од формата за слике. MathML садржи велики број XML тагова који могу да се искористе за дефинисање једначина, како за презентацију тако и за семантику. Такође, MathML, омогућава да се сачува значење иза једначина, а не само да се фокусира на то како ће се оне приказати на екрану. Ово се базира на томе да математичке једначине имају значење за многе апликације без обзира на то како се оне приказују.

MathML је формат ниског нивоа који служи за описивање математике као основе за комуникацију између машина. Није предвиђен за ручне измене, већ да служи за управљање специјализованим алатима као што су *едитори* израза, или за слање ка или од других математичких пакета.

⁴ Софтверски програм који омогућава рачунање математичких израза

⁵ World Wide Web Consortium (W3C) је главна интернационална организација стандарда за World Wide Web (WWW тј. W3)

⁶ HTML5 је пета ревизија HTML стандарда, програмског језика који служи за уређивање и презентацију садржаја на вебу

2.2. Презентација и семантика

С обзиром да се значење израза чува посебно од презентације, за шта, и на који начин ће се оно користити остављено је на кориснику да одлучи. На пример, веб страница са убаченим MathML-ом, може да се гледа као нормална веб страница у више различитих претраживача, али слабовидим корисницима може исти тај MathML да се чита помоћу читача екрана (*MathPlayer* додатак за *Internet Explorer*, *Opera 9.5+*, екстензија *Fire Vox* за *Firefox*).

2.2.1. Презентациони MathML

Задатак презентационог MathML-a је приказ израза, и има око 30 елемената. Сва имена елемената почињу са **m**. Израз презентационог MathML-a се састоји из токена који се комбинују користећи елементе вишег нивоа, који контролишу њихов изглед (такође, постоји око 50 атрибута, који углавном контролишу fine детаље).

Токен елементи генерално садрже карактере (не друге елементе) и то:

- **идентификатори** – `<mi>x</mi>`
- **оператори** – `<mo>+</mo>`
- **бројеви** – `<mn>2</mn>`
- **текст** – `<mtext>non zero</mtext>`

MathML у HTML5 омогућава већину HTML тагова у тагу `mtext` као у следећем примеру где се подебљава прва реч:

Листинг 1

```
1 <mtext><b>non</b> zero</mtext>
```

Ови елементи се комбинују користећи елементе изгледа који генерално садрже само елементе. Елементи изгледа су:

- хоризонтални ред елемената – `<mrow>`
- суперскриптови, границе изнад и испод оператора (нпр. суме) – `<msup>`, `<munderover>`, итд.
- разломци – `<mfrac>`
- корени – `<msqrt>`, `<mroot>`
- оградавање садржаја, нпр. заградама – `<mfenced>`

Као што је иначе у HTML-y и XML-y, многи ентитети су доступни за дефинисање посебних симбола, по имену, као што је `π` за π и `→` за \rightarrow . Интересантна могућност MathML-a је та што постоје ентитети који приказују невидљиве операторе, као што је `⁢` за имплицитно множење.

Једначина $ax^2 + bx + c$ се састоји из два елемента изгледа, један за хоризонтални ред елемената и један за суперскрипт. Код за тако нешто изгледа овако:

Листинг 2

```
1 <mrow>
2   a &InvisibleTimes; <msup>x 2</msup>
3   + b &InvisibleTimes; x
4 </mrow>
```

Међутим, индивидуални токени такође морају бити идентификовани, па потпуна форма изгледа овако:

Листинг 3

```
1 <mrow>
2   <mi>a</mi> <mo>&InvisibleTimes;</mo> <msup> <mi>x</mi> <mn>2</mn>
3   </msup>
4   <mo>+</mo> <mi>b</mi> <mo>&InvisibleTimes;</mo> <mi>x</mi>
5   <mo>+</mo> <mi>c</mi>
6 </mrow>
```

Валидан MathML документ се састоји из XML декларације, DOCTYPE⁷ декларације, и комбинације MathML елемената. Тело документа садржи MathML изразе који се појављују у `<math>` елементима у документу по потреби. Углавном ће MathML бити укључен у опште документе, као што су HTML и друге XML шеме. Комплетан документ који садржи претходну једначину изгледа овако:

Листинг 4

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE math PUBLIC "-//W3C//DTD MathML 2.0//EN"
3   "http://www.w3.org/Math/DTD/mathml2/mathml2.dtd">
4 <math xmlns="http://www.w3.org/1998/Math/MathML">
5   <mrow>
6     <mi>a</mi>
7     <mo>&InvisibleTimes;</mo>
8     <msup>
9       <mi>x</mi> <mn>2</mn>
10    </msup>
11    <mo>+</mo>
12    <mi>b</mi> <mo>&InvisibleTimes;</mo> <mi>x</mi>
13    <mo>+</mo>
14    <mi>c</mi>
15  </mrow>
16 </math>
```

⁷ Инструкција која повезује одређени XML документ (нпр. веб страницу) са дефиницијом типа документа

2.2.2. Семантички MathML

Семантички MathML се фокусира на значење израза пре него на његову презентацију. Главни елемент у семантичком MathML-y је `<apply>` елемент који репрезентује примену функције. Функција која треба да се примени је први елемент у оквиру овог тага, и њени операнди или параметри су остали елементи у оквиру тага. Семантички MathML користи само неколико атрибута.

Токени као што су идентификатори и бројеви су индивидуално означени, са елементима `ci` и `cn`. Уместо да буде само још један тип токена, оператори су репрезентовани као посебни елементи, чија математичка семантика је позната MathML-y, као што је `times`, `power` и други. Постоји преко 100 различитих елемената за различите функције и операторе.

Једначина $ax^2 + bx + c$ може да се репрезентује на следећи начин:

Листинг 5

```
1  <math>
2      <apply>
3          <plus/>
4          <apply>
5              <times/>
6              <ci>a</ci>
7              <apply>
8                  <power/>
9                  <ci>x</ci>
10                 <cn>2</cn>
11             </apply>
12         </apply>
13         <apply>
14             <times/>
15             <ci>b</ci>
16             <ci>x</ci>
17         </apply>
18         <ci>c</ci>
19     </apply>
20 </math>
```

2.3. Укључивање MathML-а у HTML/XHTML документе

С обзиром да је MathML уствари XML, може се укључити унутар других XML докумената као што су XHTML документи, користећи *XML namespace*⁸. Неки веб претраживачи могу да прикажу презентациони MathML укључен у XHTML.

Листинг 6

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1 plus MathML 2.0//EN"
   "http://www.w3.org/Math/DTD/mathml2/xhtml-math11-f.dtd">
3  <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
4      <head>
5          <title>Пример MathML укљученог у XHTML документ</title>
6          <meta name="description" content="Пример MathML укљученог у
   XHTML документ"/>
7      </head>
8      <body>
9          <h1>Пример MathML укљученог у XHTML документ</h1>
10         <p>Површина круга је
11         <math xmlns="http://www.w3.org/1998/Math/MathML">
12             <mi>&#x03C0;<!-- π --></mi>
13             <mo>&#x2062;<!-- &InvisibleTimes; --></mo>
14             <msup>
15                 <mi>r</mi>
16                 <mn>2</mn>
17             </msup>
18         </math>.
19         </p>
20     </body>
21 </html>

```

Пример MathML укљученог у XHTML документ

Површина круга је πr^2 .

Слика 1 Приказ претходног кода као странице у веб претраживачу

MathML је такође подржан у HTML5 документима и није потребно да се наведе *XML namespace* као у XHTML (Код 7).

⁸ Пружа јединствено именовање елемената и атрибута у XML документу

Листинг 7

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <meta charset="utf-8">
5          <title>Пример MathML укљученог у HTML5 документ</title>
6      </head>
7      <body>
8          <h1>Пример MathML укљученог у HTML5 документ</h1>
9          <p>Површина круга је
10             <math>
11                 <mi>&pi;</mi>
12                 <mo>&InvisibleTimes;</mo>
13                 <msup>
14                     <mi>r</mi>
15                     <mn>2</mn>
16                 </msup>
17             </math>.
18          </p>
19      </body>
20 </html>
```

2.4. Софтверска подршка

XML начин структурирања треба да омогући широку примену MathML-a са стандардним приказом у апликацијама као што су веб претраживачи и олакша директну интерпретацију значења у математичким софтверским производима. MathML није предвиђен да буде писан или мењан директно од стране људи.

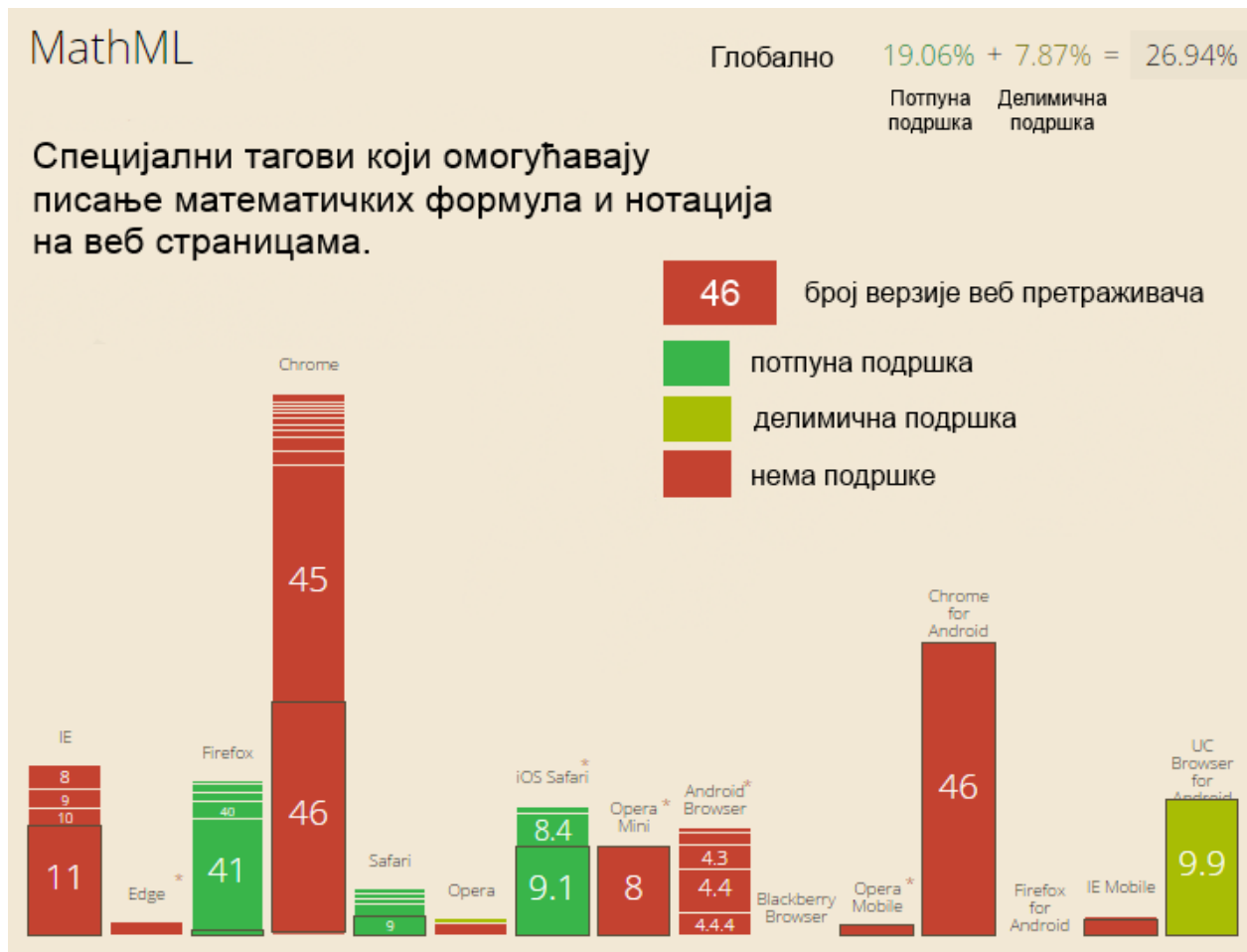
2.4.1. Веб претраживачи

Од најзаступљенијих веб претраживача, највише подршке за MathML пружају претраживачи базирани на *Gecko rendering engine*⁹-у као што су *Firefox*, *Firefox Developer*, *Sea Monkey*.

Последња верзија веб претраживача *Opera*, верзија 34, не пружа подршку за MathML. Стара *Opera* од верзије 9.5 до 12, подржава MathML, али има одређених проблема. Подршка од стране *Internet Explorer*-а такође није доступна, али за *IE6* и *IE9* може да се искористи бесплатни додатак *MathPlayer*. *MathPlayer* је универзални математички читач који пружа подршку за MathML, и омогућава визуелни приказ као и читање израза у *Internet*

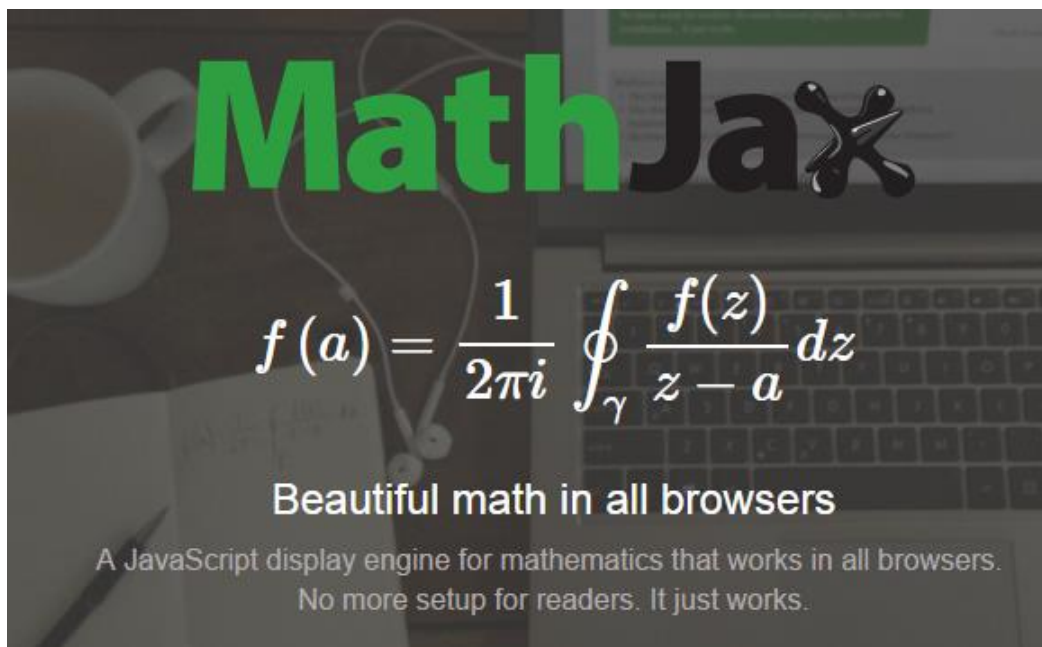
⁹ Софтверска компонента која садржај дефинисан у HTML, XML и слично, и информације о формату (CSS, XSL, ...) приказује на екран у форматираном садржају

Explorer-y. *IE10* је имао неких проблема са овим додатком па су га у *IE11* онемогућили као додаток.



Слика 2 Статистика подршке MathML-а (извор <http://caniuse.com/#feat=mathml>)

MathML такође подржава *Safari* у верзијама 5.1+. Подршка је додата за *Chrome*, верзије 24, али је касније прекинута због нестабилности. *Google* је престао са подршком за MathML са образложењем да проблеми безбедности архитектуре и мала потреба за коришћењем не вреде њиховог труда. Изаване су велике дебате када је *Google* објавио да је престао са подршком за MathML. *Google* је навео да није неопходно да веб претраживач пружа подршку за MathML. Разлог је тај што постоји библиотека **MathJax**, која омогућава приказ израза дефинисаних MathML-ом, у већини популарних веб претраживача. Друга страна је била незадовољна том одлуком и образложењем. Када се пишу озбиљни академски радови који садрже математичке и научне изразе, MathJax није идеално решење. То су показали бројни тестови, где се у раду налази доста математике. MathJax-у је потребно доста времена да такве радове процесуира и ваљано прикаже. Зато је неопходно да веб претраживач има подршку за MathML. Упркос томе, *Chrome* је и даље не пружа.



Слика 3 MathJax официјална страница

Квалитет приказа MathML-а у претраживачима зависи од инсталираних фонтова. Према мишљењу чланова **MathJax** тима, подршка већине претраживача је углавном резултат неплаћеног, волонтерског рада и времена.



Слика 4 Популарни веб претраживачи

2.4.2. Едитори

Постоји велики број едитора са подршком за MathML (на пример, *MathFlow*, *MathType*, *MathMagic*, *Publicon*, *WIRIS*, ...). Већина *office* производа такође пружа подршку за MathML (*Apache OpenOffice*, *LibreOffice*, *MS Office 2007*, ...). Такође подршку пружају математички софтверски производи (*Mathematica*, *Firemath*, ...).



Слика 5 MathML едитори

Већина едитора нуде само презентациони MathML. *MathDox* едитор формула је *OpenMath* едитор који нуди и презентациони и семантички MathML.

2.4.3. Подршка софтверских инжењера

Подршка за MathML формат убрзава развој софтверских апликација у више различитих тема, као што је едукација помоћу рачунара (даљинско учење, електронске свеске и други школски материјали), аутоматизовано стварање извештаја, рачунарски алгебарски системи, и многе друге апликације у пољу математике, науке, бизниса, економије и другим пољима.

Још један стандард, који је направљен већином од стране истих људи који су направили семантички MathML, и који служи углавном за семантичко чување формула је *OpenMath*. Може да буде укључен у MathML. Посебан формат *OMDoc*, је направљен за конструисање већих математичких структура од формула, од изјава као што су дефиниције, теореме, докази, до теорија и научних радова. Формуле у оваквим документима могу бити писане или у семантичком MathML-у или у *OpenMath*-у, а за презентацију, преводе се у презентациони MathML.

Како већина претраживача не пружа подршку за MathML најбоље решење које тренутно постоји нуди библиотека **MathJax**.

3. MathJax

С обзиром да подршке за MathML, у већини популарних веб претраживача, или нема или је омогућен рад са MathML-ом уз посебне додатке, потребно је нешто што ће кориснику омогућити приказ математичких израза у његовом веб претраживачу без додатних инсталација и већег корисниковог труда. Математички израз треба да буде приказан у свим веб претраживачима помоћу MathML-а и то управо омогућава MathJax.

MathJax је *JavaScript* алат отвореног кода који служи за приказивање LaTeX, MathML и AsciiMath нотације у свим модерним веб претраживачима. Дизајниран је са циљем да уједини скорашње помаке у веб технологијама у једну, дефинитивну, математички оријентисану платформу која је подржана у свим веб претраживачима и оперативним системима, укључујући и оне на мобилним уређајима. Не захтева никакво подешавање од стране корисника (додаци за преузимање или софтвер за инсталирање). Потребно је само укључити MathJax и математику на веб страницу, и MathJax ради остало.



Слика 6 MathJax

У веб претраживачима који их подржавају, да би произвео подешавање високог квалитета које скалира и штампа у високој резолуцији (за разлику од математичких израза убачених као слике) MathJax користи фонтове базиране на вебу. MathJax може да се користи са читачима екрана и тако омогући доступност слабовидим корисницима. Са MathJax-ом, математички изрази су базирани на тексту уместо на сликама, па је тако омогућена претрага, што значи да ти изрази могу да се претражују као регуларни текст.

MathJax је модуларан, па тако читава компоненте само када је то потребно, и може да се прошири тако да садржи нове могућности по потреби. Може обимно да се конфигурише, што омогућава персонализацију специјалних потреба веб сајтова. Такође има богат апликациони програмски интерфејс (API) који може да се користи да математички изрази и њихова манипулација буде динамичка и интерактивна.

3.1. Инсталација и конфигурација

MathJax омогућава приказ математичких израза на веб страницама, користећи или LaTeX, или MathML, или AsciiMath нотацију. Ти изрази ће бити процесуирани користећи *JavaScript* да би произвели израз који треба бити приказан у било ком модерном веб претраживачу.

Постоје два начина да се користи MathJax. Најлакши начин је да се искористи копија MathJax-а доступна на cdn.mathjax.org, али може и да се преузме и инсталира копија MathJax-а на серверу, или локално на хард диску (без потребе за приступом интернету).

3.1.1. Конфигурација путем сервера

Најлакши начин да се користи MathJax је да се повеже директно са јавном инсталацијом која је доступна преко *MathJax Content Distribution Network* (CDN). Овим путем није потребно инсталирати MathJax и може се одмах користити.

Да би користили MathJax са овог сервера, потребно је повезати MathJax са веб страницама које га укључују и додати неке математичке изразе на страницу како би их MathJax приказао.

Први корак је да у `<head>` блоку документа се дода следећи код:

Листинг 8

```
1 <script type="text/javascript"  
  src="https://cdn.mathjax.org/mathjax/latest/MathJax.js?config=TeX-AMS-  
  MML_HTMLorMML">  
2 </script>
```

Такође може и у `<body>` блоку ако је потребно али није пожељно. Овај део кода ће да очита последњу верзију MathJax-а са сервера, и конфигуришаће га да препознаје математичке изразе, како у TeX тако и у MathML нотацији, и провериће да ли треба да прикаже израз помоћу MathML-а или HTML-а са CSS-ом.

Конфигурација `TeX-AMS-MML_HTMLorMML` је општа, па самим тим и највећи конфигурациони фајл, али је најлакши за почетна подешавања са MathJax-ом. Ово није најефикаснија конфигурација и постоје разне комбинације конфигурационих фајлова за персонализоване потребе.

Добра пракса је да се за везу са овим сервером користи сигурни протокол (HTTPS), јер је приступ преко обичног протокола отворен за нападе где MathJax скрипта може бити измењена.

3.1.2. Инсталација копије MathJax-а

Уколико се серверу не може приступити, може се инсталирати MathJax на одређеном серверу или локално на хард диску. Да би се то урадило потребно је преузети копију MathJax-а и пребацити је на сервер или хард диск, а затим конфигурисати MathJax за

потребе сајта. Након тога треба повезати MathJax са страницама које садрже математичке изразе и додати саме математичке изразе како би их MathJax приказао.

Изворни код за MathJax се налази на *GitHub*¹⁰-у, одакле се може и преузети. Тај код се пребаци на одређени сервер и након тога је потребно повезати преузети MathJax са веб страницама које садрже математичке изразе. Уколико је назив директоријума, у којем се налази код за MathJax, под именом **MathJax**, и стандардна конфигурација **TeX-AMS-MML_HTMLorMML** тада део кода који треба унети у **<head>** блок изгледа овако:

Листинг 9

```
1 <script type="text/javascript" src="MathJax/MathJax.js?config=TeX-AMS-  
MML_HTMLorMML">  
2 </script>
```

Након повезивања веб странице са MathJax-ом потребно је на страницу додати код за математички израз који MathJax треба да прикаже. Тај код може бити дефинисан користећи TeX, LaTeX, MathML и AsciiMath нотацију, или комбинацију свега у оквиру исте странице.

3.1.3. Конфигурациони фајлови

MathJax долази са више предефинисаних конфигурационих фајлова, који се чувају у **MathJax/config** директоријуму. Међу њима су:

- **default.js** – фајл који садржи скоро све конфигурационе опције са коментарима који их описују, који се могу мењати по потреби.
- **TeX-AMS-MML_HTMLorMML.js** – дозвољава да математика буде дефинисана у TeX, LaTeX, или MathML нотацији, са *AMSMath*¹¹ и *AMSsymbols* пакетима, генеришући приказ користећи MathML ако то претраживач подржава, а HTML-with-CSS у супротном случају.
- **TeX-AMS_HTML.js** – дозвољава да математика буде дефинисана у TeX или LaTeX нотацији, са *AMSMath* и *AMSsymbols* пакетима, генеришући приказ користећи HTML-with-CSS.
- **MML_HTMLorMML.js** – дозвољава да математика буде дефинисана у MathML нотацији, генеришући приказ користећи MathML ако то претраживач подржава, а HTML-with-CSS у супротном случају.
- **AM_HTMLorMML.js** – дозвољава да математика буде дефинисана у AsciiMath нотацији, генеришући приказ користећи MathML ако то претраживач подржава, а HTML-with-CSS у супротном случају.

¹⁰ Репозиторијум изворних кодова

¹¹ Пакет за LaTeX који олакшава писање математичких формула и побољшава квалитет излазног формата

- **TeX-AMS-MML_SVG.js** – дозвољава да математика буде дефинисана у TeX, LaTeX, или MathML нотацији, са *AMSMath* и *AMSsymbols* пакетима, генеришући приказ користећи SVG.
- **TeX-MML-AM_HTMLorMML.js** – дозвољава да математика буде дефинисана у TeX, LaTeX, MathML, или AsciiMath нотацији, са *AMSMath* и *AMSsymbols* пакетима, генеришући приказ користећи MathML ако то претраживач подржава, а HTML-with-CSS у супротном случају.

Први од ових фајлова је фајл који се може мењати по сопственим потребама а остало су предефинисани фајлови и зову се комбиновани конфигурациони фајлови. Углавном, MathJax читава своје компоненте само по потреби, али за сваку компоненту је потребно учитавање фајла. То може да изазове кашњење у приказивању математике. Комбиновани конфигурациони фајлови читавају већину потребних фајлова као један велики фајл, смањујући број захтева. То значи да ће MathJax приступати компонентама много брже него што би без комбинованог фајла, али ће се учитати компоненте које се можда никад неће користити. При одабиру конфигурационог фајла за учитавање, треба водити рачуна о нотацијама које се користе, начину приказа, као и броју појављивања математике на страницама, и величини конфигурационог фајла. На мобилним уређајима понекад постоји ограничење величине фајлова који се кеширају, па ће можда бити присиљени да преузимају конфигурацију на свакој страници.

3.1.4. *In-line* конфигурисање

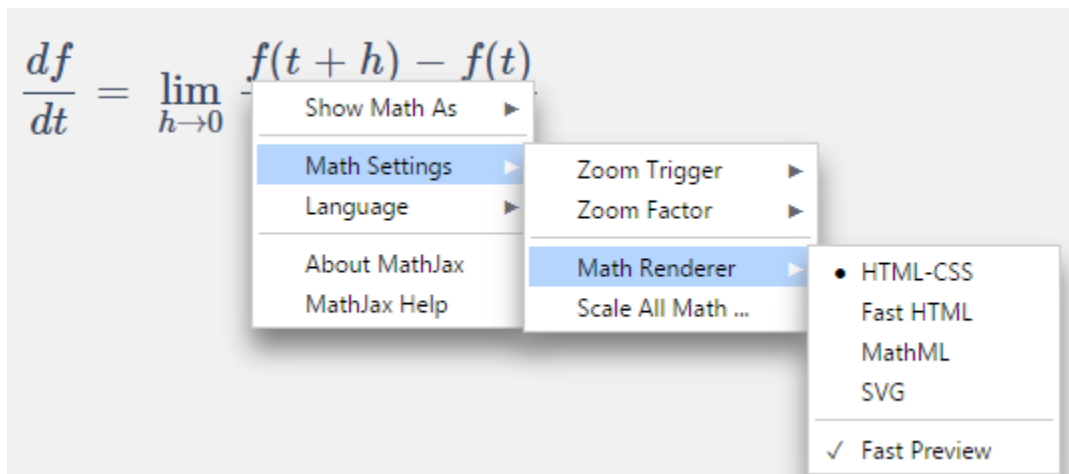
Други начин да се конфигурише MathJax је *in-line* конфигурацијом, где се дефинишу конфигурационе опције на самој страници. Овакав тип конфигурације састоји се из два `<script>` тага, један за дефинисање конфигурације и други за учитавање MathJax-а. Пошто MathJax почиње конфигурациони процес чим је прочитан, конфигурациони скрипт мора бити пре самог учитавања.

Листинг 10

```
1 <script type="text/x-mathjax-config">
2   MathJax.Hub.Config({
3     extensions: ["tex2jax.js"],
4     jax: ["input/TeX", "output/HTML-CSS"],
5     tex2jax: {
6       inlineMath: [ ['$','$'], ["\\(", "\\)"] ],
7       displayMath: [ ['$$', '$$'], ["\\[", "\\]"] ],
8       processEscapes: true
9     },
10    "HTML-CSS": { availableFonts: ["TeX"] }
11  });
12 </script>
13 <script type="text/javascript" src="path-to-MathJax/MathJax.js">
14 </script>
```

Позивом `MathJax.Hub.Config()` методе врши се подешавање конфигурације. Неке њене опције са подразумеваним вредностима су:

- `jax: ["input/TeX", "output/HTML-CSS"]` – листа улаза и излаза чији се главни код читава само када се стварно користе
- `extensions: []` – листа екстензија
- `config: []` – листа конфигурационих фајлова, могу се дефинисати локални макрои и постоји пример под именом `config/local/local.js`
- `styleSheets: []` – листа CSS фајлова
- `preJax: null` and `postJax: null` – шаблон који се брише пре и после математичких тагова скрипти. Да би се избегла грешка у IE где претраживач брише празна поља за која мисли да се понављају, треба да се додају неки карактери пре и после `<script>` тагова, јер ови тагови не генеришу садржај на страници. То значи да, када MathJax укључи математику, неће бити празних поља између, претходног текста, математике и текста који следује. На пример да би математика била у оквиру дуплих угластих заграда треба да се напише `preJax: "\\[\\[", postJax: "\\]\\]"`
- `showProcessingMessages: true` – да ли се информације о процесирању приказују
- `messageStyle: "normal"` – сложеност информација
- `elements: []` – листа ID¹²-а DOM елемената за процесирање
- `skipStartupTypeset: false` – означава да ли се прескаче почетно процесуирање математике
- `menuSettings: { ... }` – подешавања менија (*zoom*, *texHints*, *semantics*, ...)
- `errorSettings: { ... }` – подешавање приказа грешака



Слика 7 MathJax мени

¹² Јединствени идентификатор HTML елемента у документу (DOM елемент)

3.2. MathJax подршке и излазни формати

MathJax пружа подршку за TeX, LaTeX, MathML и AsciiMath, а од излазних формата HTML-with-CSS, SVG, NativeMML.

3.2.1. Подршка за TeX и LaTeX

Подршка за TeX и LaTeX се састоји из два дела, *tex2jax* претпроцесора и TeX процесора улаза. Први тражи математику унутар странице (обележена између знакова $$$$ као $$$...$$$) и обележава за касније процесирање. Други преводи TeX нотацију у интерни формат MathJax-а и касније приказује на веб страници. Оба се могу конфигурисати, први кроз MathJax конфигурацију, други кроз екстензије.

Листинг 11

```
1 MathJax.Hub.Config({
2     tex2jax: {
3         inlineMath: [['$', '$'], ['\\(', '\\)']],
4         processEscapes: true
5     }
6 });
```

Овако се могу конфигурисати знаци за обележавање математике у документу.

Да би се користиле екстензије довољно је у конфигурацији додати жељену екстензију у листу екстензија.

Листинг 12

```
1 <script type="text/x-mathjax-config">
2     MathJax.Hub.Config({ TeX: { extensions: ["autobold.js"] }});
3 </script>
```

Доступне екстензије за TeX и LaTeX су Action, AMSmath, AMSsymbols, AMScd, Autobold, BBox, Begingroup, Cancel, Color, Enclose, ExtPfeil, HTML, mhchem, noErrors, noUndefined, Unicode support, Autoload-all.

Подршка за TeX и LaTeX долази са великим бројем симбола и команди.

3.2.2. Подршка за MathML

Подршка за MathML се састоји из три дела, *mml2jax* претпроцесор, MathML процесор улаза и процесор излаза NativeMML. Први тражи $<math>$ тагове и обележава их за касније процесирање. Други преводи MathML у интерни формат који користи MathJax, и трећи претвара интерни формат у MathML у оквиру странице тако да може да се прикаже у веб претраживачима са MathML подршком.

Због модуларног дизајна MathJax-а, не морају се користити све три компоненте. На пример, може се користити *tex2jax* претпроцесор и TeX процесор улаза, а NativeMML као процесор излаза, тако да се математика дефинише у TeX формату а приказује као MathML.

Да би се користио Content MathML у документима, потребно је да се екстензија `content-mathml.js` дода у листу екстензија.

3.2.3. Подршка за AsciiMath

Подршка за AsciiMath се састоји из два дела, *asciimath2jax* претпроцесора и AsciiMath процесора улаза. Први тражи математику на страницама обележену између посебних карактера (``...``). Резултат се генерише аналогно претходним претпроцесорима и процесорима улаза. На сличан начин се може конфигурисати којим карактерима се обележава математика.

3.2.4. Излазни формати

HTML-CSS излазни процесор производи приказ високог квалитета у свим познатијим претраживачима. Ово је примарни режим MathJax-а. Предност су квалитет и конзистентност, али је спорији него NativeMML приликом рендеровања. Користи фонтове са веба па то изазива проблеме приликом штампања у неким претраживачима.

SVG излазни процесор се користи јер је у већини познатих претраживача и великом броју мобилних уређаја подржан. Високог је квалитета, и за нијансу бржи од HTML-CSS-а и нема проблема са фонтовима. Овај формат добро ради са неким од *ebook*¹³ читача. Ипак, овај формат се не препоручује јер није подржан у свим IE, као и неким верзијама *Android*¹⁴ претраживача, а такође има техничке проблеме са параметрима за величину и фонтом.

NativeMML излазни процесор користи интерну подршку претраживача за MathML (уколико постоји) за приказ математике. Предност овог процесора је његова брзина. Мана је што зависи од имплементације процесуирања MathML-а претраживача. Може имати проблема у приказивању на које MathJax не може да утиче.

С обзиром да не подржавају сви претраживачи MathML, MathJax као подразумевани излазни формат користи HTML-CSS у свим претраживачима сем у IE, када постоји *MathPlayer* додатак. За сваки тип претраживача може да се дефинише афинитет коришћења процесора, па се на основу тога приказује кроз одређени излазни формат.

Листинг 13

```
1 MathJax.Hub.Config({
2     MMLorHTML: {
3         prefer: { MSIE: "MML", Firefox: "HTML", Safari: "HTML",
4                   Chrome: "HTML", Opera: "HTML", other: "HTML" } } });
```

¹³ Књиге сачуване у дигиталној форми које могу да се читају на рачунару и сличном електронском уређају

¹⁴ Оперативни систем за мобилне уређаје

3.3. Динамичко генерисање

MathJax.Hub је главна контролна структура за MathJax. Ту се улаз и излаз повезују, и врши процесирање MathJax `<script>` тагова. Процесирање математике на страници може изискивати учитавање екстерних фајлова, а пошто је учитавање фајла асинхроно, велики број метода ће вратити резултат пре него што су њихове функције извршене. Из тог разлога, укључују се *callback* функције (функције каснијег позива) које се позивају након што се акција заврши. Ово може да се користи да се синхронизују одређене акције.

3.3.1. Ред акција

Уколико се страница учитава динамички и математика се појављује након што MathJax изврши процесирање, потребно је да MathJax поново претражи страницу, тражећи математику, за још једно процесирање и генерисање излаза. Да би се то урадило, користи се **MathJax.Hub.Typeset()** метода. Ово ће да изазове да претпроцесори прођу кроз страницу поново, а MathJax ће онда непроцесуирану математику да преведе и прикаже, не мењајући већ процесуирану математику.

Ова метода се не треба позивати директно, с обзиром да MathJax ради асинхроно. Потребно је обезбедити да је позив ове методе синхронизован са осталим акцијама MathJax-а. На пример, можда се чека на очитавање излаза, па је потребно ову методу додати у ред акција. Тада ће се метода извршити након што се остале акције, које претходе у реду, изврше. Да би се ова метода додала у ред акција користи се команда

Листинг 14

```
1 MathJax.Hub.Queue(["Typeset",MathJax.Hub]);
```

Метода **MathJax.Hub.Typeset()** такође прихвата параметар, тј. DOM елемент или његов ID.

Листинг 15

```
1 MathJax.Hub.Queue(["Typeset",MathJax.Hub,"MathExample"]);
```

MathJax ће испроцесирати DOM елемент са ID-ем **MathExample**. Ово је еквивалентно следећем коду

Листинг 16

```
1 var math = document.getElementById("MathExample");
2 MathJax.Hub.Queue(["Typeset",MathJax.Hub,math]);
```

Уколико се параметар не проследи, процесуира се цела страница.

3.3.2. MathML из генерисане математике

Да би се добио MathML код за већ генерисану математику, корисник може користити MathJax мени и кликом на поље **Show Math as -> MathML Code** добити MathML код израза.

Међутим уколико је потребно користити тај код у оквиру странице, динамичким позивом путем *JavaScript*-а, користи се екстензија **toMathML** која то омогућава.

Ова екстензија генерише текст који садржи MathML сачуван у интерном формату MathJax-а. Да би се користила екстензија, потребно је додати је у листу екстензија приликом конфигурације.

Листинг 17

```
1 MathJax.Hub.Config({
2     extensions: ["toMathML.js"]
3 });
```

Следи пример на који начин се може искористити ова екстензија.

Листинг 18

```
1 function toMathML(jax,callback) {
2     var mml;
3     try {
4         mml = jax.root.toMathML("");
5     } catch(err) {
6         if (!err.restart) {throw err}
7         return
8     }
9     MathJax.Callback.After([getMathML,jax,callback],err.restart);
10 }
```

Овим се дефинише функција којој се проследи MathJax елемент и функција каснијег позива. Функцији каснијег позива се прослеђује MathML код елемента.

Листинг 19

```
1 toMathML(jax,function (mml) {
2     alert(mml);
3 });
```


3.4. Компатибилност са веб претраживачима

Листа веб претраживача који су тестирани са MathJax-ом је дата следећом табелом.

Претраживач	Верзија	Напомена
Internet Explorer	6.0+ (Windows)	У MathJax v2.0+, IE8 и IE9 раде брже у свом IE8 и IE9 стандардном режиму него у IE7 режиму емулације.
Firefox	3.0+ (Windows, MacOS, Linux)	Firefox 2.0 није подржан. Firefox пре 3.5 користи фонт слика.
Chrome	0.3+ (Windows, MacOS, Linux)	
Safari	2.0+ (MacOSX, Windows)	
Opera	9.5+ (Windows, MacOS, Linux)	
iPad/iPhone/iTouch	Све iOS верзије	
Android stock browser	v2.1+	
BlackBerry	OS 6+	
Symbian	Подржава	Тестирано на Symbian3 емулатору
Konqueror	4.x+ (Linux)	Није тестирано у старијим верзијама
Gecko browsers		Није систематски тестирано, али би требало да је као Firefox
WebKit browsers		Није систематски тестирано, али би требало да је као Safari

Табела 1 Листа веб претраживача тестираних са MathJax-ом

4. Семантика математичких израза

У MathML-у постоји два начина дефинисања математике. **Presentation MathML** служи за контролу изгледа израза. **Content MathML** је дизајниран да дефинише семантички математички значај и да учини изразе више разумљивим за компјутерске алгебарске системе. Елементи MathML-а `<semantics>`, `<annotation>` и `<annotation-xml>` се користе да искомбинују презентацију и садржај и да пруже оба, информације о изгледу и семантичко значење математичких израза.

Елемент `<semantics>` се понаша као елемент који обавија елементе анотације и мора имати елементе нижег нивоа. Семантичке информације у не XML формату, као што су TeX и AsciiMath, се налазе под елементом `<annotation>`. Елементи који су у XML формату су обавијени елементом `<annotation-xml>` за семантику дефинисану са Content MathML-ом и OpenMath-ом.

4.1. Content MathML

Потешкоће везане за спону презентационог кода са семантичким долазе из чињенице да постоји „више према један“ мапирање из презентационог кода у семантички и обратно. На пример, математичка конструкција „ H помножено са e “ се у већини случајева дефинише користећи експлицитни оператор као у $H \times e$. У различитим презентационим контекстима овај оператор може бити невидљив, где се множење изговара, а не пише. Али такав израз He се може двосмислено протумачити као један хемијски елемент и као математички производ две променљиве. Такође, нотације варирају у различитим културама и географским регионима.

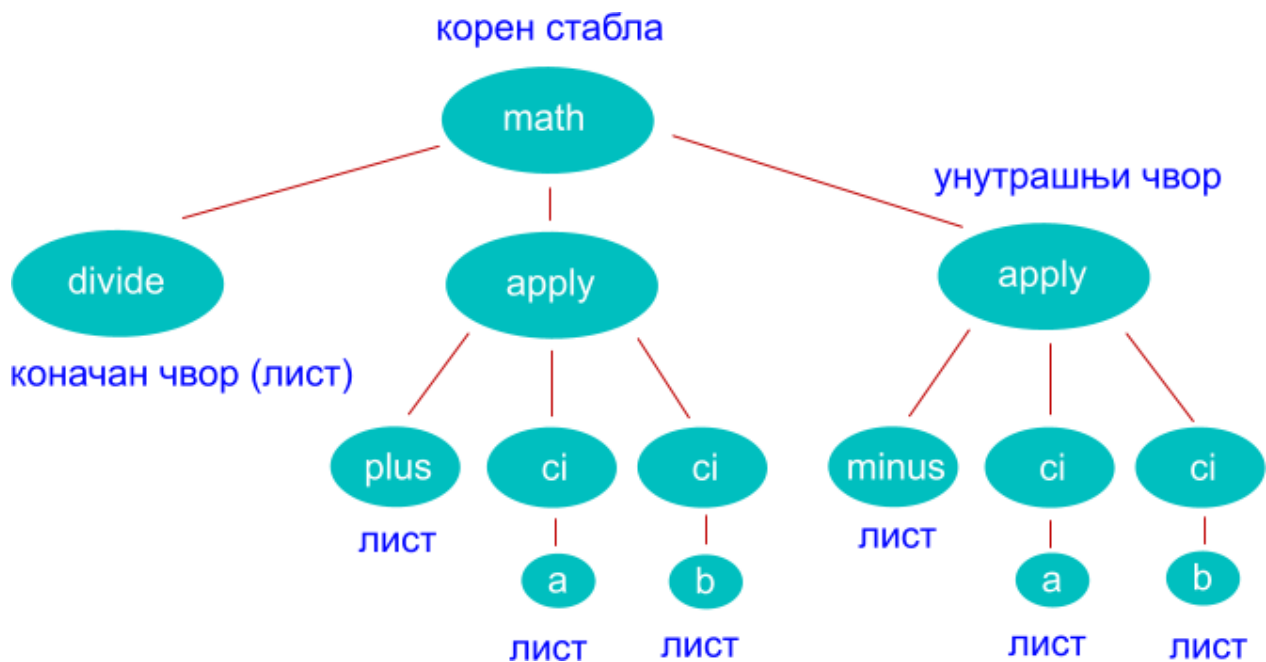
Без обзира како је презентована математичка структура визуелно, могуће је разменити информације прецизно између система који семантички процесирају математичке објекте. У битне програме спадају рачунарски алгебарски системи, индустријске и научне апликације, вишејезични системи превођења, математичке претраге, и интерактивне свеске.

4.1.1. Структура и обим израза

Content MathML представља математичке објекте као *стабла израза*. Нотација конструисања општег стабла израза је примена оператора на подобјекте. На пример, сума $x + y$ може да се дефинише као примена оператора збира на два аргумента, x и y . Исто тако $\cos(\pi)$ се може дефинисати као примена косинуса на број π . Content MathML користи префиксну нотацију где се прво наведе оператор, а затим низ аргумената где тај оператор делује на један, два или више наведених аргумената. Израз $x + y$, користећи префиксну нотацију се дефинише као $+ x y$. За сложенији израз који има више оператора и аргумената

као на пример $\frac{(x-y) \cdot (y-z)}{(x+y) \cdot (y+z)}$, користећи префиксну нотацију, израз се дефинише као $/ (* (-x y) (-y z)) (* (+ x y) (+ y z))$.

Као опште правило, коначни чворови (листови) у стаблу представљају основне математичке објекте као што су бројеви, променљиве, аритметичке операције и слично. Унутрашњи чворови стабла представљају примену функције, или друге математичке конструкције, која гради сложене објекте. Унутрашњи чвор може да представља примену функције на више аргумената, где су они сами представљени чворовима испод унутрашњег чвора. За израз $\frac{a+b}{a-b}$, стабло израза које формирају елементи на основну Content MathML кода изгледа овако:



Слика 8 Пример стабла израза Content MathML-а

Код за претходни израз, на основу ког се формира ово стабло израза изгледа овако:

Листинг 20

```

1  <math>
2      <divide/>
3      <apply>
4          <plus/> <ci>a</ci> <ci>b</ci>
5      </apply>
6      <apply>
7          <minus/> <ci>a</ci> <ci>b</ci>
8      </apply>
9  </math>

```

Семантика општих математичких израза није ствар консензуса. Број дефинисаних математичких израза никада не може бити коначан. Због проширивости математике, био би немогућ посао систематског кодирања свих математичких израза. Уместо тога, MathML формира експлицитни, мали број уобичајених математичких елемената, који су пажљиво одабрани тако да омогућавају формирање сложених израза.

4.1.2. Strict Content MathML

У MathML-у 3, постоји подниз Content MathML-а који је дефинисан као *Strict Content MathML*. Овај подниз користи, минимални, али довољан низ елемената да представи значење математичког израза, док је потпуна граматика Content MathML-а компатибилна са MathML-ом 2.0.

Content MathML пружа велики број предефинисаних функција кодираних као празни елементи (нпр. `sin`, `log`, ...) и велики број конструкција за формирање сложених објеката (нпр. `set`, `interval`, ...). Strict Content MathML користи један елемент (`csymbol`) са атрибутом који показује на екстерну дефиницију у проширивим речницима садржаја за представљање свих функција, и користи само `apply` и `bind` за прављење сложених објеката. Елементи као што су `ci` и `cn` се такође сматрају делом Strict Content MathML-а, али са ограниченим низом атрибута и са садржајем ограниченим на текст. Strict Content MathML је дизајниран да буде компатибилан са OpenMath-ом. OpenMath је стандард за представљање формалних математичких објеката и семантике кроз употребу проширивих речника садржаја (Content Dictionaries). Следећом табелом је показан однос елемената између OpenMath-а и Strict Content MathML-а.

Strict Content MathML	OpenMath
cn	OMI, OMF
csymbol	OMI, OMF
ci	OMV
cs	OMSTR
apply	OMA
bind	OMBIND
bvar	OMBVAR
share	OMR
semantics	OMATTR
annotation, annotation-xml	OMATP, OMFOREIGN
cerror	OME
cbytes	OMB

Табела 2 Одговарајући елементи Strict Content MathML-а и OpenMath-а

4.1.3. Речници садржаја (Content Dictionaries - CDs)

Због природе математике, било која метода за дефинисање значења математичких израза мора бити проширива. Кључ за проширивост је могућност дефинисања нових функција и других симбола. Да би се ово урадило, две ствари су потребне, механизам за представљање симбола који нису већ дефинисани у Content MathML-у, и начини асоцирања одређеног математичког значења са тим симболима на недвосмислен начин.

Речници садржаја су структурирани документи за дефиницију математичких концепата. Да би се максимизовала његова модуларност и поновна употреба, речник садржаја садржи релативно малу колекцију дефиниција за блиске концепте. Друштво OpenMath-а одржава велики низ јавних речника садржаја укључујући MathML CD групу која садржи дефиниције за све предефинисане симболе у MathML-у. Постоји процес за допринос самостално развијених речника садржаја репозиторијуму друштва OpenMath-а за олакшано истраживање и јавну употребу. MathML 3 не тражи да речници садржаја буду јавно доступни, али се препоручује да буду.

Битно је напоменути да су тренутни подаци од информативног значаја, а не стандард. Тачна математичка семантика предефинисаних симбола није у потпуности одређена у препоруци MathML-а 3. Семантичке дефиниције пружене са OpenMath CDs има намену да буде довољна за већину апликација. Углавном су компатибилне са семантиком одређеном за конструкције у препоруци MathML-а 2.0. Међутим, у контекстима где је неопходна доста прецизна семантика (комуникација између рачунарских алгебарских система, са формалним системима као што су доказивачи теорема и слично), одговорност пада на релевантну праксу корисника који би потврдили, проширили или заменили дефиниције пружене у OpenMath CDs као ваљане.

4.1.4. Елементи Content MathML-а за кодирање структуре израза

Елементи помоћу којих се формирају сложени изрази су подељени у више група:

- основни изрази, тј. бројеви, текстуални литерали, симболи и идентификатори
- изведени изрази, тј. примена функција и везивање израза
- семантичке анотације
- грешке

4.1.4.1 Бројеви (Content Numbers) <cn>

Елемент **cn** је елемент Content MathML-а који служи за представљање бројева. Strict Content MathML подржава целе бројеве, реалне и бројеве са зарезом са дуплом тачношћу. У овим типовима бројева, садржај овог елемента је текст. Додатно, овај елемент подржава рационалне бројеве и комплексне бројеве у којима различити делови су одвојени користећи **sep** елемент. У *non-Strict Content MathML*-у цели бројеви се могу представити помоћу различитих база.

Листинг 21

```
1 <cn base="16">7FE0</cn>
```

Користећи **e-notation** реални бројеви се могу представити у научној нотацији овог типа. Такви бројеви се састоје из два дела раздвојени `<sep/>` елементом. Први део је реални број, док други део представља цео број као експонент над базом од 10. Број $12.3 \cdot 10^5$ тј. 12.3e5 се дефинише са

Листинг 22

```
1 <cn type="e-notation">12.3<sep/>5</cn>
```

Рационални бројеви су дати са два цела броја тј. бројиоцем и имениоцем, подељени елементом сепарације.

Листинг 23

```
1 <cn type="rational">22<sep/>7</cn>
```

Комплексни бројеви су дати са два броја тј. реални и имагинарни део, подељени елементом сепарације.

Листинг 24

```
1 <cn type="complex-cartesian"> 12.3 <sep/> 5 </cn>
```

У новијим верзијама MathML-а су уведени посебни празни елементи којима се представљају константе (на пример `<pi/>` за добро познату константу π).

4.1.4.2 Идентификатори садржаја (Content Identifiers) `<ci>`

Content MathML користи **ci** елементе за конструисање променљивих. *Content Identifiers* представљају математичке променљиве које имају особине, али не и фиксiranу вредност. Овај елемент користи **type** атрибут да ближе одреди основни тип објекта који представља. У Strict Content MathML-у низ дозвољених вредности су *integer*, *rational*, *real*, *complex*, *complex-polar*, *complex-cartesian*, *constant*, *function*, *vector*, *list*, *set* и *matrix*.

Листинг 25

```
1 <ci type="integer">n</ci>
```

У non-Strict Content MathML-у дозвољена је било која текстуална вредност за атрибут.

Листинг 26

```
1 <ci type="T">n</ci>
```

```
1 <semantics>
2   <ci>n</ci>
3   <annotation-xml cd="mathmltypes" name="type" encoding="MathML-
  Content">
4     <ci>T</ci>
5   </annotation-xml>
6 </semantics>
```

4.1.4.3 Символи садржаја (Content Symbols) `<csymbol>`

Елемент `csymbol` се користи да се покаже на одређени, математички дефинисани концепт са екстерном дефиницијом. У изразу $x + y$, знак плус је симбол, пошто има тачно одређену, екстерну дефиницију, тј. функцију збира. Елементарне функције и уобичајени математички оператори су сви примери симбола.

Показивач на екстерну дефиницију је URI¹⁵. Најбитнији међу репозиторијумима математичких дефиниција је репозиторијум речника садржаја друштва OpenMath-а. У MathML-у 3, OpenMath CDs су преферирани извор екстерних дефиниција. Постоје два начина за дефинисање показивача на екстерне дефиниције или речнике садржаја. Користећи `cd` атрибут (који прати конвенцију утврђену за OpenMath) врши се повезивање са речницима садржаја. Други начин, користећи `definitionURL` атрибут, може се приступити било којем извору дефиниција које су једнозначно одређене.

Уколико се користи OpenMath CDs, преферира се метода са `cd` атрибутом. OpenMath дефиниције симбола се идентификују са три вредности, име симбола, име речника садржаја, и база речника садржаја, тј. URI који једнозначно одређује речник садржаја истог имена. Да би такав триплет повезали са `csymbol`, садржај елемента специфицира име симбола, и име речника садржаја користећи `cd` атрибут. База се одређује на три начина. Из документа који садржи `math` елемент у ком је `csymbol` користећи механизам дат форматом документа. Или користећи `cdgroup` атрибут, који се опционо дефинише. Без посебно дефинисане информације подразумева се да је <http://www.openmath.org/cd> база за све елементе `csymbol`. Израз $ax + b$ се представља

Листинг 27

```

1  <apply>
2      <csymbol cd="arith1">plus</csymbol>
3      <apply>
4          <csymbol cd="arith1">times</csymbol>
5          <ci>a</ci>
6          <ci>x</ci>
7      </apply>
8      <ci>b</ci>
9  </apply>
```

4.1.4.4 Текстуални литерали (Content String) `<cs>`

Елемент `cs` кодира текстуалне литерале који се могу користити у изразима Content MathML-а. Садржај овог елемента је текст. Код за скуп елемената $\{A, B\}$ изгледа овако

Листинг 28

```

1  <set>
2      <cs>A</cs>   <cs>B</cs>   <cs>   </cs>
3  </set>
```

¹⁵ Низ карактера који идентификују име ресурса. Таква идентификација омогућава интеракцију са ресурсом преко мреже, углавном World Wide Web-а, користећи одређене протоколе.

4.1.4.5 Примена функција `<apply>`

Фундаментални начин креирања сложеног објекта у математици је применом функције или операције на неке аргументе.

Резултујуће стабло одговара комплетном математичком стаблу, што значи да део математике који може бити ограничен са „логичким“ заградама без промене његовог значења може да се формира применом апликације или операције на његове аргументе користећи елемент `apply`. На пример $(x + y)$ се може кодирати са

Листинг 29

```
1 <apply>
2   <csymbol cd="arith1">plus</csymbol>
3   <ci>x</ci>
4   <ci>y</ci>
5 </apply>
```

Отварајући и затварајући тагови елемента `apply` тачно дефинишу домен операције или функције. Символично, садржајни модел се може описати са `<apply> op [a b ...] </apply>` где су операнди `a`, `b`, `...` MathML стабла израза, а `op` је MathML стабло израза које представља операцију или функцију. Овај елемент може имати било који број операнда, на пример за $(x + y + z)$ код изгледа овако

Листинг 30

```
1 <apply><csymbol cd="arith1">plus</csymbol>
2   <ci>x</ci>
3   <ci>y</ci>
4   <ci>z</ci>
5 </apply>
```

MathML такође дозвољава примену функција и без операнда, на пример за функције као што су `random` или `current-date`. Нема потребе за додавањем заграда, или да се оператор дефинише пре како би се израз ваљано парсирао. Сваки израз окружен `apply` елементом је добро дефинисан објекат чија интерпретација не зависи од окружујућег контекста. Ово је у великој мери супротно од презентационог дефинисања, где **исти изрази могу имати веома различита значења** у различитим контекстима. На пример, израз са презентационим изгледом $(F + G)(x)$ може бити производ као у

Листинг 31

```
1 <apply>
2   <csymbol cd="arith1">times</csymbol>
3   <apply>
4     <csymbol cd="arith1">plus</csymbol>
5     <ci>F</ci>
6     <ci>G</ci>
7   </apply>
8   <ci>x</ci>
9 </apply>
```


Исти тај израз може представљати примену функције $F + G$ са аргументом x . Ово је дефинисано конструисањем суме и њеном применом на аргумент

Листинг 32

```

1 <apply>
2   <apply>
3     <csymbol cd="arith1">plus</csymbol>
4     <ci>F</ci>
5     <ci>G</ci>
6   </apply>
7   <ci>x</ci>
8 </apply>

```

У оба случаја, интерпретација примене је експлицитна и недвосмислена, и не мења се без обзира где се израз користи.

4.1.4.6 Везивање `<bind>` и везивне променљиве `<bvar>`

Многи сложени математички изрази се конструишу користећи *везивне променљиве*, које представљају важан концепт логике и формалних језика. Везивна променљива се карактерише особином да систематичким преименовањем променљиве (у име које се није још увек појавило у изразу) се не мења значење израза.

Везивање израза је дефинисано са MathML стаблом израза користећи `bind` елемент. Његов први подчвор је MathML израз који представља *везивни оператор*, а након тога следи непразна листа `bvar` елемената који дефинишу везивне променљиве, и последњи подеlement је општи Content MathML израз, познат као *тело везивања*.

Елемент `bvar` се користи да се дефинише везивна променљива везивног израза (на пример у сумама, производима и слично). Везивне променљиве се идентификују упоређивањем имена. Таква идентификација може се експлицитно навести у `id` атрибуту `ci` елемента у `bvar` елементу и позивајућу се на њу користећи `xref` атрибут на све друге инстанце. Израз $(\forall x)(x < 1)$ се кодира овако

Листинг 33

```

1 <bind>
2   <csymbol cd="quant1">forall</csymbol>
3   <bvar><ci id="var-x">x</ci></bvar>
4   <apply>
5     <csymbol cd="relation1">lt</csymbol>
6     <ci xref="var-x">x</ci>
7     <cn>1</cn>
8   </apply>
9 </bind>

```

- везивни оператор
 - везивне променљиве
 } - тело везивања

Ово је нарочито корисно када су конструкције са везивним променљивама угњежене. У оба случаја, интерпретација примена је експлицитна и недвосмислена, и не мења се без обзира где се израз користи.

4.1.5. Одређени оператори и константе Content MathML-a

Главни низ математичких оператора, функција и константи углавном је дефинисан празним елементима, покривајући општу математику. Неке од **функција** и **релација** су:

- **inverse** – елемент који се примењује на функцију да би се конструисао израз који представља инверзију функције

Листинг 34

```

1 <apply>
2   <inverse/>
3   <ci> f </ci>
4 </apply>
```

 $f^{(-1)}$

- **compose** – представља оператор композиције функција
- **factorial** – елемент представља унарни оператор факторијел не негативних целих бројева
- **divide** – оператор дељења
- **max** – елемент који дефинише функцију максимума, која враћа највећи од аргумената
- **min** – елемент који дефинише функцију минимума, која враћа најмањи од аргумената
- **minus** – елемент који може да се користи као унарни аритметички оператор за $-x$ или као бинарни аритметички оператор за $x - y$
- **plus** – операција збира
- **power** – елемент представља експоненцијални оператор где је први аргумент степенован на други
- **rem** – представља операцију модула, и враћа остатак приликом поделе првог аргумента са другим
- **times** – n -арни оператор производа
- **root** – елемент који се користи за добијање корена, док се степен корена даје помоћу **degree** елемента као први аргумент, и уколико степен није наведен, подразумева се да се тражи квадратни корен
- **gcd** – n -арни оператор који враћа највећи заједнички делилац својих аргумената
- **lcm** – n -арни оператор који враћа најмањи заједнички садржалац својих аргумената
- **and** – елемент који представља конјункцију тј. n -арну функцију која за аргументе добија тачну и/или нетачну вредност, тачно је ако су сви аргументи тачни иначе је нетачно
- **or** – елемент који представља дисјункцију, тачно је ако је барем један аргумент тачан иначе је нетачно

- **xor** – елемент који представља ексклузивну дисјункцију, тачно је ако је непаран број тачних аргумената иначе је нетачно
- **not** – елемент који представља негацију, враћа обрнуту вредност аргумента
- **implies** – елемент који представља импликацију која узима два аргумента, нетачно је ако је први аргумент тачан а други нетачан иначе је тачно
- **equivalent** – представља еквиваленцију која узима два аргумента, тачно је уколико су логичке вредности оба аргумента исте иначе је нетачно
- **forall** – користи се са везивним променљивама и значи „за свако“

Листинг 35

```

1  <bind>
2      <forall/>
3      <bvar><ci>x</ci></bvar>
4      <apply>                                 $\forall x.(x - x = 0)$ 
5          <eq/>
6          <apply><minus/><ci>x</ci><ci>x</ci></apply>
7          <cn>0</cn>
8      </apply>
9  </bind>

```

- **exists** – користи се са везивним променљивама и значи „постоји“
- **abs** – елемент који представља функцију апсолутне вредности
- **floor** – операција која заокружује број ка најближем мањем целом броју
- **ceiling** – операција која заокружује број ка најближем већем целом броју
- **eq** – представља релацију једнакости и може се користити са више аргумената иако је бинарна релација дефинишући тако низ једнакости
- **neq** – бинарна релација неједнакости која враћа логичку вредност тачно уколико два елемента нису једнака
- **gt** – функција која враћа тачно уколико је први аргумент већи од другог иначе враћа нетачно, такође може да се користи са више аргумената где проверава да ли је први аргумент већи од осталих
- **lt** – функција која враћа тачно уколико је први аргумент мањи од другог иначе враћа нетачно, такође може да се користи са више аргумената где проверава да ли је први аргумент мањи од осталих

Листинг 36

```

1  <apply>
2      <lt/>
3      <cn>2</cn>
4      <cn>3</cn>
5      <cn>4</cn>
6  </apply>                                 $2 < 3 < 4$ 

```

- **geq** – функција која враћа тачно уколико је први аргумент већи или једнак од другог иначе враћа нетачно, такође може да се користи са више аргумената где проверава да ли је први аргумент већи или једнак од осталих
- **leq** – функција која враћа тачно уколико је први аргумент мањи или једнак од другог иначе враћа нетачно, такође може да се користи са више аргумената где проверава да ли је први аргумент мањи или једнак од осталих
- **factorof** – елемент који служи да покаже математичку везу да ли је први аргумент чинилац другог аргумента
- **int** – елемент је оператор за одређени или неодређени интеграл

Листинг 37

```

1  <apply>
2      <int/>
3      <bvar><ci>x</ci></bvar>
4      <lowlimit><cn>0</cn></lowlimit>
5      <uplimit><cn>1</cn></uplimit>
6      <apply><power/><ci>x</ci><cn>2</cn></apply>
7  </apply>

```

$$\int_0^1 x^2 dx$$

- **diff** – оператор диференцијације за функције или изразе у једној променљивој
- **partialdiff** – оператор парцијалне диференцијације за функције или изразе у више променљивих

За **скупове** се користе следећи елементи:

- **set** – представља функцију која конструише математички скуп од својих аргумената
- **list** – елемент представља n -арну функцију која конструише листу од својих аргумената, и разликује се од скупова по томе што је битан редослед елемената
- **union** – дефинише n -арну унију скупова
- **intersect** – дефинише n -арни пресек скупова
- **in** – представља релацију припадања скупу, има два аргумента, елемент и скуп
- **notin** – представља релацију неприпадања скупу, има два аргумента, елемент и скуп
- **subset** – представља релацију подскупа, дефинише да је први аргумент подскуп другог, такође може бити n -арни оператор да се изрази да је сваки аргумент подскуп свог следбеника
- **prsubset** – представља релацију правог подскупа, дефинише да је први аргумент прави подскуп другог, такође може бити n -арни оператор да се изрази да је сваки аргумент прави подскуп свог следбеника

- **notsubset** – представља обрнуту релацију подскупа, дефинише да први аргумент није подскуп другог
- **notprsubset** – представља обрнуту релацију правог подскупа, дефинише да први аргумент није прави подскуп другог
- **setdiff** – представља оператор разлике, узима два аргумента и дефинише скуп који садржи све елементе који се појављују у првом али не и у другом скупу
- **card** – елемент који представља функцију кардиналности, која узима скуп као аргумент и враћа његову кардиналност, тј. број елемената у скупу
- **cartesianproduct** – елемент који се користи за представљање оператора Декартовог производа, узима скупове као аргументе

Неки од **секвенцијалних** и **серијских** елемената су:

- **sum** – представља n -арни оператор збира, може се користити са или без везивних променљивих и уколико се користе, овај елемент прате један или више **bvar** елемената дајући индексне променљиве, и затим је дат домен за индексне променљиве, где се након тога дефинише израз који се сумира за сваку тачку домена индексних променљивих

Листинг 38

```

1 <apply>
2   <sum/>
3   <bvar><ci>x</ci></bvar>
4   <lowlimit><ci>a</ci></lowlimit>
5   <uplimit><ci>b</ci></uplimit>
6   <apply><ci>f</ci><ci>x</ci></apply>
7 </apply>

```

$$\sum_{x=a}^b f(x)$$

```

1 <apply>
2   <sum/>
3   <bvar><ci>x</ci></bvar>
4   <condition>
5     <apply><in/><ci>x</ci><ci>
type="set">B</ci></apply>
6   </condition>
7   <apply><ci>f</ci><ci>x</ci></apply>
8 </apply>

```

$$\sum_{x \in B} f(x)$$

```

1 <apply>
2   <sum/>
3   <domainofapplication>
4     <ci>B</ci>
5   </domainofapplication>
6   <ci>f</ci>
7 </apply>

```

$$\sum_B f$$

- **product** – представља n -арни оператор производа, може се користити са или без везивних променљивих и уколико се користе, овај елемент прате један или више **bvar** елемената дајући индексне променљиве, и затим је дат домен за индексне променљиве, где се након тога дефинише израз који се множи за сваку тачку домена индексних променљивих

Листинг 39

```

1 <apply>
2   <product/>
3   <bvar><ci>x</ci></bvar>
4   <lowlimit><ci>a</ci></lowlimit>
5   <uplimit><ci>b</ci></uplimit>
6   <apply>
7     <ci type="function">f</ci>
8     <ci>x</ci>
9   </apply>
10 </apply>

```

$$\prod_{x=a}^b f(x)$$

- **limit** – елемент који представља лимес оператор, изражен је дефинисањем **lowlimit** и **bvar**, или дефинисањем **condition** на једну или више везивних променљивих

Листинг 40

```

1 <apply>
2   <limit/>
3   <bvar><ci>x</ci></bvar>
4   <lowlimit><cn>0</cn></lowlimit>
5   <apply><sin/><ci>x</ci></apply>
6 </apply>

```

$$\lim_{x \rightarrow 0} \sin x$$

```

1 <apply>
2   <limit/>
3   <bvar><ci>x</ci></bvar>
4   <condition>
5     <apply><tendsto/><ci>x</ci><cn>0</cn></apply>
6   </condition>
7   <apply><sin/><ci>x</ci></apply>
8 </apply>

```

$$\lim_{x \rightarrow 0} \sin x$$

- **tendsto** – користи се да се изрази релација тежње ка одређеној вредности, углавном се користи у оквиру лимеса

Листинг 41

```

1 <apply>
2   <tendsto type="above"/>
3   <apply><power/><ci>x</ci><cn>2</cn></apply>
4   <apply><power/><ci>a</ci><cn>2</cn></apply>
5 </apply>

```

$$x^2 \rightarrow a^{2+}$$

Неки од термина **линеарне алгебре** вектор, матрица, детерминанта, функција за транспоновање матрице или вектора, векторски производ, скаларни производ су дати елементима `vector`, `matrix`, `determinant`, `transpose`, `vectorproduct`, `scalarproduct` респективно.

Листинг 42

```

1  <matrix>
2    <bvar><ci type="integer">i</ci></bvar>
3    <bvar><ci type="integer">j</ci></bvar>
4    <condition>
5      <apply>
6        <and/>       $[m_{i,j} | m_{i,j} = i^j; i \in [1, 5] \wedge j \in [5, 9]]$ 
7        <apply>
8          <in/>
9          <ci>i</ci>
10         <interval><ci>1</ci><ci>5</ci></interval>
11       </apply>
12       <apply>
13         <in/>
14         <ci>j</ci>
15         <interval><ci>5</ci> <ci>9</ci></interval>
16       </apply>
17     </condition>
18     <apply><power/><ci>i</ci><ci>j</ci></apply>
19 </matrix>
20 <apply>
21   <determinant/>
22   <ci type="matrix">A</ci>       $\det A$ 
23 </apply>

```

```

1  <apply>
2    <eq/>
3    <apply>
4      <vectorproduct/>
5      <ci type="vector"> A </ci>
6      <ci type="vector"> B </ci>
7    </apply>
8    <apply>
9      <times/>       $A \times B = ab \sin \theta N$ 
10     <ci>a</ci>
11     <ci>b</ci>
12     <apply>
13       <sin/><ci>&#x3b8;</ci>
14     </apply>
15     <ci type="vector"> N </ci>
16   </apply>
17 </apply>

```

Уобичајене **тригонометријске функције** су дате елементима `sin`, `cos`, `tan`, `sec`, `csc`, `cot` и **инверзне тригонометријске функције** `arcsin`, `arccos`, `arctan`, `arcsec`, `arccsc`, `arccot`.

За **експоненцијалне функције**, користи се елемент `exp` а за **природни логаритам** `ln` и оба елемента имају један аргумент. Елемент `log` представља функцију **логаритма** са датом базом, која уколико није наведена са `logbase` елементом, се подразумева да је 10.

Статистички термини као што су средња вредност, стандардна девијација, варијанса, медијана се дефинишу елементима `mean`, `sdev`, `variance`, `median` респективно.

Скупови бројева \mathbb{Z} , \mathbb{R} , \mathbb{Q} , \mathbb{C} , \mathbb{N} , \mathbb{P} су респективно описани елементима `integer`, `reals`, `rational`s, `complexes`, `naturalnumbers`, `primes`.

Неке од **константи** и **уобичајених вредности** се дефинишу са елементима:

- `exponentiale` – база природног алгоритма, $\ln e = 1$, приближна вредност 2,718
- `imaginaryi` – имагинарно „i“, $i^2 = -1$
- `notanumber` – непостојећи број, $0/0 + NaN$
- `true` – логичка константа за тачно, \top
- `false` – логичка константа за нетачно, \perp
- `emptyset` – празан скуп, скуп без елемената, $\mathbb{Z} \neq \emptyset$
- `pi` – однос обима и пречника круга, π , приближна вредност 3,142
- `eulergamma` – гама константа, γ , приближна вредност 0,5772
- `infinity` – бесконачност, ∞

5. Резултати истраживања

MathML је још у почетку замишљен тако да може да:

- Погодно, за сву едукациону и научну комуникацију, кодира математички материјал.
- Кодира и математичку нотацију и математичко значење.
- Олакшава превођење из и у друге математичке формате, и презентационе и семантичке, и излазни формати би требало да садрже:
 - графички приказ
 - погодну организацију за говорно читање израза
 - друге математичке језике, као што је TeX
 - стандардни приказ текста
 - интернационално штампање, укључујући и Брајеву азбуку

Познато је да превођење из и у друге нотационе системе може имати губитке информација у процесу.

- Омогућава прослеђивање специјалних информација за одређене апликације.
- Пружа подршку за ефикасан рад са обимнијим изразима.
- Омогућава проширивост.
- Одговара уобичајеним техникама за генерисање формула.
- Буде читљив људима, као и софтверима за генерисање и процесирање.

Без обзира колико успешно MathML испуњава своје циљеве, јасно је да је MathML користан само уколико је добро имплементиран. Додатни циљеви имплементације везани су за минималну функционалност које софтвер за процесирање и обраду MathML-а треба да пружи:

- са највећим квалитетом могућим, веб претраживач треба ваљано да обради MathML изразе у HTML (и XHTML) страницама
- HTML (и XHTML) документи који садрже MathML изразе се морају штампати са високим квалитетом
- MathML изрази у веб страницама треба да буду интерактивни
- едитори и преводиоци математичких израза морају бити развијени да олакшају формирање веб страница које садрже MathML изразе

Домен у ком су ови циљеви испуњени зависи од сарадње и подршке компанија које развијају веб претраживаче и других програмера. Нажалост, подршка веб претраживача је заступљена, приближно, у само 25% верзија веб претраживача који се данас користе. У тих 25%, подршка није потпуна. Појављује се доста кварова. Компаније које се тиме баве имају доста проблема. Компанија *Google*, која развија веб претраживач *Chrome*, је престала са

подршком за MathML и мисли да је тренутно најбоље решење које ће заменити ту подршку, било којег веб претраживача, JavaScript библиотека MathJax.

5.1. Закључци истраживања

Замену подршке веб претраживача за MathML нуде разне библиотеке и софтверске апликације (*MathPlayer*, *jqMath*, *LaTeXML*, ...) али MathJax заиста јесте најбоље решење. Ова библиотека садржи скуп разних библиотека које су програмери ван MathJax тима самостално развијали. MathJax тим, поред свог великог доприноса, је успео да интегрише те друге библиотеке и споји у једну вишефункционалну библиотеку. Ова библиотека је бесплатна за коришћење и отвореног је кода. Сваки модул у оквиру библиотеке се може искористити и даље развијати. Документација за MathJax је пристојна, нуди основне и битне информације са једноставним примерима и добро објашњеним упутствима. Уколико се залази у детаље, мало је теже наћи све одговоре, па је потребно темељно читање документације. MathJax пружа подршку и превођење за разне математичке нотације (*AsciiMath*, *TeX*, *LaTeX*, *MathML*) и пружа више излазних формата (*HTML-CSS*, *NativeMML*, *SVG*) са високим квалитетом. Има потпуну подршку за већину данашњих популарних веб претраживача. Све ово доказује да се MathJax може искористити да испуни тражене функционалности пројектног задатка.

Пошто се помињу и друге математичке нотације, добро је и њих на неки начин додати у оквиру пројектног задатка. Presentation MathML треба да буде генерисан од стране неког софтвера, па ће се ова нотација користити као излазни формат апликације *Генератор излаза*. Такође биће омогућен приказ Presentation MathML кода као и његово укључивање у друге документе. LaTeX и AsciiMath нотација имају сличну синтаксу кода али је AsciiMath доста једноставнији и природнији за корисника. Да би се показала веза између свих нотација, LaTeX нотација ће бити додата иако није неопходна за пројектни задатак.

Иако MathJax библиотека пружа велики број превођења између математичких нотација, нажалост, не пружа превођење у Content MathML или у било који други језик за чување семантичког значења. Један од разлога је тај што не постоји добар начин за превођење, а да се при процесу превођења не изгубе неке информације. Рецимо, да се у математичком изразу налази слово F . Ово слово може да има више значења, као што су функција у математици, сила у физици, флуор у хемији. Иако има исти визуелни приказ, има различито значење. При превођењу, то значење се не може јединствено дефинисати. Постоји неколико библиотека које врше превођење у математичку нотацију која чува семантичко значење математичког израза. Углавном су то библиотеке које нису намењене за веб, већ служе за десктоп апликације. Превођење се базира на претпоставци значења појединих делова израза. Због недостатка доброг превођења, треба посебно дефинисати код за чување семантичког значења. Апликација која би генерисала такав код мора бити специјално дизајнирана за потребе корисника. Како то уопште није циљ пројектног задатка, корисник ће морати сам да дефинише код за чување семантичког значења математичког израза. Језици у ком се овај код може дефинисати су Content MathML и OpenMath.

OpenMath има синтаксу која захтева боље познавање речника садржаја па је Content MathML логичнији избор.

За бојење кода, које даје значај на изгледу, и едитор за лакше кодирање користиће се додатне библиотеке.

Клијентски део сајта биће имплементиран користећи HTML, CSS, JavaScript и jQuery¹⁶. За структуру и изглед сајта користиће се HTML са CSS¹⁷-ом. JavaScript и jQuery омогућиће динамику сајта. Серверски део сајта кодираће се у PHP-у. За манипулацију над подацима користиће се MySQL. PHP је један од популарнијих програмских језика за веб и пружа доста функција и могућности које ће бити корисне у развијању пројектног задатка. Сајт ће бити постављен на *Apache HTTP* серверу.

¹⁶ JavaScript библиотека дизајнирана да поједностави JavaScript код

¹⁷ Cascading Style Sheets (CSS) је програмски језик за описивање презентације документа писаног у HTML и XHTML

6. Презентација пројектног задатка

Пројектни задатак се остварује кроз функционалан веб сајт који показује примену и коришћење Presentation MathML-a, Content MathML-a и библиотеке MathJax. Функционалности и начин рада сајта биће приказани и објашњени по веб страницама у оквиру сајта. Приказ страница зависи од тога да ли је корисник пријављен на сајт. *Сликама 9 и 10* су дати дијаграми структуре сајта по страницама у зависности да ли је корисник пријављен на сајт или не.



Слика 9 Дијаграм архитектуре сајта пријављеног корисника



Слика 10 Дијаграм архитектуре сајта непријављеног корисника

Registracija

E-mail

Korisničko ime

Šifra

Ponovi šifru

Prijava

Korisničko ime

Šifra

Слика 11 Формулар за регистрацију и пријаву корисника

6.1. Почетна (*home*) страница

На почетној страници сајта се налази главни део пројектног задатка, апликација *Генератор израза*. Сврха апликације је да се помоћу једноставног корисничког интерфејса, динамички формира математички израз и да се тај израз визуелно прикаже. Уколико корисник није пријављен на сајт, и даље ће моћи да користи апликацију, али неће постојати опција за чување формираног израза. Поред тога, биће му пружена могућност регистрације и пријаве на сајт, и биће му доступан линк за претрагу већ сачуваних израза.

Izrazi

Operacije Stepenovanje i indeksiranje Zgrade Relacije Simboli Logički simboli i strelice Grčka slova

$\frac{\square}{\square}$
 $\frac{\square}{M}$
 $\frac{M}{\square}$
 $+$
 $-$
 \times
 \cdot
 $/$
 \cup
 \cap
 \pm
 $\sum(\square)$
 $\prod(\square)$

\wedge
 $\bigwedge(\square)$
 \vee
 $\bigvee(\square)$
 $\bigcap(\square)$
 $\bigcup(\square)$
 $\left[\begin{smallmatrix}\square & \square \\ \square & \square\end{smallmatrix}\right]$
 \div

AsciiMath

$x_{(1,2)} = (-b \pm \sqrt{b^2 + 4ac}) / (2a)$

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 + 4ac}}{2a}$$

Registracija

E-mail

Korisničko ime

Šifra

Ponovi šifru

Prijava

Korisničko ime

Šifra

Pretraga

1. скуп дугмића са основним операторима и константама за формирање математичког израза
2. текстуално поље за код математичког израза
3. визуелни приказ математичког израза
4. дугме за преузимање изворног MathML кода формираног математичког израза

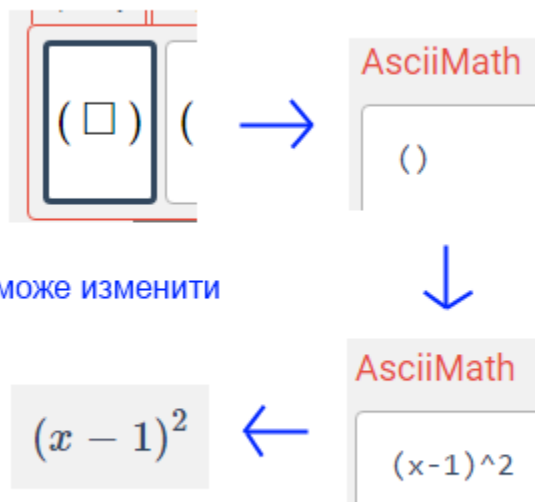
Слика 12 Почетна страница када корисник није пријављен

6.1.1. Генератор израза

Апликација *Генератор израза* треба да пружи једноставан графички интерфејс за формирање математичких израза. Потребни су дугмићи у оквиру тог интерфејса који генеришу основне изразе. Такође потребан је начин мануелне измене вредности (на пример унос бројева или ознака за променљиве). На крају је потребан визуелни приказ формираног математичког израза. Визуелни приказ израза треба да буде динамички, тј. изглед израза треба да се мења на сваку мануелну измену или клик дугмета.

Презентациони MathML је језик који треба рачунар да генерише (није намењено да човек куца код за дефинисање неког израза). Ако се то узме у обзир, као и чињеница да сложени графички интерфејс превазилази оквире и циљеве пројектног задатка, потребно је одабрати одговарајући начин за формирање математичких израза. Како библиотека MathJax, поред MathML-а пружа подршку за TeX, LaTeX и AsciiMath, то треба искористити. Једноставан начин за формирање израза може бити кодирање у неком другом језику који MathJax подржава, а није MathML. Најједноставнију синтаксу и правила за кодирање пружа **AsciiMath**. Кодирање математичких израза у овом језику је врло природно и лако га је схватити.

Кликом на дугме генерише се AsciiMath код у текстуалном пољу



Код у текстуалном пољу се може изменити

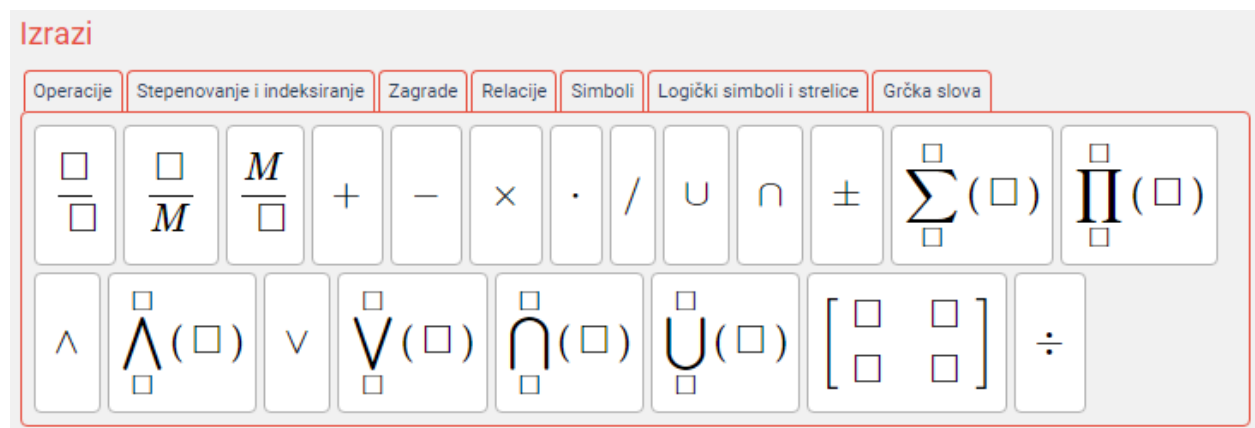
Визуелни приказ дефинисаног кода динамички формира израз

Слика 13 Процес формирања израза

Апликација ће се састојати из више компоненти које доприносе формирању и приказивању израза.

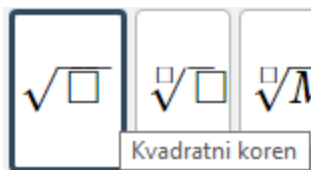
Прва компонента апликације садржи скуп дугмића основних израза уређених по категоријама:

- **операције** – скуп дугмића основних операција, као што су *разломак*, *плус*, *минус*, *векторски производ*, *подељено*, *унија*, *пресек*, *плус-минус*, *конјункција*, *дисјункција*, *сума*, *производ* и *матрица*



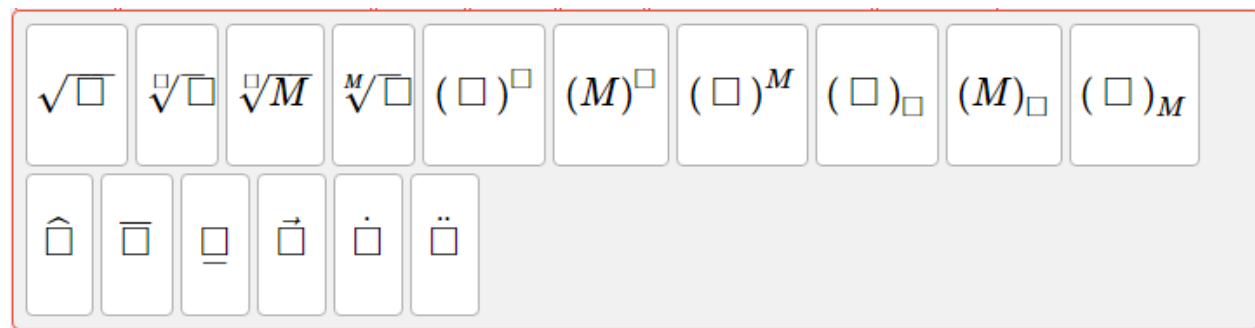
Слика 14 Операције

Када се пређе преко дугмета појави се кратак опис израза.



Слика 15 Опис израза у дугмету

- **степеновање и индексирање** – дугмићи са основним функцијама за степеновање, индексирање и обележавање као што су *корен*, *степен*, *индекс*, *надвучена* и *подвучена црта*, *векторска стрелица*, *једна* и *две тачке над*



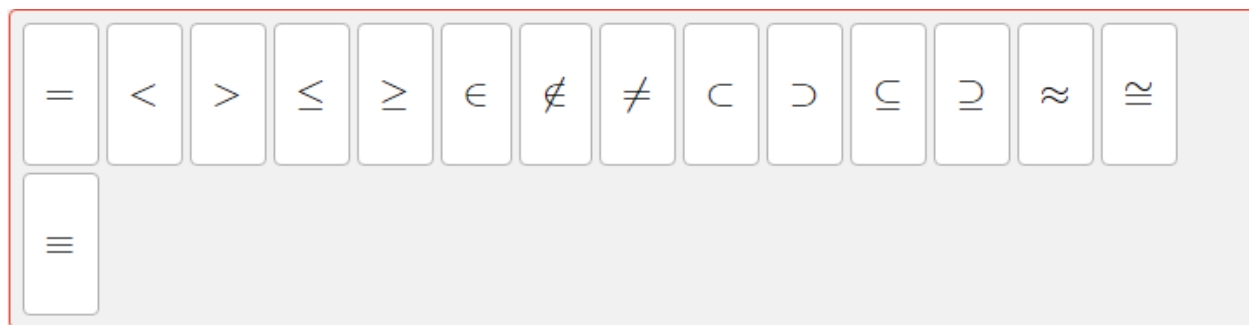
Слика 16 Степеновање и индексирање

- **заграде** – леве, десне, горње и доње заграде које могу бити *обичне, угласте, витичасте, усправне, за цео део*



Слика 17 Заграде

- **релације** – скуп релација везаних за бројеве и скупове међу којима су *једнако, мање, веће, мање-или-једнако, веће-или-једнако, припада, не припада, различито, подскуп, приближно*



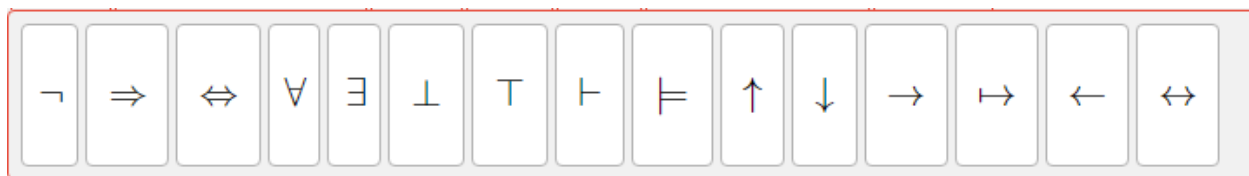
Слика 18 Релације

- **симболи** – разни математички симболи, бројевни, скуповни као и специјални симболи који представљају сложеније изразе, и у њих спадају *празан скуп, интеграл, набла, извод, бесконачно, угао, три тачке, бројевни скупови, лимес, квадратић, звезда, кружић*



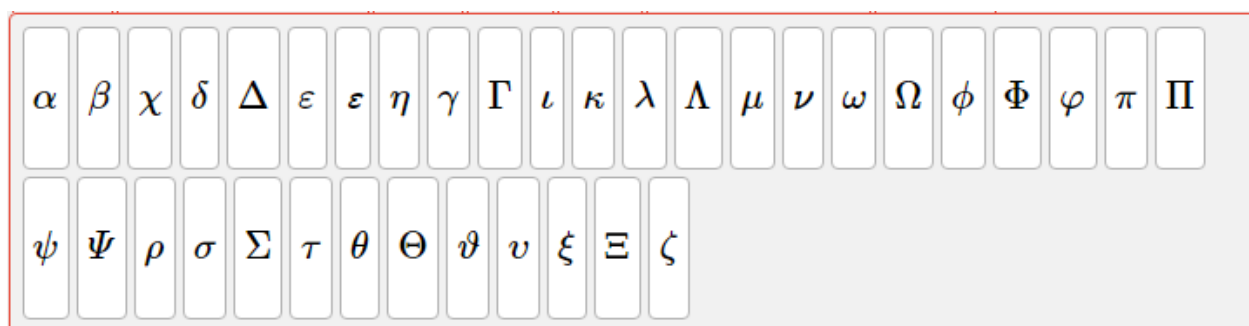
Слика 19 Симболи

- **логички симболи и стрелице** – разни логички симболи међу којима су *негација, импликација, еквиваленција, за свако, постоји, тачно, нетачно, доказ, једносмерне и двосмерне стрелице*



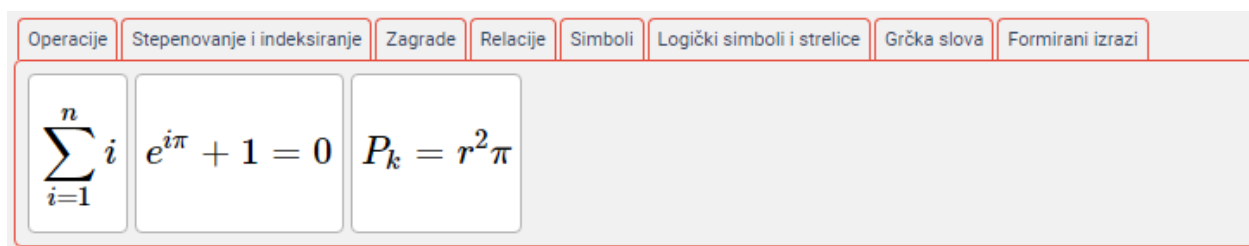
Слика 20 Логички симболи и стрелице

- **грчка слова** – међу њима су сва мала и нека велика слова



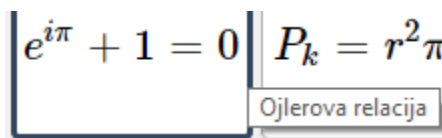
Слика 21 Грчка слова

- **формирани изрази** – ова категорија се појављује уколико је корисник пријављен и садржи скуп већ сачуваних израза које корисник по жељи додаје у ову категорију



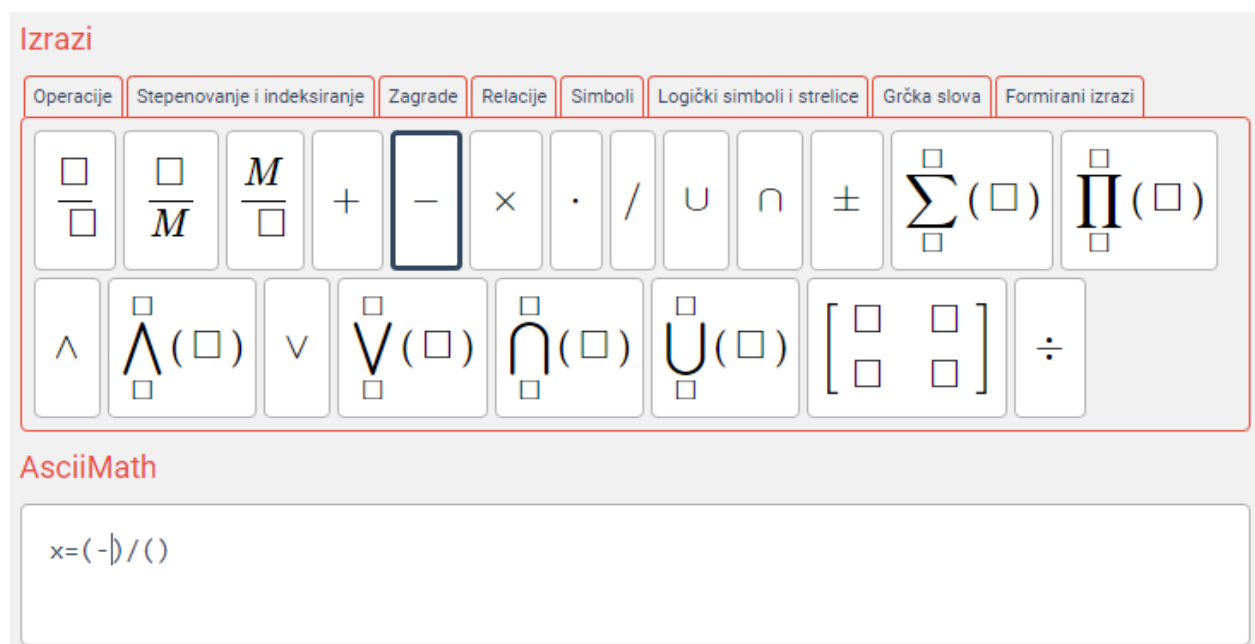
Слика 22 Формирани изрази

Када се пређе преко таквог дугмета приказаће се назив додељен том изразу.



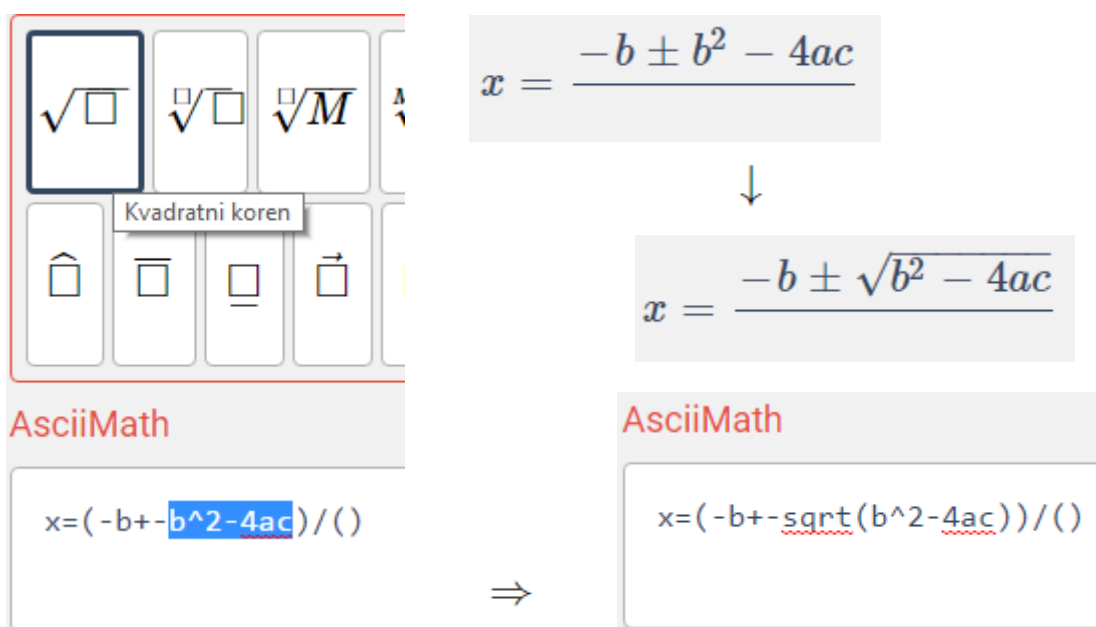
Слика 23 Опис формираног израза

Друга компонента апликације је текстуално поље у које се могу уносити карактери са тастатуре. Кликом на неко од дугмића из прве компоненте, у текстуално поље се додаје AsciiMath код за то дугме. Где ће се код додати зависи од положаја курсора у пољу.



Слика 24 Додавање AsciiMath кода у зависности од позиције курсора

Неки дугмићи у првој компоненти пружају могућност обухватања кода који селектован у другој компоненти.



Слика 25 Додавање кода селектовањем

Код у текстуалном пољу се може изменити. На сваку измену динамички се мења визуелни приказ израза. Да би корисник имао у увид команде за одређене симболе, којих нема у скупу дугмића прве компоненте, дат је списак свих команди AsciiMath-a.

Операциони симболи		Разни симболи		Грчка слова			
Команда	Приказ	Команда	Приказ				
+	+	int	\int	alpha	α		
-	-	oint	\oint	beta	β		
*	.	del	∂	chi	χ		
**	*	grad	∇	delta	δ	Delta	Δ
***	*	+-	\pm	epsilon	ϵ		
//	/	O/	\emptyset	varepsilon	ϵ		
\	\	oo	∞	eta	η		
xx	\times	aleph	\aleph	gamma	γ	Gamma	Γ
÷	÷	/_	\angle	iota	ι		
@	\circ	∴	\therefore	kappa	κ		
o+	\oplus	...	$ \dots $	lambda	λ	Lambda	Λ
ox	\otimes	cdots	$ \cdots $	mu	μ		
o.	\odot	vdots	\vdots	nu	ν		
sum	\sum	ddots	\ddots	omega	ω	Omega	Ω
prod	\prod	v	$ $	phi	ϕ	Phi	Φ
^^	\wedge	quad	$ $	varphi	φ		
^^^	\bigwedge	diamond	\diamond	pi	π	Pi	Π
vv	\vee	square	\square	psi	ψ	Psi	Ψ
vvv	\bigvee	_	$ $	rho	ρ		
nn	\cap	~	$ $	sigma	σ	Sigma	Σ
nnn	\bigcap	~	$ $	tau	τ		
uu	\cup	cc	\mathbb{C}	theta	θ	Theta	Θ
uuu	\bigcup	NN	\mathbb{N}	vartheta	ϑ		
		QQ	\mathbb{Q}	upsilon	υ		
		RR	\mathbb{R}	xi	ξ	Xi	Ξ
		ZZ	\mathbb{Z}	zeta	ζ		

Релациони симболи	
Команда	Приказ
=	$=$
!=	\neq
<	$<$
>	$>$
<=	\leq
>=	\geq
-<	\prec
>-	\succ
in	\in
lin	\notin
sub	\subset
sup	\supset
sube	\subseteq
supe	\supseteq
=	\equiv
~	\cong
~	\approx
prop	\propto

Логички симболи

Команда	Приказ
and	and
or	or
not	\neg
=>	\Rightarrow
if	if
iff	\Leftrightarrow
AA	\forall
EE	\exists
\perp	\perp
\top	\top
\vdash	\vdash
\models	\models

Заграде

Команда	Приказ
($($
)	$)$
[$[$
]	$]$
{	$\{$
}	$\}$
(:	\langle
:)	\rangle
{:	
:}	$:}$

Стрелице

Команда	Приказ
uarr	\uparrow
darr	\downarrow
rarr	\rightarrow
->	\rightarrow
->	\mapsto
larr	\leftarrow
harr	\leftrightarrow
rArr	\Rightarrow
lArr	\Leftarrow
hArr	\Leftrightarrow

Акценти

Команда	Приказ
hat x	\hat{x}
bar x	\bar{x}
ul x	\underline{x}
vec x	\vec{x}
dot x	\dot{x}
ddot x	\ddot{x}

Команде за фонтове

Команда	Приказ
bb "AaBbCc"	AaBbCc
bbb "AaBbCc"	AaBbCc
cc "AaBbCc"	<i>AaBbCc</i>
tt "AaBbCc"	AaBbCc
fr "AaBbCc"	<i>AaBbCc</i>
sf "AaBbCc"	AaBbCc

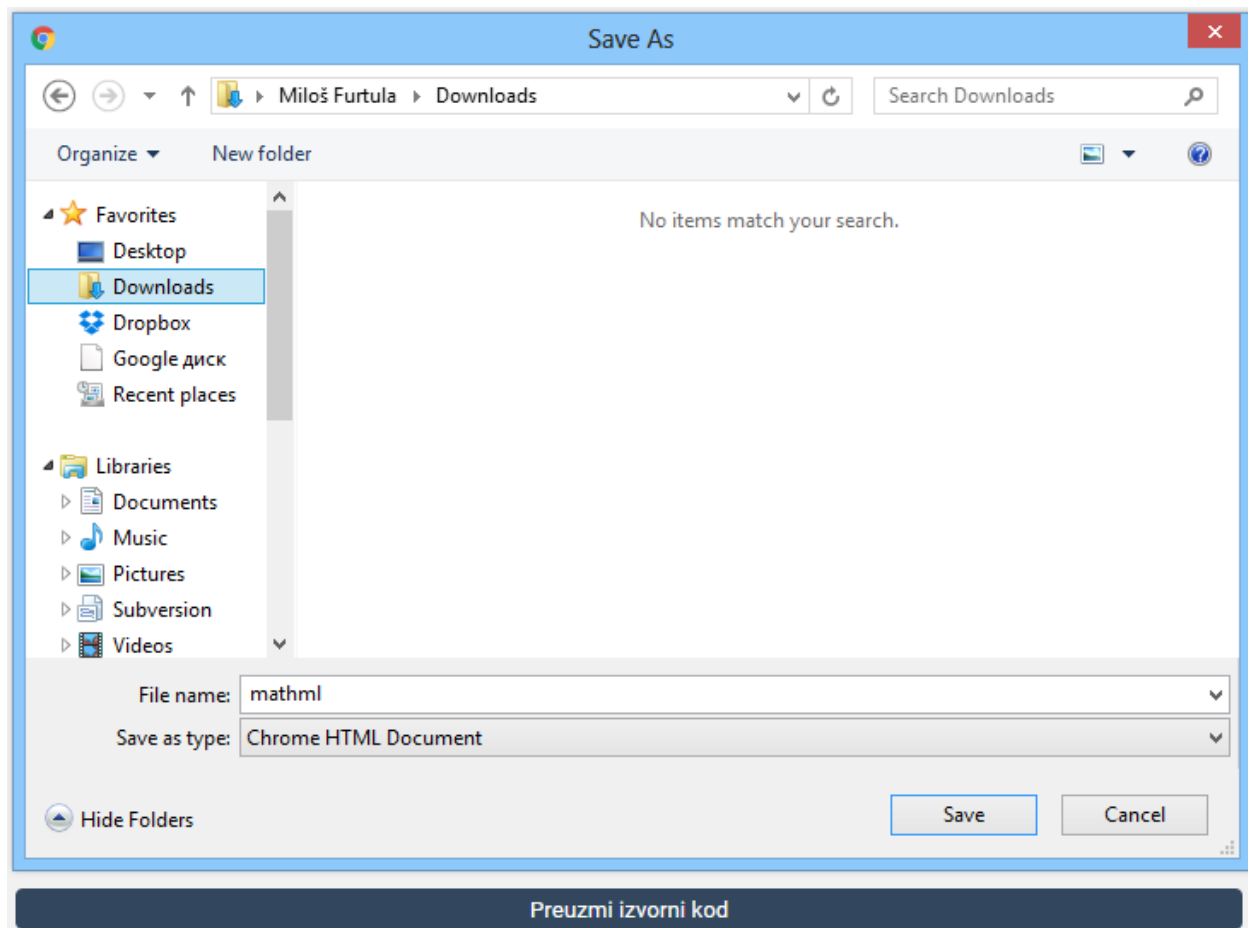
Специјални случајеви су:

- матрице – $[[a,b],[c,d]]$ дефинише $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$
- вектори колона – $((a,b),(c,d))$ дефинише $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$
- сложени индекси – $\lim_{x \rightarrow \infty}$ дефинише $\lim_{x \rightarrow \infty}$

Стандардне функције су $\sin, \cos, \tan, \csc, \sec, \cot, \sinh, \cosh, \tanh, \log, \ln, \det, \dim, \lim, \text{mod}, \gcd, \text{lcm}, \min, \max$.

Визуелни приказ израза за дефинисани код представља трећу компоненту апликације.

Додатна компонента, дугме за преузимање изворног Presentation MathML кода, није потребна за саму апликацију тј. формирање математичког израза и његов визуелни приказ али употпуњује апликацију јер омогућава укључивање формираног израза на неки други документ. Кликом на дугме преузима се документ (*mathml.html*), у ком се налази упутство за коришћење и MathML код формираног израза.



Слика 26 Преузимање изворног кода

За следећи математички израз (Слика 27) добија се документ са оваквим садржајем (Слика 28).

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 + 4ac}}{2a}$$

Слика 27 Математички израз

```

1 <!-- ide u head blok -->
2 <script src='http://cdn.mathjax.org/mathjax/latest/MathJax.js?config=MML_HTMLorMML'></script>
3 <!-- ----- -->
4 <!-- ide u deo gde zeli formula da se doda
5     ukoliko formula treba da bude u okviru teksta
6     dodati u math tagu atribut display="inline" inace je display="block" -->
7 <math xmlns="http://www.w3.org/1998/Math/MathML">
8   <mstyle displaystyle="true">
9     <msub>
10      <mi>x</mi>
11      <mrow>
12        <mn>1</mn>
13        <mo>,</mo>
14        <mn>2</mn>
15      </mrow>
16    </msub>
17    <mo>=</mo>
18    <mfrac>
19      <mrow>
20        <mo>-</mo>
21        <mi>b</mi>
22        <mo>&#xB1;</mo>
23      </mrow>
24      <msqrt>
25        <mrow>
26          <msup>
27            <mi>b</mi>
28            <mn>2</mn>
29          </msup>
30          <mo>+</mo>
31          <mn>4</mn>
32          <mi>a</mi>
33          <mi>c</mi>
34        </mrow>
35      </msqrt>
36    </mfrac>
37    <mo>=</mo>
38    <mi>a</mi>
39  </mstyle>
40 </math>

```

Слика 28 Преузети изворни код

Регистровани корисник који је пријављен на сајт, у оквиру апликације, има додатну могућност да сачува формиран израз.

SAČUVAJ IZRAZ

$n!$

Слика 29 Чување израза

6.2. Страница за чување израза

Притиском на дугме за чување израза отвара се нова страница где се могу дефинисати назив, опис и Content MathML код израза. Сва три поља су опциона и не морају се попунити.

$n!$

Naziv

Opis

Proizvod svih celih brojeva od n do 1.
#matematika

Content MathML

```
1 <math>
2   <apply>
3     <factorial/>
4     <ci>n</ci>
5   </apply>
6 </math>
```

SAČUVAJ IZRAZ

Слика 30 Страница за чување израза

Након што се унесу жељени подаци, притиском на дугме за чување израза, у базу формираних израза се дода нови израз. Након тога се отвара страница са различитим подацима о сачуваном изразу. Текстуални едитор за Content MathML код је функционалан HTML едитор за веб под именом [Ace](#).

6.3. Страница сачуваног израза

furtula

Faktorijel [0]

$$n!$$

Proizvod svih celih brojeva od n do 1. #matematika

AsciiMath

LaTeX

Presentation MathML

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <mstyle displaystyle="true">
    <mi>n</mi>
    <mo>!</mo>
  </mstyle>
</math>
```

Preuzmi izvorni kod

Content MathML

```
<math>
  <apply>
    <factorial/>
    <ci>n</ci>
  </apply>
</math>
```

Unesi vrednosti

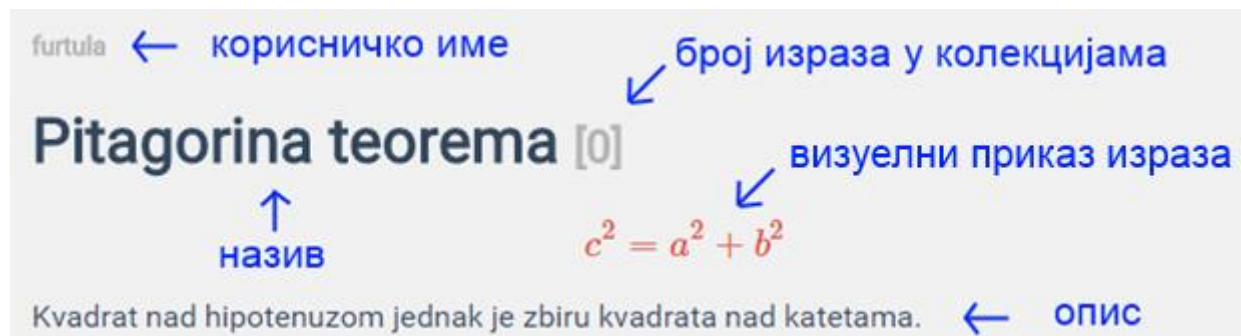
n

IZRAČUNAJ

120

Слика 31 Страница сачуваног израза

На овој страници се налазе разне информације о неком од сачуваних израза. У визуелне информације спада корисничко име корисника који је дефинисао израз, назив израза, број израза у колекцијама, визуелни приказ израза и опис.



Слика 32 Визуелне информације сачуваног израза

Након тога се могу видети изворни кодови за дефинисани математички израз у AsciiMath-y, LaTeX-y и Presentation MathML-y.

AsciiMath

```
sqrt(x)
```

LaTeX

```
\sqrt{{{x}}}
```

Presentation MathML

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <mstyle displaystyle="true">
    <msqrt>
      <mrow>
        <mi>x</mi>
      </mrow>
    </msqrt>
  </mstyle>
</math>
```

Слика 33 Изворни кодови

Такође, постоји и дугме за преузимање изворног кода, које ради исто што и дугме за преузимање изворног кода у оквиру апликације на почетној страници. Бојење кода је регулисано помоћу библиотеке [highlight.js](#).

Уколико постоји дефинисан Content MathML за математички израз, изворни код се може видети.

Content MathML

```
<math>
  <apply>
    <root/>
    <ci>x</ci>
  </apply>
</math>
```

Слика 34 Content MathML изворни код

Исто тако, само уколико постоји дефинисан Content MathML за математички израз, може се израчунати вредност изрази на основу дефинисаног Content MathML-а. Тако се за израз \sqrt{x} може израчунати вредност:

Unesi vrednosti	Unesi vrednosti
x <input type="text" value="761"/>	x <input type="text" value="-10"/>
IZRAČUNAJ	IZRAČUNAJ
27.586228448267	NAN

Слика 35 Израчунавање изрази

Елементе које прихвата програм за израчунавање су `math`, `apply`, `ci`, `cn`, `plus`, `minus`, `times`, `divide`, `power`, `root`, `factorial`, `eq`, `neq`, `gt`, `lt`, `geq`, `leq`, `abs`. Уколико је Content MathML код дефинисан са неким елементом ван овог скупа, програм за израчунавање неће вратити никакав резултат.

Поред ових информација које се могу видети на овој страници, за корисника који је овај израз сачувао, постоји опција за брисање изрази и линк за измену где могу да се измене подаци о изразу. Кад са кликне на дугме за брисање појави се прозор са поруком за потврду.

Da li ste sigurni da želite da obrišete ovaj izraz?

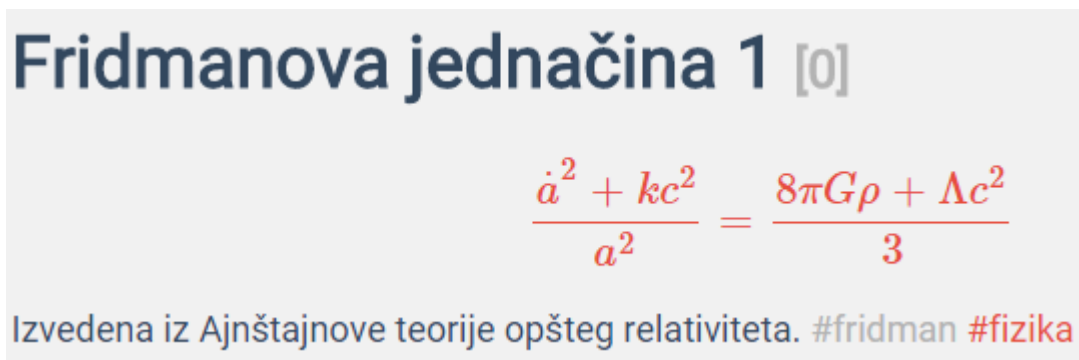
☐ Prevent this page from creating additional dialogs.

OK **Cancel**

OBRIŠI **IZMENI**

Слика 36 Брисање и измена изрази

У опису израза се може додати хештег (*hashtag*), а то је *тарабица*(#) праћена не *бланко* (празно поље) карактерима. Уколико опис израза садржи хештег, када се кликне на њега отвориће се страница за претрагу са тим хештегом као критеријумом. Хештег је користан за груписање израза из истих научних грана (математика, физика, хемија, ...) или из истих области (геометрија, логика, статика, ...).



Слика 37 Пример хештега

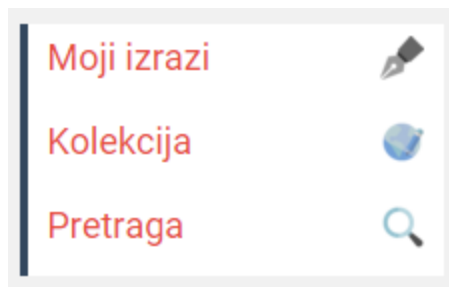
Уколико израз није корисников, уместо опције за брисање и измену израза биће понуђено да корисник дода у своју колекцију, или обрише из ње, израз који је неки други корисник дефинисао.



Слика 38 Додавање и брисање у и из колекције

6.4. Кориснички мени

Када је корисник пријављен на сајт, на свакој страници приказаће му се кориснички мени. Он служи да корисник може лако да приступи свим страницама сајта.



Слика 39 Кориснички мени

У кориснички мени спадају три линка, *Моји изрази*, *Колекција* и *Претрага* и сваки води на посебну страницу.

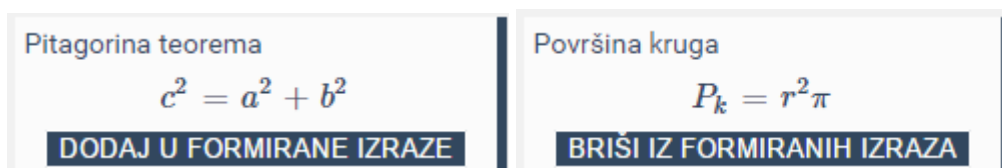
6.5. Моји изрази

На овој страници се налази скуп свих постојећих израза које је пријављени корисник дефинисао.



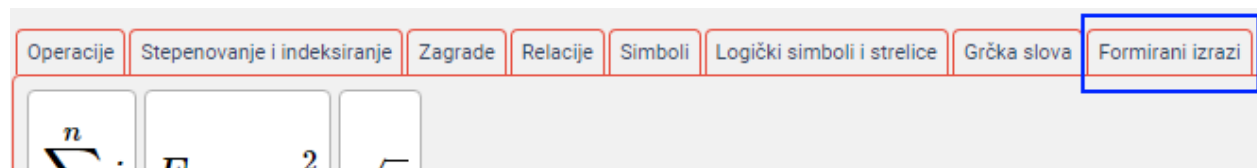
Слика 40 Моји изрази

Изрази су подељени у блокове са основним информацијама. Сваки блок води на страници сачуваног израза са више информација о изразу. Поред тога, у сваком блоку се налази дугме које динамички додаје и брише израз у и из скупа формираних израза.



Слика 41 Додавање и брисање из формираних израза

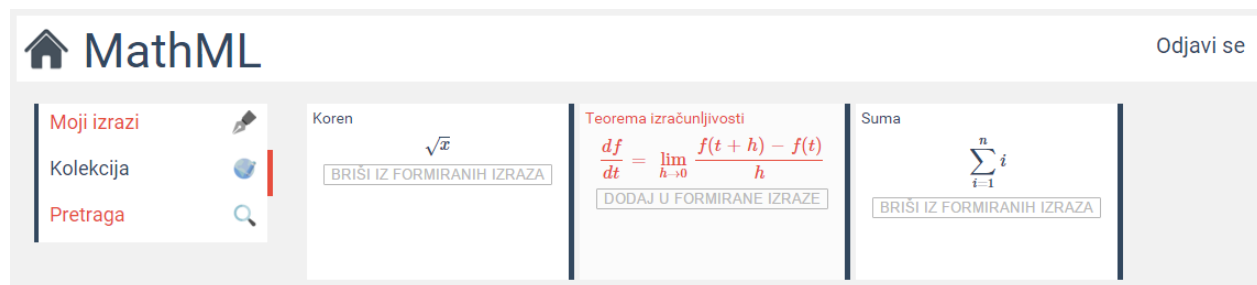
Формирани изрази су посебна категорија дугмића у апликацији која постоји само за регистроване кориснике и о томе је већ било речи.



Слика 42 Формирани изрази

6.6. Колекција

Када корисник жели да има брз приступ неком изразу, који је дефинисао неки други корисник, а да не мора стално да претражује, и притом може да дода тај израз у формиране изразе, онда је потребно да има своју колекцију израза које није сам дефинисао. Ова страница томе служи. На страници сачуваног израза, налази се дугме за додавање и брисање из колекције. Тако корисник може било који израз, сем свој, да сачува у своју колекцију израза.



Слика 43 Колекција

Страница *Колекција* има потпуно исти изглед као страница *Моји изрази*, и нуди исте функционалности. Једина разлика је у скупу израза.

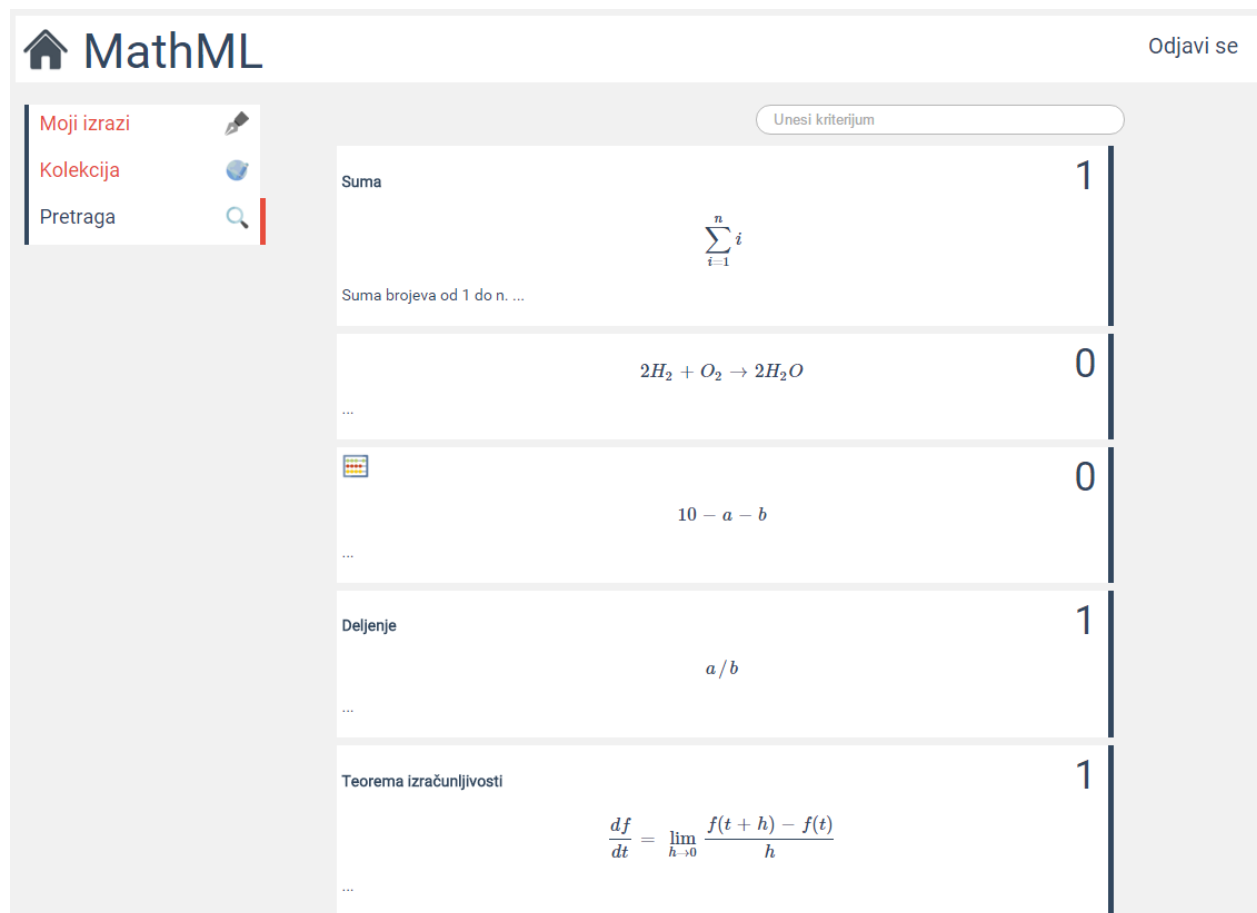
6.7. Претрага

Било да је пријављен корисник или не, може се приступити страници за претрагу. За пријављене кориснике линк који води ка овој страници се налази у корисничком менију, док се за не пријављене кориснике налази испод блока за пријаву.



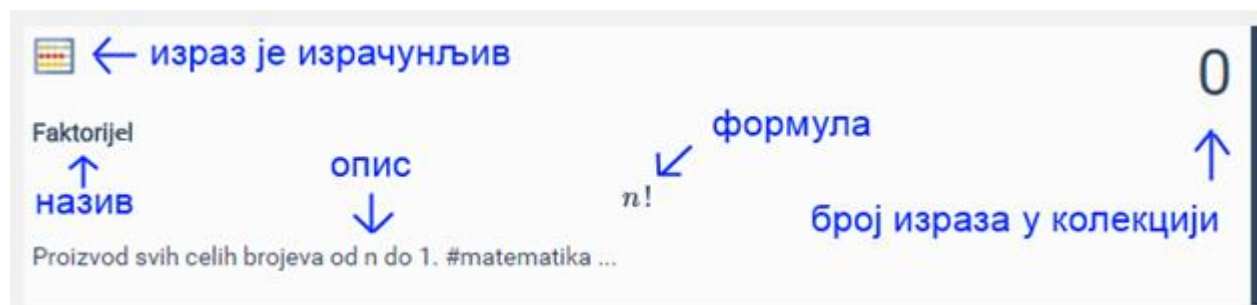
Слика 44 Линк за страницу претраге

Још један начин за приступ страници претраге је путем хештега који се може наћи у опису неког сачуваног израза. Након клика на хештег, отвара се страница претраге са критеријумом из хештега.



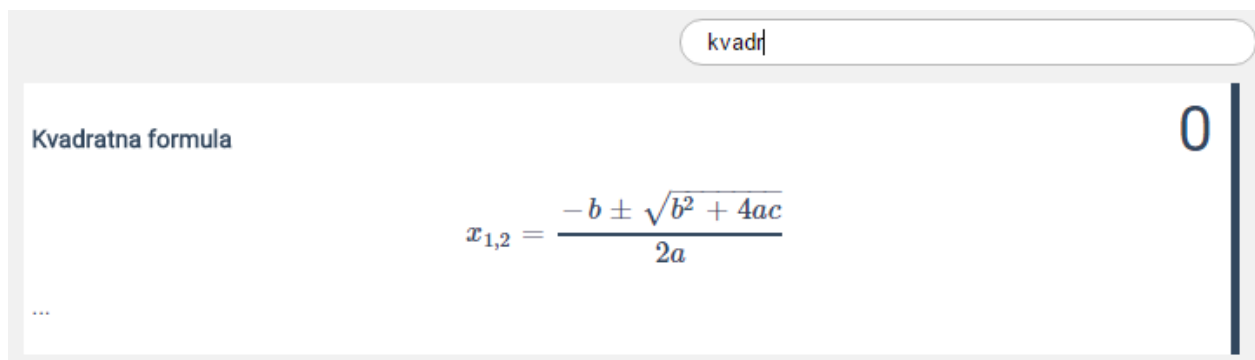
Слика 45 Страница претраге

На овој страници се налази скуп свих израза који задовољавају одређени критеријум. Изрази су организовани у блокове са основним информацијама. Кликом на блок отвара се страница сачуваног израза са више информација о изразу.



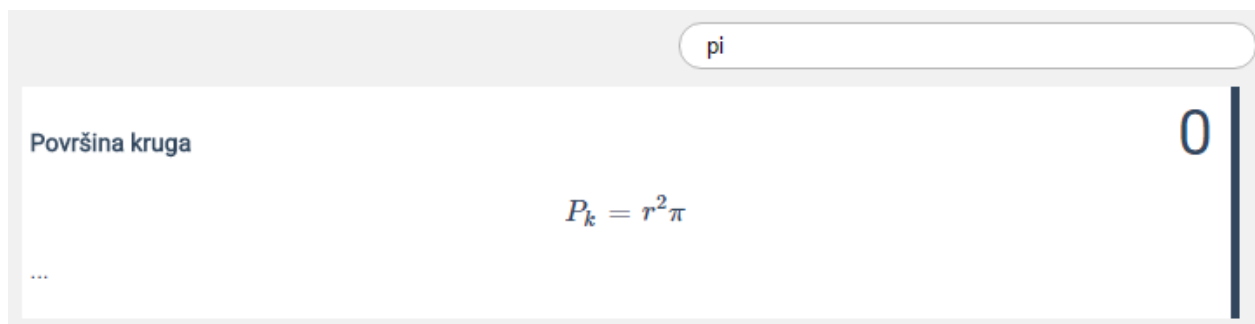
Слика 46 Блок израза

Критеријум за претрагу се уноси у текстуално поље и динамички се листа свих израза који задовољавају критеријум мења. Израз задовољава критеријум уколико се он налази у називу, опису, изразу или Content MathML коду.



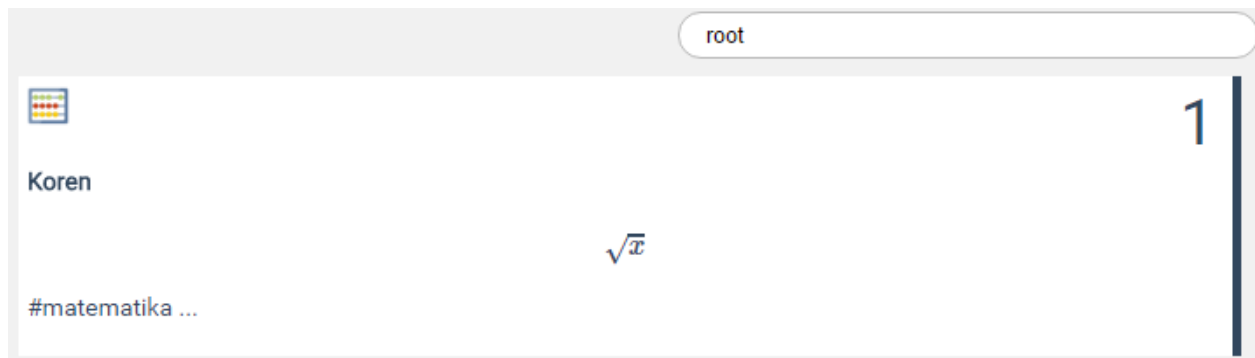
Слика 47 Резултат претраге за критеријум "kvadr"

Критеријум је пронађен у називу.



Слика 48 Резултат претраге за критеријум "pi"

Критеријум је пронађен у изразу.



Слика 49 Резултат претраге за критеријум "root"

Критеријум је пронађен у Content MathML коду.

Овим су описане све функционалности сајта.

7. Софтверско решење пројектног задатка

Сајт има многобројне функционалности које су уједно и на серверској и на клијентској страни. За лакше разумевање сајта у целини, биће описана архитектура софтверског решења. С обзиром да је код за функционисање пројекта велики, биће издвојени најважнији делови софтверског решења пројектног задатка.

7.1. Архитектура сајта

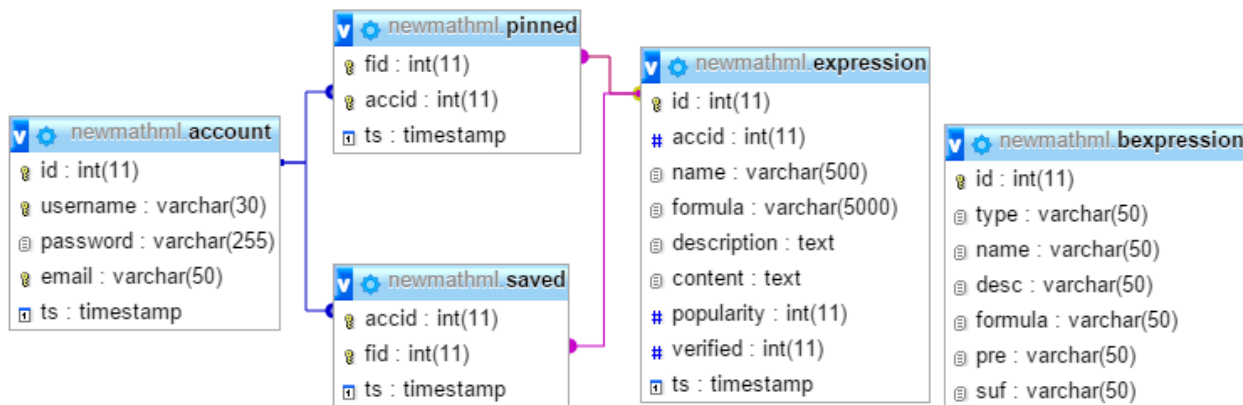
Функционалности сајта као што су регистравање корисника, дефинисање и претрага израза, додавање израза у колекцију и слично захтевају серверску страну сајта где ће се наћи база за чување оваквих података.



Слика 50 Архитектура сајта

7.2. База података

У бази података налази се пет табела и то *account*, *bexpression*, *expression*, *pinned*, *saved*.



Слика 51 Табеле у бази података

У табели *bexpression* се чувају подаци о основним изразима. Служи за генерисање дугмића у апликацији. Табела је предефинисана и непроменљива:

- *id* – јединствени идентификатор
- *type* – категорија у коју спада
- *name* – назив
- *desc* – AsciiMath код израза чији ће се визуелни приказ наћи на дугмету
- *formula* – AsciiMath код
- *pre* – представља префикс за изразе који могу да обухвате друге изразе као што су на пример заграде
- *suf* – представља суфикс за изразе који могу да обухвате друге изразе

У табели *account* се чувају подаци о кориснику:

- *id* – јединствени идентификатор
- *username* – корисничко име
- *password* – шифра
- *email* – електронска пошта
- *ts* – време уноса или измене података

Табела *expression* служи да се чувају подаци о дефинисаним изразима:

- *id* – јединствени идентификатор израза
- *accid* – јединствени идентификатор корисника који је дефинисао израз
- *name* – назив израза
- *formula* – AsciiMath код израза

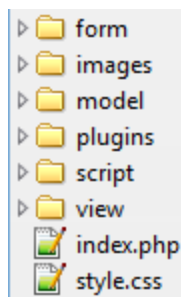
- **description** – опис израза
- **content** – Content MathML код за израчунавање израза
- **popularity** – број израза у колекцији
- **verified** – ваљаност израза
- **ts** – време уноса или измене података о изразу

Табеле **saved** и **pinned** имају исту структуру али различиту намену. Обе табеле представљају везу између корисника и дефинисаног израза. У првој табели се дефинише који корисник је који израз додао у своју колекцију. Док у другој се дефинише који корисник је који израз додао у скуп *формираних израза*. Структура табела садржи, **fid** – јединствени идентификатор израза из табеле **expression**, **accid** – јединствени идентификатор корисника из табеле **account**, и **ts** – време уноса података.

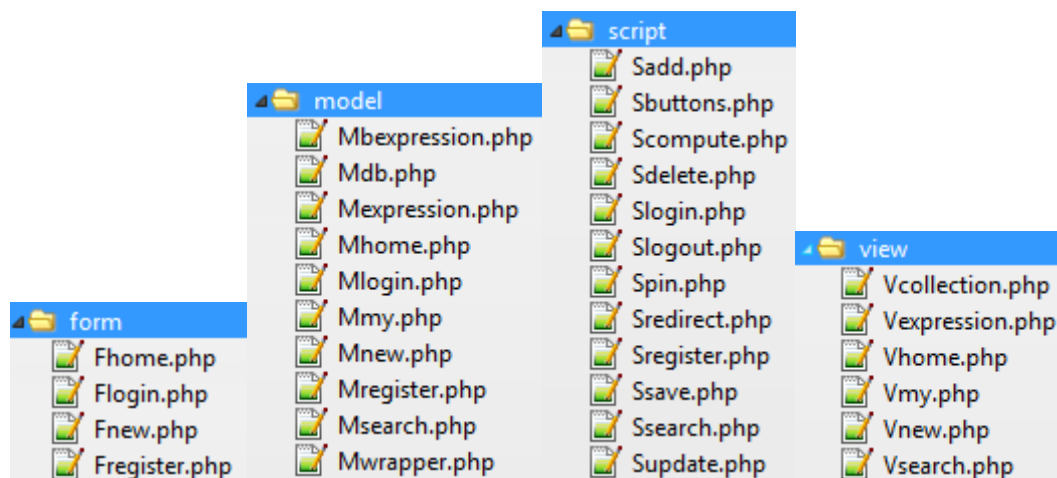
7.3. Организација кода

Због добре организације и лаке прегледности, код је организован тако да су странице, форме, реакције на интеракције корисника и веза са базом одвојени. Функционалности и значење кода су одвојени по директоријумима:

- **form** – садржи посебне компоненте које не зависе од других компоненти а могу се наћи на више страница, код је писан у HTML-у
- **images** – садржи слике и иконице које се користе на сајту
- **model** – у сваком фајлу је дефинисана класа која одговара или некој табели из базе података са функцијама уноса, измена или преузимања података, или неком одређеном скупу акција које су логички везане за класу, код је писан у PHP-у
- **plugins** – у овом директоријуму се налазе додатне библиотеке, код је писан у JavaScript-у
- **script** – скуп фајлова који представљају динамичко извршавање кода на интеракције корисника, код је писан у PHP-у
- **view** – сваки фајл представља једну страницу на сајту, код је писан у PHP-у



Слика 52 Организација кода



Слика 53 Именовање фајлова

Сваки фајл је именован тако да се лако добија информација коју сврху има и којем скупу фајлова припада. Име фајла почиње великим словима **F**, **M**, **S** и **V** и редом означавају **form**, **model**, **script** и **view** категорије фајлова. У наставку имена фајла, налази се једна реч која ближе одређује функцију фајла. Оваквом организацијом и именовањем се наглашава модуларност и лака читљивост кода, као и лако препознавање функционалности докумената. Приметна лака читљивост кода може се видети у наредним примерима.

Листинг 43: Flogin.php – Форма за пријављивање

```

1  <div id="login" class="side">
2      <h3>Prijava</h3>
3      <form method="POST" action="../script/Slogin.php">
4          <table style="border: 0px">
5              <tr>
6                  <td>Korisničko ime</td>
7                  <td>
8                      <input type="text" name="username"
9                      value="" placeholder="johnsmith123">
10                 </td>
11            </tr>
12            <tr>
13                <td>Šifra</td>
14                <td>
15                    <input type="password" name="password"
16                    value="" placeholder="*****">
17                </td>
18            </tr>
19            <tr>
20                <td colspan="2" align="right">
21                    <input type="submit" name="login"
22                    value="PRIJAVI SE">
23                </td>
24            </tr>
25        </table>
26    </form>
27 </div>

```

Листинг 44: Mdb.php – Приступ бази

```
1  <?php
2
3  class Mdb {
4      const HOST = "localhost";
5      const BAZA = "mathml";
6      const USER = "root";
7      const PASS = "";
8
9      public static function sqlRowQuery($query) {}
10     public static function sqlTableQuery($query) {}
11     public static function sqlScalarQuery($query) {}
12     public static function nonSelectQuery($query) {}
13 }
14
15 ?>
```

Листинг 45: Sdelete.php – Брисање израза

```
1  <?php
2      session_start();
3      include_once "../model/Mmy.php";
4      $id=$_GET['id'];
5      Mmy::delete($id);
6      header("Location: ../view/Vmy.php");
7  ?>
```

Листинг 46: Vcollection.php – Страница колекције израза

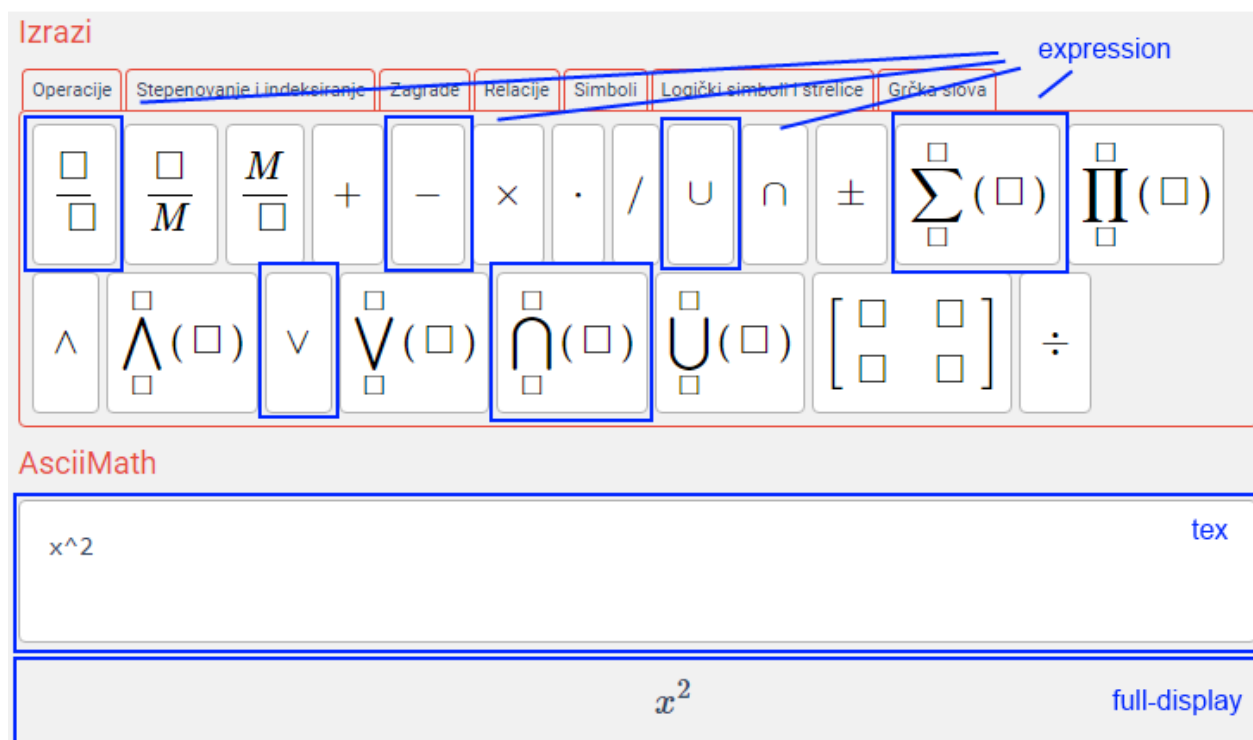
```
1  <?php
2
3  session_start();
4  include_once "../model/Mwrapper.php";
5  include_once "../model/Mlogin.php";
6  include_once "../model/Mexpression.php";
7  include_once "../model/Mregister.php";
8  if (!Wrapper::online()) header("Location: ../index.php");
9  Wrapper::begin(["mathmlf()","add()"]);
10     Wrapper::contentBegin();
11         if (Wrapper::online()) Wrapper::sideMenu();
12         Wrapper::centerBegin();
13             Mexpression::getAll();
14         Wrapper::centerEnd();
15         if (!Wrapper::online())
16         {
17             Mregister::form();
18             Mlogin::form();
19         }
20     Wrapper::contentEnd();
21 Wrapper::end();
22 ?>
```

7.4. Битна софтверска решења

Најважнији део сајта је апликација *Генератор израза*. Апликација се састоји из три дела. Први део је скуп основних израза. Други део је текстуално поље. Трећи део представља визуелни приказ израза. Аналогно томе постоје процеси који омогућавају формирање израза.

7.4.1. Формирање израза

Прво се врши претпроцесирање, тј. припрема интерфејса за коришћење. То представља преузимање података из базе о основним изразима, и генерисање дугмића по категоријама. Након припреме, корисник има могућност да притиска дугмиће и тако у текстуално поље генерише AsciiMath код. Где ће се код генерисати зависи од положаја курсора у пољу. Када је текст у текстуалном пољу селектован, и након тога се притисне неко од дугмића, и ако тај израз то омогућава, селектовани текст ће бити обухваћен.



Слика 54 Имена компоненти у апликацији

Након што се вредност текстуалног поља измени, динамички се тај AsciiMath код у пољу копира у `div` блок на страници, под именом `full-display`. Додају се `"`"` карактери јер тако MathJax проналази математику на страници дефинисану AsciiMath кодом. Тада се поново позове функција библиотеке MathJax за тај блок и динамички се измени визуелни изглед израза који одговара новом AsciiMath коду. Дугмићи су под именом `expression` а текстуално поље под `tex`.

Листинг 47

```

1  function updateMathML() {
2      $("#full-display").html("`"+$ascii.val()+"`");
3      MathJax.Hub.Queue(["Typeset",MathJax.Hub,"full-display"]);
4  }

1  $(".expression").click(function() {
2      $selTex = selectedText();
3      if ($selTex!="") $ascii.insertAtCaret($(this).attr("data-pre") +
4      $selTex + $(this).attr("data-suf"));
5      else $ascii.insertAtCaret($(this).attr("data-latex"));
6      updateMathML();
7  });

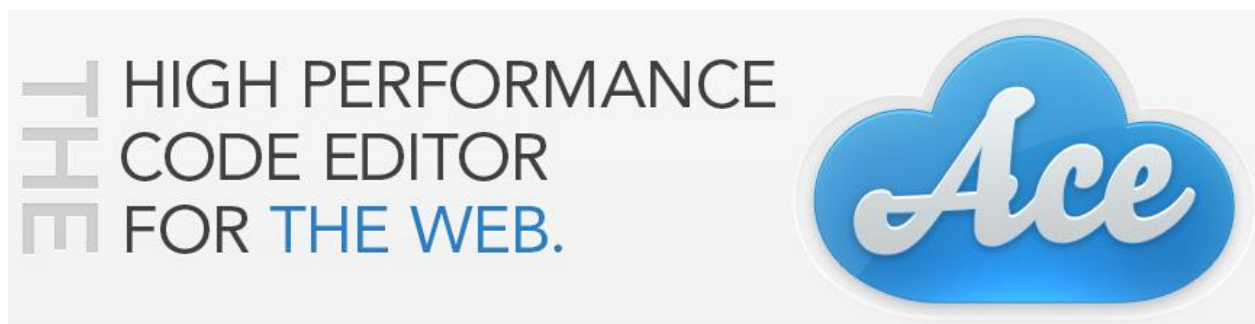
1  $("#tex").keyup(function () {
2      updateMathML();
3  });

```

Сваки пут када се дугмићи притисну провери се да ли је текст селектован и уколико јесте обухвати се префиксом и суфиксом тог израза. Уколико није, на тој позицији се дода AsciiMath код израза. Било да се дугме израза притисне или измени вредност текстуалног поља, визуелни приказ израза се динамички мења.

7.4.2. Дефинисање Content MathML кода

Када корисник одабере да сачува формирани израз, пружа му се могућност да дефинише Content MathML код. Content MathML је проширени XML. Идеалан едитор који омогућава једноставно кодирање је [Ace](#).



Слика 55 Ace

Ace је код едитор, писан у JavaScript-у, који се може укључити на веб страницу. Подржава наглашавање кода бојењем за преко сто десет програмских језика. Пружа аутоматско увлачење кода, и може да подржи велике документе (до четири милиона линија кода). Претрага у документу и измена са регуларним изразима као и *Cut*, *Copy* и *Paste* опције су омогућене. Наглашава отворену и затворену заграду која једна другој одговарају. Текст се може превлачити у оквиру едитора. Динамички проверава синтаксу кода.

Да би се укључио у страницу, прво је потребно преузети библиотеку (<http://github.com/ajaxorg/ace>). Након тога се укључи у страницу и конфигурише.

Листинг 48

```

1 <script src="../../plugins/ace/ace.js" type="text/javascript" charset="utf-
  8"></script>
2 <script>
3     var editor = ace.edit("editor");
4     editor.getSession().setMode("ace/mode/xml");
5 </script>

```

Конфигурацијом је дефинисано да ће се у њему писати XML код.

Content MathML

```

1 <math>
2   <apply>
3     <factorial/>
4     <ci>n</ci>
5   </apply>
6 </math>

```

Слика 56 Асе укључен у сајт

7.4.3. Наглашавање изворних кодова бојењем

Користећи библиотеку [highlight.js](https://highlightjs.org/download/) омогућен је читљив приказ изворног кода на веб страницама. Ова библиотека служи за наглашавање изворних кодова бојењем. Подржава сто тридесет седам програмских језика и шездесет пет различитих стилова. Има аутоматско препознавање програмског језика али може се и експлицитно навести. Такође има могућност вишејезичног обележавања, уколико код садржи мешавину више језика (на пример PHP + HTML + CSS + JavaScript). Предефинисана верзија библиотеке се може укључити али пружа подршку за само двадесет два програмска језика. Други начин је преузимање библиотеке (<https://highlightjs.org/download/>) са одабраним језицима који одговарају захтевима сајта. AsciiMath, као програмски језик, није подржан али постоје стилови за бојење неких уобичајених команди. Постоји подршка за TeX (LaTeX) и XML (MathML).

AsciiMath

$$x_{(1,2)} = \frac{-b \pm \sqrt{b^2 + 4ac}}{2a}$$

LaTeX

$$\{x\}_{\{1,2\}} = \frac{-b \pm \sqrt{b^2 + 4ac}}{2a}$$

Слика 57 Наглашавање AsciiMath и LaTeX кода помоћу highlight.js библиотеке

Библиотека се укључује на страницу и конфигурише тако што се прво одабере стил бојења, затим се укључи библиотека и на крају се позове функција за иницијализацију.

Листинг 49

```
1 <link rel='stylesheet' href='../plugins/highlight/styles/github.css'>
2 <script src='../plugins/highlight/highlight.pack.js'></script>
3 <script>hljs.initHighlightingOnLoad();</script>
```

Чим се страница учита, тражи се комбинација тагова која изгледа овако:

Листинг 50

```
1 <pre><code id='mathml' disabled='true' class='exp-area html'></code></pre>
```

Користећи тагове `pre` и `code` означава се поље које садржи код који треба обојити.

Presentation MathML

```
<math xmlns="http://www.w3.org/1998/Math/MathML">
  <mstyle displaystyle="true">
    <mi>n</mi>
    <mo>!</mo>
  </mstyle>
</math>
```

Слика 58 Наглашавање Presentation MathML кода помоћу highlight.js библиотеке

Пошто се бојење врши како се страница учита, а генерисање LaTeX и Presentation MathML кода се врши након тога, потребно је позвати функцију ове библиотеке како би поново претражила и обојила одговарајући код.

Листинг 51

```
1 $('pre code').each(function(i, block) {
2     hljs.highlightBlock(block);
3 });
```

7.4.4. Генерисање изворних кодова

Дефиниција израза се чува у бази података помоћу AsciiMath кода, како је то најефикаснији и најједноставнији начин. Чување Presentation MathML кода би било неефикасно и непотребно. Зато је неопходно некако добити Presentation MathML код. То омогућава MathJax библиотека. Након што се на страници сачуваног израза појави визуелни приказ израза, а то ради MathJax библиотека, може се искористити погодност MathJax екстензије `toMathML.js`. Помоћу ове екстензије из неког израза који је генерисао MathJax може се добити Presentation MathML код било ког израза.

Листинг 52

```
1 function toMathML(jax,callback) {
2     var mml;
3     mml = jax.root.toMathML("");
4     MathJax.Callback(callback)(mml);
5 }
```


Ова функција, за прослеђени MathJax елемент, враћа Presentation MathML код тог елемента кроз променљиву `mm1`.

Да би се добио LaTeX код користи се библиотека `ascii2tex.js`. Она преводи AsciiMath код у LaTeX.

Листинг 53

```
1 AMTparseAMtoTeX($('.formula').text());
```

7.4.5. Израчунавање израза на основу Content MathML кода

Content MathML је проширени XML. Да би се овај код искористио потребно је превести XML структуру тако да се на неки начин добије стабло израза. Стабло израза се састоји из чворова где се сваки чвор, у програмском језику, може представити као објекат. Како је већина чворова сложено, тј. садржи више подчворова, најлакше је те објекте представити као низове. На тај начин стабло израза представља низ низова. Користећи функције PHP-а, превођење се може лако извршити.

Листинг 54

```
1 $xml = simplexml_load_string($contentMathML);
2 $json = json_encode($xml);
3 $array = json_decode($json, TRUE);
```

Структура Content MathML-а формира стабло израза. Када се Content MathML преведе, формира се низ низова. За израз $\frac{a+b}{a-b}$, стабло израза у PHP-у изгледа овако:

Листинг 55

```
1 Array
2 (
3     [divide] => Array()
4     [apply] => Array
5     (
6         [0] => Array
7         (
8             [plus] => Array()
9             [ci] => Array
10            (
11                [0] => a
12                [1] => b
13            )
14        )
15        [1] => Array
16        (
17            [minus] => Array()
18            [ci] => Array
19            (
20                [0] => a
21                [1] => b
22            )
23        )
24    )
25 )
```

Да би се вршило израчунавање, прво је потребно издвојити променљиве из стабла.

Променљиве се чувају у **ci** елементима стабла израза. Код за формирање низа променљивих се налази у додатку(Код 56).

Да би се приступило свим елементима стабла потребно је рекурзивно проћи кроз све чворове стабла док се не дође до чворова који садрже бројевне или логичке вредности. На тај начин се могу издвојити случајеви типова чворова:

- **cn** – чвор садржи бројевну или логичку вредност
- **ci** – овакви чворови садрже назив променљиве, и у том случају се из низа променљивих узима вредност која одговара називу променљиве
- **op** – недефинисани чвор који представља скуп свих елемената који нису осталих типова, а ту спадају углавном елементи који су оператори
 - **plus** – оператор збира
 - **minus** – оператор одузимања
 - **times** – оператор множења
 - **divide** – оператор дељења
 - **power** – степеновање
 - **root** – корен
 - **factorial** – факторијел
 - **eq** – оператор једнакости
 - **neq** – оператор неједнакости
 - **gt** – веће од
 - **lt** – мање од
 - **geq** – веће или једнако од
 - **leq** – мање или једнако од
 - **abs** – апсолутно
- **apply** – представља чвор који садржи подстабло израза чији су чворови комбинације сва четири типа чворова

Код за израчунавање израза на основу Content MathML-а се налази у додатку(Код 57).

8. Закључак

MathML је увек изазивао расправу. Неслагања су се сводила на то да ли је MathML најбољи начин за дефинисање математичких израза. Разне компаније и самостални програмери су развијали софтвере који би донели математику на веб. Најпопуларнији међу њима је (*Wolfram*) *Mathematica*. Међутим, синтакса за дефинисање математичких израза, коју користи Mathematica, није проширива и наилази на велики број проблема. Такође, интернет заједница се не може ослонити само на допринос једне или више приватних компанија. Потребан је јединствен стандард за дефинисање математичких израза тако да у било ком софтверу дефинисани математички израз има недвосмислено значење. MathML својом XML структуром баш то и пружа, јединственост и проширивост а уз то је и отворени изворни код.

W3C, који су дефинисали и осмислили MathML, се ослањају на труд индивидуалних компанија и самосталних програмера да уврсте и искористе MathML стандард у својим програмима. До сада је направљен велики број апликација које могу да увезу и извезу MathML као и да рачунају многе изразе на основу MathML-а.

MathML је замишљен као стандард који представља математику на вебу. Само време може да покаже да ли је овај језик право решење за то, или не.

Додатак

Листинг 56: Формирање низа променљивих

```

1  public static function getCi($content){
2      $xml = simplexml_load_string($content);
3      $json = json_encode($xml);
4      $array = json_decode($json,TRUE);
5      function search($array, $key){
6          $results = array();
7          if (is_array($array)) {
8              if (isset($array[$key])) {
9                  $results[] = $array[$key];
10             }
11             foreach ($array as $subarray) {
12                 $results = array_merge($results, search($subarray,
13 $key));
14             }
15             return $results;
16         }
17         $srch = search($array,"ci");
18         function array_flatten($array) {
19             if (!is_array($array)) {
20                 return FALSE;
21             }
22             $result = array();
23             foreach ($array as $key => $value) {
24                 if (is_array($value)) {
25                     $result = array_merge($result,
26 array_flatten($value));
27                 } else {
28                     $result[$key] = $value;
29                 }
30             }
31             return $result;
32         }
33         $flat = array_flatten($srch);
34         $final = array_unique($flat);
35         asort($final);
36         echo "<table>";
37         foreach ($final as $var){
38             echo "<tr>";
39             echo "<td>`".$var."`</td>";
40             echo "<td>";
41             echo "<input class='ci' style='text-align:right;' type='text'";
42             echo "name='".$var."' />";
43             echo "</td>";
44             echo "</tr>";
45         }
46         echo "</table>";
47     }

```

Листинг 57: Израчунавање израза

```
1 include_once '../model/Mexpression.php';
2 $id = $_POST['id'];
3 $jNiz = json_decode($_POST['niz']);
4 $jNiz = json_decode(json_encode($jNiz), true);
5 $xml = simplexml_load_string(Mexpression::getContent($id));
6 $json = json_encode($xml);
7 $array = json_decode($json, TRUE);
8 if (key($array)=="apply") $array = $array["apply"];
9 function pravi($niz,$op,$jNiz)
10 {
11     switch ($op)
12     {
13         case "apply":
14             $res = pravi($niz,key($niz),$jNiz);
15             return $res;
16             break;
17         case "ci":
18             return $jNiz[$niz];
19             break;
20         case "cn":
21             return $niz;
22             break;
23         default:
24             switch ($op)
25             {
26                 case "plus": $res=0;
27                     while (next($niz))
28                     {
29                         $nn=current($niz);
30                         if (key($nn)=="0")
31                         {
32                             for ($i=0;$i<count($nn);$i++)
33                             {
34                                 $res += pravi($nn[$i],key($niz),$jNiz);
35                             }
36                         } else $res+=pravi(current($niz),key($niz),$jNiz);
37                     }
38                     return $res;
39                     break;
40                 case "times": $res=1;
41                     while (next($niz))
42                     {
43                         $nn=current($niz);
44                         if (key($nn)=="0")
45                         {
46                             for ($i=0;$i<count($nn);$i++)
47                             {
48                                 $res*=pravi($nn[$i],key($niz),$jNiz);
49                             }
50                         } else $res*=pravi(current($niz),key($niz),$jNiz);
51                     }
52                     return $res;
53                     break;
```

```

54     case "power": next($niz);
55     $nn=current($niz);
56     if (key($nn)=="0")
57     {
58         $f=pravi($nn[0],key($niz),$jNiz);
59         $s=pravi($nn[1],key($niz),$jNiz);
60     } else {
61         $f=pravi(current($niz),key($niz),$jNiz);
62         next($niz);
63         $s=pravi(current($niz),key($niz),$jNiz);
64     }
65     $res = pow($f,$s);
66     return $res;
67     break;
68     case "root": next($niz);
69     $t=pravi(current($niz),key($niz),$jNiz);
70     $res = sqrt($t);
71     return $res;
72     break;
73     case "minus": next($niz);
74     $nn=current($niz);
75     if (key($nn)=="0")
76     {
77         $res=pravi($nn[0],key($niz),$jNiz);
78         for ($i=1;$i<count($nn);$i++)
79         {
80             $res-=pravi($nn[$i],key($niz),$jNiz);
81         }
82     }
83     else $res=pravi(current($niz),key($niz),$jNiz);
84     while (next($niz))
85     {
86         $nn=current($niz);
87         if (key($nn)=="0")
88         {
89             for ($i=0;$i<count($nn);$i++)
90             {
91                 $res-=pravi($nn[$i],key($niz),$jNiz);
92             }
93         } else $res-=pravi(current($niz),key($niz),$jNiz);
94     }
95     return $res;
96     break;
97     case "divide": next($niz);
98     $nn=current($niz);
99     if (key($nn)=="0")
100    {
101        $f=pravi($nn[0],key($niz),$jNiz);
102        $s=pravi($nn[1],key($niz),$jNiz);
103    } else {
104        $f=pravi(current($niz),key($niz),$jNiz);
105        next($niz);
106        $s=pravi(current($niz),key($niz),$jNiz);
107    }

```

```

108     $res = $f/$s;
109     return $res;
110     break;
111 case "factorial": next($niz);
112     $f=pravi(current($niz),key($niz),$jNiz);
113     $res=1;
114     for ($i=2;$i<=$f;$i++) $res=$res*$i;
115     return $res;
116     break;
117 case "eq": next($niz);
118     $nn=current($niz);
119     if (key($nn)=="0")
120     {
121         $f=pravi($nn[0],key($niz),$jNiz);
122         $s=pravi($nn[1],key($niz),$jNiz);
123     } else {
124         $f=pravi(current($niz),key($niz),$jNiz);
125         next($niz);
126         $s=pravi(current($niz),key($niz),$jNiz);
127     }
128     if ($f==$s) return "TRUE"; else return "FALSE";
129     break;
130 case "neq": next($niz);
131     $nn=current($niz);
132     if (key($nn)=="0")
133     {
134         $f=pravi($nn[0],key($niz),$jNiz);
135         $s=pravi($nn[1],key($niz),$jNiz);
136     } else {
137         $f=pravi(current($niz),key($niz),$jNiz);
138         next($niz);
139         $s=pravi(current($niz),key($niz),$jNiz);
140     }
141     if ($f!=$s) return "TRUE"; else return "FALSE";
142     break;
143 case "gt": next($niz);
144     $nn=current($niz);
145     if (key($nn)=="0")
146     {
147         $f=pravi($nn[0],key($niz),$jNiz);
148         $s=pravi($nn[1],key($niz),$jNiz);
149     } else {
150         $f=pravi(current($niz),key($niz),$jNiz);
151         next($niz);
152         $s=pravi(current($niz),key($niz),$jNiz);
153     }
154     if ($f>$s) return "TRUE"; else return "FALSE";
155     break;
156 case "lt": next($niz);
157     $nn=current($niz);
158     if (key($nn)=="0")
159     {
160         $f=pravi($nn[0],key($niz),$jNiz);
161         $s=pravi($nn[1],key($niz),$jNiz);

```

```
162     } else {
163         $f=pravi(current($niz),key($niz),$jNiz);
164         next($niz);
165         $s=pravi(current($niz),key($niz),$jNiz);
166     }
167     if ($f<$s) return "TRUE"; else return "FALSE";
168     break;
169 case "geq": next($niz);
170     $nn=current($niz);
171     if (key($nn)=="0")
172     {
173         $f=pravi($nn[0],key($niz),$jNiz);
174         $s=pravi($nn[1],key($niz),$jNiz);
175     } else {
176         $f=pravi(current($niz),key($niz),$jNiz);
177         next($niz);
178         $s=pravi(current($niz),key($niz),$jNiz);
179     }
180     if ($f>=$s) return "TRUE"; else return "FALSE";
181     break;
182 case "leq": next($niz);
183     $nn=current($niz);
184     if (key($nn)=="0")
185     {
186         $f=pravi($nn[0],key($niz),$jNiz);
187         $s=pravi($nn[1],key($niz),$jNiz);
188     } else {
189         $f=pravi(current($niz),key($niz),$jNiz);
190         next($niz);
191         $s=pravi(current($niz),key($niz),$jNiz);
192     }
193     if ($f<=$s) return "TRUE"; else return "FALSE";
194     break;
195 case "abs": next($niz);
196     $f=pravi(current($niz),key($niz),$jNiz);
197     return abs($f);
198     break;
199 }
200 break;
201 }
202 }
203
204 $v=pravi($array,key($array),$jNiz);
205 echo $v;
```


Листа скраћеница

- **MathML** – Mathematical Markup Language
- **MML** – Mathematical Markup Language
- **XML** – Extensible Markup Language
- **W3C** – World Wide Web Consortium
- **HTML** – HyperText Markup Language
- **HTML5** – HyperText Markup Language Five
- **CSS** – Cascading Style Sheets
- **XHTML** – Extensible HyperText Markup Language
- **IE6, 9, 10, 11** – Internet Explorer 6, 9, 10, 11
- **SVG** – Scalable Vector Graphics
- **OMDoc** – Open Mathematical Documents
- **API** – Application Programming Interface
- **CDN** – Content Distribution Network
- **HTTPS** – HyperText Transfer Protocol Secure
- **DOM** – Document Object Model
- **CD** – Content Dictionary
- **CDs** – Content Dictionaries
- **URI** – Uniform Resource Identifier
- **JSON** – JavaScript Object Notation
- **PHP** – Personal Home Page
- **PDF** – Portable Document Format

Литература

- [1] Mathematical Markup Language (MathML) Version 3.0 2nd Edition, <https://www.w3.org/TR/MathML3/>, 2015.
- [2] MathML, <https://en.wikipedia.org/wiki/MathML>, 2015.
- [3] Gecko (software), [https://en.wikipedia.org/wiki/Gecko_\(software\)](https://en.wikipedia.org/wiki/Gecko_(software)), 2015.
- [4] WebKit, <https://en.wikipedia.org/wiki/WebKit>, 2015.
- [5] OpenMath, <https://en.wikipedia.org/wiki/OpenMath>, 2015.
- [6] MathJax, <http://docs.mathjax.org/en/latest/mathjax.html>, 2015.
- [7] Thomas Cool, The Disappointment and Embarrassment of MathML, <http://thomascool.eu/Papers/MathML/OnMathML.html>, 2015.

Универзитет у Крагујевцу
Природно-математички факултет
Институт за математику и информатику

Завршни рад под називом

MathML

одбрањен је _____.

МЕНТОР:

др Име Презиме, звање, Институција

ЧЛАНОВИ КОМИСИЈЕ:

др Име Презиме, звање, ПМФ Крагујевац

др Име Презиме, звање, Институција

Завршни рад је оцењен оценом _____.