

# Optimizacija matematičkog sloga u e-knjigama

---

**Turčić, Maja**

**Doctoral thesis / Doktorski rad**

**2017**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Graphic Arts / Sveučilište u Zagrebu, Grafički fakultet**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:216:443074>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-04-02**



*Repository / Repozitorij:*

[Faculty of Graphic Arts Repository](#)





Sveučilište u Zagrebu

Grafički fakultet

Maja Turčić

**OPTIMIZACIJA MATEMATIČKOGA  
SLOGA U E-KNJIGAMA**

DOKTORSKI RAD

Zagreb, 2017.



Sveučilište u Zagrebu

Grafički fakultet

Maja Turčić

**OPTIMIZACIJA MATEMATIČKOGA  
SLOGA U E-KNJIGAMA**

DOKTORSKI RAD

Mentor: prof. dr. sc. Klaudio Pap

Zagreb, 2017.



Sveučilište u Zagrebu  
Faculty of Graphic Arts

Maja Turčić

**MATHEMATICAL TYPESETTING  
OPTIMIZATION IN E-BOOKS**

DOCTORAL THESIS

Supervisor: prof. dr. sc. Klaudio Pap

Zagreb, 2017.

**UDK BROJ: 655.28.022.2:51:02:004**

**Povjerenstvo za ocjenu doktorskog rada:**

1. prof. dr. sc. Nikola Mrvac, Sveučilište u Zagrebu Grafički fakultet, predsjednik
2. doc. dr. sc. Nikolina Stanić Loknar, Sveučilište u Zagrebu Grafički fakultet, članica
3. izv. prof. dr. sc. Damir Vusić, Sveučilište Sjever, Koprivnica, vanjski član

**Povjerenstvo za obranu doktorskog rada:**

1. prof. dr. sc. Nikola Mrvac, Sveučilište u Zagrebu Grafički fakultet, predsjednik
2. izv. prof. dr. sc. Ivana Žiljak Stanimirović, Sveučilište u Zagrebu Grafički fakultet, članica
3. izv. prof. dr. sc. Damir Vusić, Sveučilište Sjever, Koprivnica, vanjski član

**Mentor:** prof. dr. sc. Klaudio Pap, Sveučilište u Zagrebu Grafički fakultet

**Datum obrane doktorskog rada:** 13. ožujak 2017.

**Mjesto obrane doktorskog rada:** Sveučilište u Zagrebu Grafički fakultet

**Povjerenstvo za obranu doktorskog rada donijelo je sljedeću odluku:**

„Obrabila s ocjenom summa cum laude (*s najvećom pohvalom*) jednoglasnom odlukom Povjerenstva“

## SAŽETAK

Ključne riječi: e-knjiga, matematički slog, pristupačnost, dinamički prijelom

Razvojem tehnologije e-knjiga i važnosti semantike prilikom izrade sadržaja postavlja se problem optimizacije matematičkog sloga i njegova prijeloma kako bi se omogućila čitljivost i pristupačnost takvog tipa sadržaja na različitim platformama e-čitača. Tijekom istraživanja pokazalo se da se autori često koriste slikama za prikaz matematičkih formula ili primjenjuju rješenja koja ne opisuju sadržaj semantički ispravno, što je neupotrebljivo za dinamički matematički slog. U cilju postizanja optimalnoga matematičkog dinamičkog sloga u e-knjigama analiziraju se postojeći formati e-knjiga te se istražuju načini kreiranja i ugrađivanja semantički ispravnog i pristupačnog matematičkog sadržaja u e-knjigu. U radu se predlaže proces izrade semantički ispravnoga matematičkog sadržaja u e-knjigama primjenom odgovarajućih skriptnih i označnih jezika, što poboljšava pristupačnost, tečnost i razvitak naprednih načina korištenja e-čitačima te olakšano arhiviranje i konverziju u druge formate. U radu se istražuju pojedini problemi različitih prikazanih tehnologija e-čitača te se nude rješenja za postizanje dinamičkog i tipografski oblikovanog, pristupačnog matematičkog sloga.

## **ABSTRACT**

Key words: e-book, mathematical layout, accessibility, dynamic layout

With the development of e-book technology and the importance of semantics while creating content, a problem of mathematical type optimization and its layout is presented in order to enable readability and accessibility of that kind of content on different platforms of e-readers. Research shows that authors often use images for displaying mathematical formulae or use semantically incorrect markup which is unusable for fluid mathematical layout. In order to achieve optimal dynamic mathematical layout, available e-book formats are analyzed, different kinds of creating and embedding of semantically correct and accessible mathematical content in e-books are researched. This research proposes a process of creating semantically correct mathematical content through the use of adequate script and markup languages, thus enhancing accessibility, fluidity and development of advanced ways of using e-readers, easier archiving and format conversions. This research also examines specific problems of different e-reader renderers and solutions are offered for achieving dynamic and typographically formatted, accessible mathematical layout.

## SADRŽAJ

<b>1. UVOD .....</b>	<b>1</b>
1.1. Pregled dosadašnjih istraživanja .....	3
1.2. Cilj i hipoteze.....	6
1.3. Očekivani znanstveni doprinos .....	7
1.4. Metodologija i plan istraživanja .....	8
<b>2. ANALIZA FORMATA E-KNJIGA .....</b>	<b>11</b>
2.1. Format Mobipocket .....	12
2.2. Format AZW (Amazon Kindle format).....	14
2.3. Format KF8 (Kindle Format 8).....	15
2.4. Format EPUB .....	17
2.5. Format EPUB 2 .....	18
2.6. Format EPUB 3 .....	19
2.7. Format iBooks (IBA) .....	22
2.8. Format PDF (Portable Document Format).....	24
2.9. Evaluacija pristupačnosti formata i uređaja.....	26
<b>3. ANALIZA PRISTUPAČNOSTI E-ČITAČA .....</b>	<b>28</b>
<b>4. KREIRANJE E-KNJIGA RAZLIČITIH FORMATA .....</b>	<b>30</b>
<b>5. USPOREDBA FORMATA E-KNJIGA PREMA DEFINIRANIM PARAMETRIMA OPTIMALNOGA MATEMATIČKOG SLOGA.....</b>	<b>32</b>
<b>6. KREIRANJE MODELA E-KNJIGE STANDARDOM EPUB 3.....</b>	<b>37</b>
6.1. Dokument <i>package</i> .....	41
6.2. Metapodaci .....	43
6.3. Elementi za popis izvora publikacije i slijed čitanja – <i>manifest</i> i <i>spine</i> .....	45
6.4. Navigacijski dokument .....	46
6.5. Validacija .....	48
6.6. Analiza softverskih alata za kreiranje EPUB e-knjiga.....	49
<b>7. STANDARDI UGRAĐIVANJA MATEMATIČKIH FORMULA U E-KNJIGE .....</b>	<b>52</b>
7.1. Ugrađivanje matematičkog sloga u obliku slika u e-knjige .....	54
7.2. Usporedba pristupačnosti matematičkog sloga u obliku slika u e-knjigama .....	56



<b>8.</b>	<b>UGRAĐIVANJE MATEMATIČKOG SLOGA U E-KNJIGE PUTEM JEZIKA MATHML....</b>	<b>59</b>
8.1.	Analiza softverskih alata za kreiranje matematičkih izraza u jeziku MathML.....	61
8.2.	Analiza pristupačnosti sadržaja oblikovanog jezikom MathML .....	64
8.3.	Analiza i komparacija primjera e-knjiga s ugrađenim matematičkim sadržajem.....	66
<b>9.</b>	<b>EKSPERIMENTALNA RJEŠENJA PRIKAZA MATEMATIČKOG SLOGA U EPUB E-KNJIZI NA STANDARDNIM E-ČITAČIMA .....</b>	<b>69</b>
9.1.	Eksperimentalno rješenje za e-čitač EPUBReader .....	70
9.2.	Eksperimentalno rješenje za e-čitač Apple iBooks.....	79
9.3.	Eksperimentalno rješenje za e-čitač Adobe Digital Editions .....	83
9.4.	Eksperimentalno rješenje za e-čitač Readium verzije 2.16.X.....	86
9.5.	Eksperimentalno rješenje za IDEAL Group Reader na platformi Android i e-čitač Calibre na platformi Windows i OS X .....	93
<b>10.</b>	<b>EKSPERIMENTALNI REZULTATI I ANALIZA .....</b>	<b>99</b>
10.1.	Analiza podrške MathML-a .....	100
10.2.	Analiza metode kreiranja semantički pravilnoga matematičkog sloga .....	101
10.3.	Dinamičko traženje točke prijeloma u matematičkom slogu.....	102
10.4.	Oblikovanje dinamičkoga matematičkog prijeloma.....	103
10.5.	Pristupačnost matematičkog sloga u e-knjigama .....	105
<b>11.</b>	<b>ZAKLJUČAK.....</b>	<b>108</b>
<b>12.</b>	<b>LITERATURA .....</b>	<b>112</b>
<b>13.</b>	<b>POPIS SLIKA I TABLICA.....</b>	<b>119</b>
<b>14.</b>	<b>PRILOZI .....</b>	<b>122</b>
14.1.	SADRŽAJ DOKUMENTA <i>PACKAGE.OPF</i> .....	123
14.2.	KOD MATEMATIČKE FORMULE U JEZIKU MATHML .....	125
<b>15.</b>	<b>ŽIVOTOPIS I POPIS OBJAVLJENIH RADOVA .....</b>	<b>131</b>

## 1. UVOD

Prijelom matematičkog sloga oduvijek je bio jedna od kompleksnijih zadaća grafičkog dizajnera zbog vlastitih tipografskih pravila i drugačijih slovnih znakova koji se podrazumijevaju. U digitalnom okruženju sustavi na koji se većina oslanja pri oblikovanju matematičkog sadržaja, posebice za znanstvene i stručne publikacije, omogućuju besprijekorno oblikovanje statične stranice. Međutim, prilikom oblikovanja tečnog i semantički korektnoga matematičkog sadržaja u e-knjigama dizajner se suočava s problemima odabira optimalnog formata e-knjige te načina oblikovanja i ugrađivanja matematičkog sadržaja. Problem je složeniji ako postoji potreba za prelamanjem matematičkih izraza u više redaka, što ovisi o uređaju i veličini zaslona te specifičnim tipografskim pravilima koji osiguravaju čitkost. Optimalan format e-knjige uključuje: otvorenost koja podrazumijeva slobodu korištenja i implementacije, pretraživost, prilagođavanje sadržaja svim veličinama i rezolucijama zaslona, pristupačnost sadržaja osobama s invaliditetom (posebice slabovidnim i slijepim osobama) te podržanost svih potrebnih slovnih znakova.

Nakon uvodnog pregleda dosadašnjih istraživanja, postavljanja cilja i hipoteza te plana i metoda istraživanja, u drugom se poglavlju analiziraju različiti formati e-knjiga koji su popularni na trenutačnom tržištu u okviru tehničkog razvoja. Analizira se struktura, pojam otvorenosti, pristupačnosti i korisnosti, vrste podržanog sadržaja, tečnosti, mogućnosti ili obveze ugrađivanja DRM (Digital Rights Management) sheme, e-čitača na kojima se format može čitati, alata za kreaciju te ugrađivanje i izrada matematičkog sadržaja na formatima Mobipocket (2.1), AZW (2.2), KF8 (2.3), EPUB (2.4), EPUB 2 (2.5), EPUB 3 (2.6), IBA (2.7) i PDF (2.8). Nakon toga se pristupa evaluaciji pristupačnosti formata i uređaja osobama s posebnim potrebama u polju vida te potrebi semantičke korektnosti označavanja sadržaja (2.9).

U trećem poglavlju analiziraju se e-čitači u vidu njihovih tehničkih mogućnosti prikazivanja sadržaja i mehanizama za iščitavanje sadržaja. U četvrtom se poglavlju istražuju različiti načini i softveri za kreiranje e-knjiga različitih formata.

Peto poglavlje prikazuje usporedbu formata e-knjiga prema zahtjevima postavljenima za kreiranje pristupačne i tečne e-knjige matematičkog sadržaja prema tečnosti,

podržanosti, vremenu i jednostavnosti izrade, prijelomu i ažuriranju matematičkog sadržaja, otvorenosti te ugrađivanju DRM sheme.

Šesto poglavlje opisuje proces izrade modela e-knjige prema standardu EPUB 3, prikazuje se struktura EPUB formata e-knjige te obvezni elementi. Slijedi opis ključnih dijelova formata EPUB započevši s XML dokumentom *package* (6.1) koji se sastoji od obveznih i opcionalnih metapodataka (6.2), obveznih elemenata za popis izvora publikacije i slijeda čitanja (6.3) te obveznoga navigacijskog dokumenta (6.4). Također se opisuje potreba i proces validacije EPUB e-knjige kako bi kreator znao da je e-knjiga oblikovana u skladu sa specifikacijom (6.5). Na kraju se analiziraju softverski alati za kreiranje EPUB e-knjiga, čija je usporedba tablično prikazana (6.6).

U sedmom poglavlju istražuju se načini ugrađivanja matematičkog sadržaja u EPUB e-knjigu uzimajući u obzir tipografske zahtjeve prezentacije sadržaja zbog razumljivosti i čitkosti, ali i pristupačnosti, pretraživost i prebacivanja u druge formate. Izvodi se analiza načina ugrađivanja matematičkog sadržaja u obliku slike (7.1) i usporedba pristupačnosti matematičkog sadržaja u obliku slika (7.2).

Osmo poglavlje opisuje način ugrađivanja matematičkog sadržaja putem jezika MathML, njegovu pristupačnost i podržanost u e-čitačima. Slijedi analiza alata za kreiranje matematičkih izraza u jeziku MathML (8.1) i analiza pristupačnosti jezika MathML i važnost semantičke korektnosti označavanja matematičkih izraza (8.3). Slijedi analiza dostupnih primjera EPUB e-knjiga matematičkog sadržaja (8.3), koja je tablično prikazana.

U devetom poglavlju opisuju se eksperimentalna rješenja prikaza matematičkog sadržaja EPUB e-knjiga na odabranim e-čitačima. Slijede opisi problema prikaza i vizualni prikazi stanja prije i nakon apliciranja JavaScript skripti za dinamički prijelom te njihova rješenja u e-čitačima EPUBReader (9.1), Apple iBooks (9.2), Adobe Digital Editions (9.3), Radium (9.4), IDEAL Group Reader i Calibre (9.5).

U desetom se poglavlju opisuju i analiziraju rezultati eksperimentalnog dijela putem podrške MathML-a (10.1), važnosti kreiranja semantički pravilnog sadržaja (10.2), dinamičkog prijeloma matematičkog sadržaja (10.3), tipografskog prijeloma (10.4) i pristupačnosti matematičkog sadržaja (10.5) te su tablično prikazani rezultati eksperimentalnog dijela.

## 1.1. Pregled dosadašnjih istraživanja

E-knjige u većini su slučajeva degradirane na jednostavne digitalne inačice tiskanih knjiga [1]. Danas se mogu proizvoditi e-knjige čiji sadržaj može biti multimedijskog tipa, visoko pristupačne osobama s invaliditetom, otvorenog tipa, sadržaja koji se može ažurirati i pretraživati, koje podržavaju sve znakove, pisma i jezike te se mogu čitati na različitim platformama (monitorima, tabličnim računalima, e-čitačima i mobilnim telefonima). Stvara se potreba i podloga za stvaranjem semantički ispravnog sadržaja koji adekvatno označava elemente sadržaja označnim jezikom, a time pomaže pri ažuriranju, pristupačnosti, konverziji u druge formate, pretraživanju i arhiviranju. Zbog tih su prednosti najveće tržište za e-knjige akademske i znanstvene publikacije koje imaju potrebu za primjenom najsloženijega matematičkog sloga. Matematički je slog najzahtjevniji vid prijeloma kako u tisku tako i u e-knjigama. Jedno je od rješenja upotreba slika, čime se gubi semantička ispravnost podataka, smanjuje se kvaliteta prikaza, komplicira ažuriranje podataka te onemogućuje konverzija u druge formate. Također se onemogućuje tečnost sadržaja, odnosno prilagodba sadržaja različitim veličinama, orijentaciji i razlučivosti zaslona. Drugi pristupi koriste se najnovijim označnim jezicima, ali zbog izbjegavanja rješavanja problema prikaza i prijeloma koriste se semantički neispravnim elementima pri opisu sadržaja. Tim se pristupima uvode novi problemi procesiranja podataka te onemogućuje korištenje pomoćnim tehnologijama, zanemarujući time općenitu primarnu prednost korištenja semantički ispravnim označnim jezikom.

Zbog novih tehnologija te nove namjene i konzumacije sadržaja, u usporedbi s prijelomom za tisak, evidentna je potreba za definiranjem novih procesa izrade e-knjiga, koja je osnovana na otvorenim mrežnim standardima, a uključuje razdvajanje sadržaja, stila i strukture [2]. Semantička ispravnost pri izradi e-knjiga, ali i razdvajanje sadržaja od stila, omogućuje pomoćnim tehnologijama da sadržaj učine pristupačnim i osobama s određenim vrstama invaliditeta (prvenstveno slijepima), ali i čitateljima u neoptimalnim situacijama (npr. izravna Sunčeva svjetlost) [3]. Semantički ispravan opis sadržaja također igra važnu ulogu prilikom procesiranja i konvertiranja u druge slične formate i iz njih, poput LaTeXML-a [4].

Semantički bogat sadržaj u smislu detaljnog opisa vrsta podataka i dodavanja metapodataka (podataka za opis podataka) ima jasne prednosti povećavanja vrijednosti sadržaja u znanstvenim radovima te se već primjenjuje u e-izdavaštvu [5].

Najpopularniji su formati e-knjiga, koje većina e-čitača podržava, PDF (Portable document format) i EPUB 3 (Electronic publication) [6]. Međutim, PDF nije praktičan za čitanje na manjim ekranima zbog fiksne veličine stranice, dok je EPUB 3 tečan format koji se uvijek prilagođava veličini zaslona na kojem se prikazuje [7].

Format EPUB 3 ima svoju osnovu u otvorenim mrežnim standardima, što ima smisla posebno zbog lakoće implementacije, mogućnosti upotrebe multimedije i interakcija, postojećih knjižnica i alata te mnoštva razvojnih programera koji su upoznati s tim jezicima [8].

U vidu standardizacije e-knjige, osim navedenih prednosti, EPUB 3 (i MathML – Mathematical Markup Language [9]) također se osniva na jeziku XML (Extensible Markup Language), čije se prednosti očitavaju u interoperabilnosti, otvorenosti, primjenjivosti i proširivosti [10].

Iako e-knjige postoje više od 20 godina, a najnoviji standardi podržavaju multimedijske sadržaje i vrhunsku semantiku te odlično čitateljsko iskustvo, opcije prijeloma i grafičkog dizajna na koje su navikli grafički dizajneri pri prijelomu za tisak još uvijek većinom nisu podržane u e-čitačima. Postoje određena rješenja dinamičkog prijeloma teksta koja primjenjuju skripte kako bi se zadržao standard tipografije [11].

U cilju čitljivosti, istraživanja su pokazala da svi važni parametri koji utječu na čitljivost tiskane knjige utječu i na čitljivost u e-knjizi. Konkretno, to su elementi dizajna poput: uvlaka, stupaca, stilova paragrafa i tipografije [12].

Postoje strogi tipografski standardi prilikom prelamanja matematičkih formula, koji osiguravaju čitljivost i razumljivost sadržaja. Između ostalog: mjesto prijelaza u novi redak, uvlaka idućeg retka, ponavljanje znaka jednakosti ili matematičke operacije u idućem retku [13, 14].

Prilikom oblikovanja matematičkih formula u e-knjigama u obliku *bitmap* slika stvaraju se problemi poput efekta pikselizacije uslijed povećavanja (ako je povećavanje uopće dostupno); dugotrajnog i kompliciranog procesa izmjena formula; konverzije u druge formate; audio reprodukcije sadržaja; prijeloma dugačkih formula u novi redak s

obzirom na logično mjesto loma i zadržavanja matematičkih pravila prijeloma kako bi korisnik usvojio sadržaj bez poteškoća [15]. Prednosti su primjene MathML-a višestruke i uključuju jednostavnost izmjena, stiliziranja, eventualne interaktivnosti i skalabilnosti sadržaja. Podrška za prikaz u preglednicima i e-čitačima MathML-a raznolika je i stalno se mijenja, a često se prikaz oslanja na MathJax JavaScript knjižnicu, međutim ne podržavaju svi e-čitač JavaScript [16].

S novim stilskim mrežnim standardima (Cascading style sheets Fonts Module Level 3)[17], koji se primjenjuju i u standardu e-knjige, moguća je upotreba bilo kojeg fonta i bilo kojeg Unicode znaka [18]. U matematičkom prijelomu vrlo je važno koji će se font upotrijebiti, kako je pripremljen za digitalni prikaz te sadrži li sve potrebne znakove [19].

Dinamički prijelom teksta koji omogućuje format EPUB 3 svoj konačni izgled duguje ugrađenim algoritmima i parametrima zaslona e-čitača na kojem se prikazuje. Osim osnovnih naredbi za prijelom i nekih koje još uvijek nisu implementirane, stvarateljima e-knjiga nisu pružene opcije za kontrolu dinamike prijeloma [20].

Zbog procesiranja, digitalne analize dokumenata te konvertiranja u druge formate postoji potreba za izradom optimalnoga matematičkog sloga [21]. Osim izrade, zbog nepostojećih adekvatnih alata koji bi to omogućili važno je i odabirom formata kojim se sadržaj stvara osigurati čitateljima jednako kvalitetan prikaz sadržaja [22]. Prikaz samog sadržaja ovisi o označnom jeziku i verziji standarda koji se primjenjuje prilikom izrade, platformi na kojoj se sadržaj prikazuje te načinu izrade sadržaja [23].

Važno je čitateljima omogućiti jednak standard čitljivosti bez obzira na uređaj kojim se koriste (veličinu ili proizvođača), sadržaj mora biti dostupan svima, bez obzira na individualna svojstva, te mora biti prilagodljiv u smislu veličine pisma [24].

## 1.2. Cilj i hipoteze

Cilj disertacije je definirati izradu optimalnog matematičkog sloga u e-knjigama čiji sadržaj mora biti semantički ispravan, tečan i prelomljen po tipografskim standardima za matematičke formule, što je preduvjet za procesiranje, pretraživost i audio reprodukciju sadržaja na raznim e-čitačima.

Postavljene su hipoteze:

1. Optimalan matematički slog u e-knjigama za različite platforme dobit će se upotrebom matematičkoga označnog jezika.
2. Kontrolom semantike matematičkog sloga dobiva se mogućnost njegova procesiranja, pretraživosti i korištenja pomoćnim tehnologijama u e-knjigama.
3. Dinamički prijelom matematičkog sadržaja prema tipografskim standardima matematičkog sloga moguće je ostvariti skriptnim i označnim jezikom.

### 1.3. Očekivani znanstveni doprinos

Matematički slog u e-knjigama neistraženo je područje u kojem se očekuje mnogo novih i inovativnih rješenja. Tehničke knjige i njihovi dijelovi sadržaja u obliku matematičkog sloga sve će se više i više prezentirati i prebacivati u e-knjige. U ovoj se disertaciji za to područje očekuju tri znanstvena doprinosa:

1. izrada optimalnog i tipografski pravilnog dinamičnog matematičkog sloga u e-knjigama korištenjem matematičkoga označnog i skriptnog jezika.
2. oblikovanje semantički ispravnoga matematičkog sloga koji omogućuje njegovo procesiranje, pretraživost i korištenje pomoćnim tehnologijama u e-knjigama
3. definiranje parametara i njihovih optimalnih vrijednosti, koje je potrebno pratiti pri odabiru formata e-knjige, dinamičkoga matematičkog sadržaja te prikaza na e-čitačima.



#### 1.4. Metodologija i plan istraživanja

Prilikom istraživanja upotrebljavat će se standardni HTML/XML/JavaScript uređivač teksta za kreiranje datoteka formata EPUB, što uključuje i MathML, opisni jezik za matematičke formule, te ePub Zip/Unzip program za pakiranje EPUB knjige. Za testiranje prikaza upotrebljavat će se različiti e-čitači na različitim platformama: Radium (Chrome), EPUBReader (Firefox), iBooks (iOS), Gitden Reader (Android), Calibre (OS, Windows). Primjenjivat će se jezici XHTML, XML i MathML za opis sadržaja koji su dio standarda EPUB te skriptni jezik JavaScript. Tehnologija koja će se primjenjivati za oblikovanje sadržaja jest jezik CSS3.

Plan istraživanja izvodi se u sljedećim fazama:

1. istraživanje različitih standarda izrade e-knjige
2. istraživanje načina uvrštavanja matematičkih formula u e-knjigu
3. izrada modela e-knjige u standardu EPUB 3 s jezikom MathML za matematički slog
4. testiranje prijeloma matematičkog sadržaja na različitim e-čitačima
5. uvrštavanje JavaScript skripti za postizanje optimalnoga dinamičnog matematičkog sloga
6. klasifikacija e-čitača i prikazivača matematičkog sloga.

Nakon provedenog istraživanja izvršit će se klasifikacija i međusobno vrednovanje postojećih standarda izrade e-knjiga. Parametri koji se planiraju analizirati uključuju: tečnost sadržaja (prilagođava li se sadržaj veličini, razlučivosti i orijentaciji zaslona), podržanost formata (jesu li su dostupni odgovarajući softverski alati za čitanje), otvorenost tehnologije (je li format vezan za određenu aplikaciju ili tehnologiju, je li moguća konverzija, ažuriranje i arhiviranje), lakoća izrade (vremenska komponenta izrade, dostupnost alata i znanja potrebnog za izradu), mogućnost i način uvrštavanja matematičkog sadržaja i njegov prijelom. U istraživanju se metodom dedukcije i analize formata e-knjige zaključuje koji je format optimalan za izradu e-knjige dinamičkoga matematičkog sadržaja.

Nakon odabranog standarda e-knjige istražuju se različiti načini uvrštavanja matematičkog sadržaja. Istraživanje se provodi analizom dostupnih primjera,

raspoloživih metoda te korelacijom parametara oblikovanja i konzumacije ovog tipa sadržaja. Prezentiraju se i analiziraju prednosti i mane uvrštavanja matematičkih formula u obliku slike, uz pomoć označnog jezika MathML na semantički neispravan i ispravan način. Pretpostavka je da matematički sadržaj u obliku slike zahtijeva kraće vrijeme izrade i jednostavniju kontrolu prijeloma formula te tako osigurava prikaz sadržaja, ali onemogućuje korištenje pomoćnim tehnologijama, ažuriranje sadržaja, procesiranje sadržaja, pretraživanje sadržaja te prilagodbu veličini i orijentaciji zaslona. Prilikom primjene matematičkoga označnog jezika koji je semantički netočan gubi se mogućnost ponovnog obrađivanja sadržaja i korištenje pomoćnim tehnologijama, ali moguće je zadržavanje optimalnoga matematičkog dinamičkog sloga.

U nastavku istraživanja primjenjuje se metoda modeliranja u svrhu eksperimentalnog istraživanja te se izvodi analiza alata izrade i korelacija varijabli izrade. Izrađuje se konkretni model e-knjige u standardu EPUB 3, koji se temelji na otvorenim standardima jezika XHTML5, CSS3 i XML. Prednost upotrebe ovih jezika njihova je otvorenost te upoznatost stvaratelja sadržaja s tehnologijom izrade. Matematički sadržaj izrađuje se uz pomoć matematičkog označnog jezika MathML koji predstavlja standard opisa ovog tipa sadržaja na internetu i u e-knjigama. Analiziraju se dostupni alati za izradu te se izvodi korelacija jednostavnosti izrade i semantičke točnosti proizvedenog sadržaja.

Optimalnost procesa prati se jednostavnošću, brzinom i semantičkom ispravnošću izrade.

Nakon izrade semantički ispravnoga matematičkog sadržaja pristupa se oblikovanju kako bi se omogućio optimalan dinamički prikaz. Osiguranje prikaza svih potrebnih specifičnih Unicode znakova izvest će se uz pomoć tehnologije CSS3 i upotrebom fonta STIX koji je osmišljen specifično za potrebe prikaza matematičkog sloga na zaslonu te zadovoljava najviše znanstvene, tehničke i tipografske standarde.

U završnom dijelu istraživanje će se vršiti eksperimentalnom metodom uspoređivanja prikaza kreiranog sadržaja, metodom klasifikacije tehnologija prikaza matematičkog sadržaja u e-knjigama te metodom sinteze tipografskih pravila pri oblikovanju matematičkih formula. Izrađeni model e-knjige testira se na različitim e-čitačima.

Eksperimentalnom metodom uspoređuje se prikaz izrađenog sadržaja na različitim platformama – iOS (iBooks), Android (Gitden Reader), OS i Windows te e-čitači u

preglednicima Chrome (Radium), Firefox (EPUBReader), kao i desktop e-čitač Calibre. Uspoređuju se prikazi s obzirom na tehnologiju prikazivača koja može biti ugrađeno svojstvo ili putem MathJax JavaScript knjižnice. Pritom se izrađuje klasifikacija e-čitača i prikazivača matematičkog sloga koje upotrebljavaju. Ako optimalan matematički slog nije moguće izvesti uz pomoć osnovnih jezika MathML te CSS3, koji moraju biti podržani u e-čitačima prema standardu EPUB 3, u proces izrade uvrstit će se skriptni jezik JavaScript kojim će se definirati stroga pravila prijeloma. Skriptni jezik ubacuje se nakon stvaranja i oblikovanja sadržaja zbog mogućnosti da ga e-čitači neće podržati te kako bi se izbjeglo povećanje kompleksnosti procesa izrade i dugotrajnije procesiranje konačnog prikaza. Pravila koja se moraju obuhvatiti jesu: mjesto prijeloma koje mora biti na mjestu nekog matematičkog operatora ili znaka jednakosti. Idući redak mora započeti matematičkim operatorom ili znakom jednakosti zbog razumijevanja čitatelja da je riječ o jednoj, a ne o više formula. Idući redak mora također imati uvlaku zbog čitljivosti i razumijevanja. Prati se kvaliteta prikaza u različitim segmentima tijekom procesa izrade i pri završetku:

1. prikazuje li se sadržaj
2. nedostaju li neki specifični znakovi potrebni za čitljivost i razumijevanje
3. prelama li se sadržaj na logičnim mjestima prema tipografskim standardima matematičkog sloga
4. je li osigurana pristupačnost sadržaja (povećavanje fonta, audio reprodukcija)
5. prilagođava li se sadržaj veličini, razlučivosti i orijentaciji zaslona.

Na temelju uzorka i analize definirat će se proces izrade optimalnoga dinamičnog matematičkog sloga u e-knjigama.

## **2. ANALIZA FORMATA E-KNJIGA**

E-knjige kreiraju se u „otvorenom” i „zatvorenom” formatu. Otvoreni formati e-knjiga (EPUB) oblikovani su kao neovisni o uređaju, odnosno tako da se mogu čitati na bilo kojem uređaju koji podržava taj format. Zatvoreni formati e-knjiga (AZW, PDF) često su osnovani na otvorenom formatu, ali ponovno konfigurirani da se mogu čitati na specifičnom e-čitaču, a ne na drugima, zbog marketinške strategije. Općenito, otvoreni formati mnogo su pristupačniji od zatvorenih formata.

Svi trenutačno važniji formati e-knjiga nasljednici su ranijih standarda OEBPS ili EPUB 2.0.1; to se ne odnosi samo na EPUB 3.0 već i na Amazonov format KF8 [25].

## 2.1. Format Mobipocket

Amazon upotrebljava Mobipocket kako bi zadržao vlasničku (eng. *proprietary*) prednost nad konkurencijom. S tehničke strane nema nijednog razloga zašto bi zadržao podršku za format koji je manje razvijen od formata EPUB 2.1. Mobipocket se osniva na standardu OEBPS – Open eBook Publication Structure (1.0, 2.0), koji je združen s XHTML-om, JavaScriptom i Frameovima. Kreiran je u francuskoj tvrtki Mobipocket SA koju je Amazon kupio 2005. godine. Format Mobipocket (.mobi ili .prc) namijenjen je tekstualnom sadržaju (i slikama) i ne podržava multimediju (uključujući audio) ni interaktivnost.

Mobipocket je tečan format, što znači da se sadržaj e-knjige uvijek prilagođava veličini ekrana e-čitača. Korisnik (čitatelj) može sam birati postavke odabira fonta, veličine pisma, veličine margine, proreda i sl., bez obzira na početne vrijednosti koje su odredili kreatori e-knjige. Slike se mogu ugrađivati u sadržaj u različitim rezolucijama, koje se tada mogu učitavati ovisno o rezoluciji i veličini e-čitača. Dokumente je moguće zaštititi od piratstva uz pomoć DRM sheme. S obzirom na to da se osniva na standardu Open Publication Structure (OPS), Mobipocket sadrži dokument .OPF XML koji sadrži metapodatke o e-knjizi i popis svih HTML dokumenata koje e-knjiga sadrži. Mobipocket Creator, softverski alat za kreiranje formata Mobipocket, pakira sve sadržaje u jedan dokument, optimizira ga i komprimira, indeksira (zbog mogućnosti pretraživanja) i po potrebi vrši enkripciju. Mobipocket ima identičnu strukturu dokumenta poput formata EPUB, ali s dodatnim elementima potrebnima za renderiranje na Mobi i Amazonovim uređajima. Osim na Kindleu, može se čitati i na Mobipocket Readeru na platformi Windows, uređajima Blackberry i uređajima Nokia (Symbian) [26].

Uz pomoć raznih alata (poput Calibre) moguće je konvertirati Mobipocket dokument u drugi željeni format poput PDF-a ili EPUB-a. Konverzije znaju producirati neželjene rezultate prijeloma (npr. svaka rečenica postavljena je u vlastiti paragraf), ovisno o originalnom formatiranju. Ako je u e-knjigu uključena enkripcija u obliku DRM-a, konverzija više nije moguća, kao ni arhiviranje (kopiranje), bez uklanjanja DRM-a. Uklanjanje DRM-a u većini se slučajeva i zemalja smatra nelegalnim.

Najsigurniji način kako uključiti matematički sadržaj u format Mobipocket jest putem

slike, jer ne podržava matematički označni jezik. Proces izrade takvog sadržaja dugotrajan je i kompliciran. Najprije se moraju kreirati matematičke formule u odabranom softveru (npr. Microsoft Word ili Adobe InDesign), zatim se svaka eksportira u obliku slike pa se uvrštava u HTML potreban za knjigu. Slike također mogu biti premale zbog prilagođavanja ekranu pa je svaku potrebno ručno povećavati prilikom čitanja, što je zamoran i nepraktičan proces.

## **2.2. Format AZW (Amazon Kindle format)**

Identičan je formatu Mobipocket s DRM zaštitom i dodatnom kompresijom. DRM onemogućuje korisnika da dijeli knjigu te je zaključava na dotični uređaj. Format AZW bez DRM zaštite jest zapravo samo format Mobipocket u komprimiranoj varijanti. AZW je zatvoreni format kojim se koriste Amazonovi Kindle e-čitači. Ovaj se format može čitati isključivo na uređaju Kindle ili aplikaciji Kindle [27].

### 2.3. Format KF8 (Kindle Format 8)

Amazon 2012. godine pušta svoj vlastiti novi format za e-knjige KF8 namijenjen čitanju na novom uređaju Kindle Fire. Taj XML format zamjenjuje dosadašnji format Mobipocket, odnosno format AZW. KF8 se distribuira isključivo putem trgovine Amazon Kindle te se može čitati na uređajima Kindle ili aplikacijama Kindle (na različitim platformama) ili izravno u preglednicima. Specifičnost za uređaje Kindle (osim Kindle Fire) jest da koriste tehnologiju e-ink koja pruža opciju čitanja isključivo u crno-bijeloj varijanti. Strogo je zabranjeno uvjetima Amazona da se e-knjige kupljene putem njihova dućana prebacuju na uređaje drugih proizvođača. Upotrebom Amazonovih alata za distribuciju e-knjiga autor i izdavač obvezuju se na isključivu distribuciju putem trgovine Kindle prvih 90 dana nakon izdavanja.

Tehnički, KF8 podržava HTML5 i CSS3, a zbog kompatibilnosti sa starijim uređajima sadrži dva formata: Mobipocket i strukturu podataka sličnu EPUB-u. Detaljna specifikacija nije javno dostupna. KF8 se kreira putem alata KindleGen ili dodatka za Adobe InDesign. KindleGen je alat komandne linije koji konvertira formate HTML, XHTML ili EPUB u KF8.

KF8 podržava audio i video putem HTML5 elemenata, no ne mogu svi uređaji (aplikacije) Kindle prikazati takav sadržaj. Bilo kakav *text-to-speech* ili mehaničko iščitavanje nisu podržani u ovom formatu, odnosno na uređaju Kindle Fire. Kao i EPUB 3, KF8 sada podržava CSS Media Queries, SVG i fiksni prijelom stranice, ali ne podržava JavaScript, *canvas* element ni matematički označni jezik (MathML). Amazon je kreirao zatvoreni, nekompatibilni sustav izdavanja, kreiranja i čitanja e-knjiga u svojem vlastitom formatu. Međutim, još gore od Applea, Amazon ne dozvoljava čak ni čitanje formata EPUB na svojim uređajima [27].

KF8 je tečan format, koji se prilagođava veličini ekrana te omogućava pretraživanje i postavljanje korisničkih preferencija prijeloma (pismo, veličina, prored, margine). Konvertiranje je moguće i preporučeno primjenom Amazonovih alata KindleGen i dodatka za Adobe InDesign. Arhiviranje (i kopiranje) nije moguće zbog obvezne upotrebe DRM-a prilikom distribucije koja nužno mora ići putem Amazonova dućana.



Ugrađivanje matematičkog sadržaja putem označnog jezika nije moguće u ovom formatu, za matematički sadržaj moraju se upotrebljavati ili *bitmap* slike ili SVG.

## 2.4. Format EPUB

Cilj kreiranja standarda EPUB bio je razvitak otvorenog standarda e-knjige koji se osniva na XML-u, koji bi omogućio upotrebu e-knjiga neovisnih o platformi.

EPUB se smatra standardom industrije, format oko kojeg su se „svi” složili (IDPF), osim Amazona (IDPF se sastoji od članova predstavnika industrije kao što su Barnes&Noble, Apple i Sony). International Digital Publishing Forum (IDPF) jest krovna organizacija koja koordinira standard e-knjiga, preimenovana 2005. godine iz inicijalne organizacije Open eBook Forum, stvorene 1998. godine. U to su vrijeme svi formati bili zatvoreni, odnosno svaki se format mogao čitati na svojem vlastitom e-čitaču. Iz prvog kreiranog standarda OEBPS (Open eBook Publication Structure) 1.0 proizašlo je mnogo „ekstenzijskih” formata prilagođenih distribuciji kao što je Mobi ili Microsoftov LIT. IDPF kao odgovor tim novonastalim formatima osniva novu grupu koja se bavi kreacijom kontejner formata koji može spojiti sve dokumente e-knjige tako da se može lako distribuirati. EPUB 2.0 2007. godine postaje službeni standard i zamjenjuje stari OEBPS 1.2 [27].

## 2.5. Format EPUB 2

EPUB 2.0.1 jest standard za e-knjige od 2010. godine i sastoji se od tri djela: Open Publication Structure (OPS) koji opisuje sadržaj formata; Open Packaging Format (OPF) koji definira strukturu XML dokumenta; Open Container Format (OCF) koji komprimira sve dokumente u ZIP arhivu. OPS upotrebljava XHTML 1.1 za tekst i strukturu teksta te dio CSS-a 2.0 za dizajn, a podržava i vektorsku grafiku (SVG) te standardne formate za slike kao što su JPG, PNG i GIF. EPUB 2 namijenjen je za tekstualne knjige, čiji se sadržaj prilagođava veličini ekrana na tečan način pa se može čitati na uređajima različitih veličina zaslona. Zbog mana formata EPUB 2 (najviše manjak podrške multimedije i interaktivnosti) kreiran je format EPUB 3.

Tehnologije na kojima se osniva EPUB 2.0.1 jesu HTML4 i CSS2 te su orijentirane na izradu tekstualnih sadržaja. U ovom formatu iznimno je limitirana izrada multimedijских i interaktivnih sadržaja. Zato što HTML4 nema inkorporiran označni jezik MathML, nije moguće kreiranje matematičkih formula označnim jezikom, već isključivo uz pomoć slika.

Tablica sadržaja EPUB 2.0.1 jest XML dokument s ekstenzijom .ncx koja je bila nepraktična u vidu izrade i razumijevanja koda te teška za upotrebu [27].

EPUB 2 se može čitati na većini e-čitača kao što su Nook, Sony Reader, Apple iBooks i Adobe Digital Editions. Na Amazonovom e-čitaču Kindle nije moguće bez konverzije čitati EPUB 2.

## 2.6. Format EPUB 3

EPUB 3 kreiran je 2011. godine i sastoji se od četiri specifikacije: EPUB Publication 3.0 koji definira semantiku i sveobuhvatne zahtjeve usklađenosti; EPUB Content Documents 3.0 koji definira oblikovanje sadržaja; EPUB Open Container Format (OCF) 3.0 koji definira format dokumenta i pakiranje u ZIP kontejner; EPUB Media Overlays 3.0 koji definira format i model procesiranja za sinkronizaciju teksta i zvuka. EPUB 3 osniva se na tehnologijama HTML5 i CSS3 koje omogućuju izradu e-knjiga s multimedijским, interaktivnim sadržajem, kompleksnih prijeloma i matematičkih formula. Također je moguće kreiranje nelinearnih dokumenata i funkcija *text-to-speech*. Za razliku od formata EPUB 2, moguća je i upotreba skriptnog jezika JavaScript, što između ostalog omogućuje kreiranje interaktivnih i nestatičnih e-knjiga. Iako se osniva na standardu za označavanje sadržaja HTML5, koristi se XML inačicom tog standarda XHTML5, koja podrazumijeva XML sintaksu. Na taj je način olakšano automatsko procesiranje dokumenata. Gotovo sve prednosti formata EPUB 3 u odnosu na EPUB 2 proizlaze iz prednosti tehnologija HTML5 i CSS3 u odnosu na HTML4 i CSS2. Uvijek su popularna nova podržana svojstva `<video>` i `<audio>` element, HTML element `<canvas>` za crtanje i interaktivnost, slobodna primjena SVG-a za vektorske grafike, inkorporiranost označnog jezika MathML za matematičke formule, novi elementi forma, slobodna primjena bilo kojeg fonta (i japanskih i azijskih pisama i smjerova čitanja) te sinkroniziranost audio naracije sa sadržajem.

Novi HTML5 element `<nav>` zamjenjuje staru tablicu sadržaja .ncx koja je jednostavna za čitanje i interpretiranje.

Novi moduli tehnologije CSS3 koji su podržani u formatu EPUB 3 su modul Speech (za kontroliranje i manipulaciju mehaničkog iščitavanja) i Media Queries (za kontroliranje prilagođavanja sadržaja veličini zaslona).

Nepostojanje e-čitača koji u potpunosti podržava sva svojstva koja nudi ovaj format razlog je tome što izdavači sporo implementiraju ovaj standard. Zbog toga je kreiran Readium (IDPF, Adobe, Google, Barnes&Noble, Kobo, O'Reilly, Samsung, Sony), otvoreni sustav referenciranja i mehanizam renderiranja za EPUB 3. Cilj je projekta

izgradnja referentnog sustava za izgradnju EPUB 3 e-čitača. Radiumu se može pristupiti putem preglednika Google Chrome [27].

Malo softvera podržava mogućnosti koje nudi ovaj format (bilo za kreiranje bilo za čitanje). Malo je ljudi upoznato s kreiranjem formata u praktičnom smislu pa nije dostupno mnogo informacija ili savjeta.

EPUB 2 polagano je zamijenjen novijom varijantom standarda EPUB 3 koji je sofisticiraniji. Podržava sve što podržavaju i najnovije internetske tehnologije: najnoviju tipografiju, video, animacije, zvučne sadržaje... Usvajanje ovog novog formata problematično je iz više aspekata. Najprije, ne postoji podrška u obliku softvera za kreiranje takvih e-knjiga, zatim ne postoje uređaji koji u potpunosti podržavaju sve što nudi EPUB 3. Izdavači nisu navikli na takve tipove sadržaja u knjigama pa su nespremni za nove, kompleksnije i skuplje proizvodne procese (npr. do sada nitko nije morao kreirati i pripremati video za knjigu). Adaptacija tržišta još je u punom jeku. EPUB 3 mijenja mogućnosti i definicije e-knjige te uklanja tehnološke prepreke koje su do sada prisiljavale e-knjige da budu isključivo tekstualnog sadržaja. Proces izrade je stoga još uvijek dugotrajan i skup [28].

Tehnologija reproduciranja odnosno mehaničkog iščitavanja u osnovnom obliku ovisi isključivo o tome uključuje li e-čitač uključuje TTS (*text-to-speech*) tehnologiju ili ima pristup takvoj pomoćnoj tehnologiji. Međutim, kompleksniji izrazi vokabulara često budu izrečeni na krivi način bez poboljšanja. EPUB 3 dodaje tri nove komplementarne tehnologije kojima se ta poboljšanja mogu izvesti. Pronunciation Lexicon Specification (PLS) definira XML bazu podataka koja sadrži izgovore riječi čije se iščitavanje ne mijenja s obzirom na kontekst. Synthetic Speech Markup Language dozvoljava da se izgovori unesu izravno u označni jezik te se upotrebljavaju u kombinaciji s PLS leksikonima za izgovore u posebnim situacijama poput heteronima. CSS Speech Modules sadrže razne atribute za kontrolu izgovora poput sricanja riječi i brojki ili ugrađivanja pauza. Nažalost, još uvijek ne postoje e-čitači koji podržavaju ove tri tehnologije, iako postoje razni EPUB 3 e-čitači koji imaju osnovnu tehnologiju mehaničkog iščitavanja [29].

EPUB je format osnovan na otvorenim standardima, ali nudi mogućnost ugrađivanja DRM protekcije. Samo ugrađivanje u potpunosti ovisi o samim izdavačima i/ili distributerima. Ako e-knjiga nema DRM, arhiviranje, kopiranje i dijeljenje sadržaja u

potpunosti je omogućeno. Za razliku od Amazona i Applea, ugrađivanje DRM sheme jest opcionalno.

## 2.7. Format iBooks (IBA)

Apple kreira svoj vlastiti zatvoreni format koji se također osniva na formatu EPUB 2.0.1, ali ima određena nadodana svojstva koja nedostaju formatu EPUB 2. Apple sudjeluje u kreiranju standarda EPUB 3 te se posljedično velik dio tog standarda (npr. osnovni koncepti izrade i distribucije te strukture podataka) može pronaći u formatu iBooks. Apple 2010. godine pušta u distribuciju uređaj iPad s aplikacijom iBooks za čitanje e-knjiga (PDF i EPUB). Također kreira i novi format Apple Fixed Layout EPUB namijenjen za sadržaj koji je neodvojiv od slika, poput kuharica, putopisa, knjiga fotografija ili dječjih knjiga. Taj se format može čitati isključivo na uređajima s operativnim sistemima iOS 4 i više te nije podržan na drugim e-čitačima koji podržavaju EPUB. Godine 2012. puštaju softver iBooks Author koji olakšava izradu interaktivnih i multimedijских e-knjiga u tom zatvorenom formatu. Takve e-knjige mogu se kreirati od kuće bez mnogo znanja o razvojnom programiranju ili formatima i distribuirati isključivo putem Apple iBooks Storea s ekstenzijom .ibooks. E-knjige kupljene putem iBooks Storea na sebi imaju DRM zaštitu, što onemogućuje korisnike da međusobno dijele knjige. Prednost formata iBooks je to što je kreiran da optimalno funkcionira na Appleovim uređajima (primarno na iPadu) pa kreator sadržaja ima osiguranje da svi čitatelji imaju pristup tom sadržaju na jednak i optimalan način. Postoje i određene restrikcije prilikom kreacije koje se odnose na veličinu dokumenta, naslovne slike, veličine videosadržaja i nekih CSS svojstva. Format također podržava funkcionalnost *text-to-speech* i skriptiranje, ali je fiksna, što znači da se ne prilagođava veličini zaslona (zamišljen da se čita na uređaju iPad). Nakon izlaska specifikacije EPUB 3 Apple odbacuje ovaj format i prelazi na punu podršku formata EPUB 3, iako i dalje postoje izdavači koji preferiraju ovaj format i izdaju putem iBooks Storea.

Od 2012. godine iBooks podržava ugrađivanje fontova, napredne matematičke formule i interaktivnost. Za razliku od formata EPUB 3, iBooks ima svoje vlastite *namespaceove* i CSS ekstenzije. Može se čitati isključivo na aplikaciji iBooks na uređajima sa iOS operativnim sistemima. Kreiranje je vrlo jednostavno putem besplatne aplikacije iBooks Author. Najveća mana formata jest da je dio zatvorenog Appleova ekosustava iz kojeg ne može izići. Izdavači i kreatori sadržaja prisiljeni su izdavati knjige putem Apple iBooks

Storea, a čitatelji čitati putem aplikacije iBooks. Format je moguće eksportirati u PDF putem aplikacije iBooks Author, ali ni u jedan drugi format [27].

Pretraživanje unutar samog e-čitača (aplikacije iBooks) u potpunosti je omogućeno, isto kao i prilagođavanje čitateljevim preferencijama fonta, veličine i boje pozadine. E-knjige kupljene putem iTunes Storea također omogućavaju automatsko ažuriranje poput bilo koje druge aplikacije. Također, iBooks nudi opciju kreiranja tečnog formata koji se prilagođava ekranu ili pak fiksnog formata koji je ponekad potreban pogotovo kod dječjih knjiga s ilustracijama ili manga stripova. Aplikacija iBooks Author omogućuje automatsko kreiranje e-knjiga s pristupačnošću za one s vizualnim invaliditetom iskorištavajući svoju vlastitu tehnologiju Voice Over, koja je trenutačno najnaprednija na tržištu. Najveća mana formata jest da je zatvoren te namijenjen prvenstveno čitanju na iPadu (iako je moguće i na ostalim Appleovim uređajima).

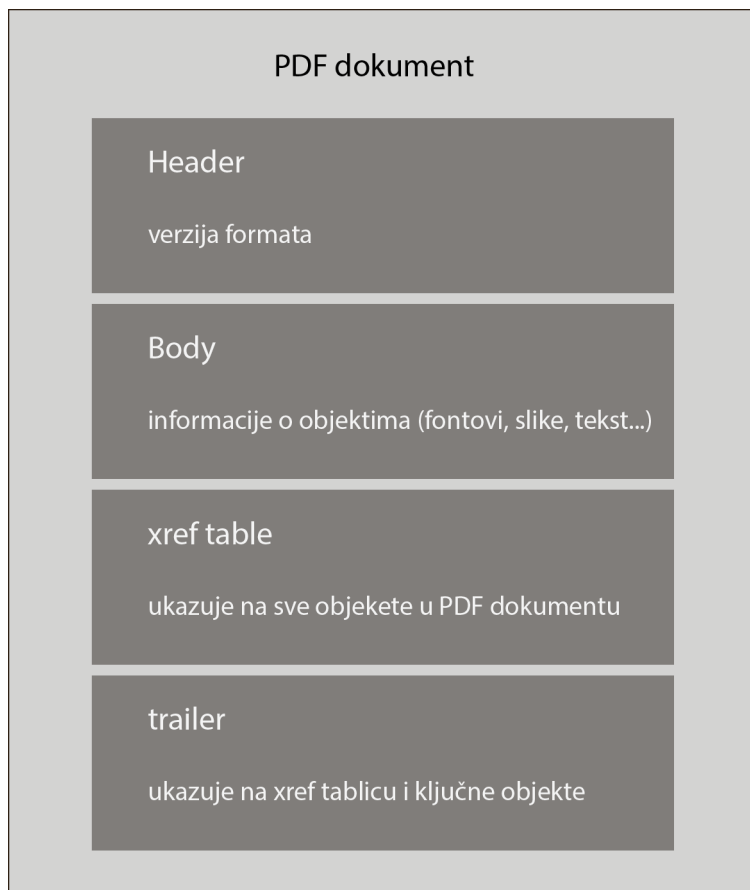


## 2.8. Format PDF (Portable Document Format)

PDF je format koji svoj izgled duguje tisku i otisnutoj stranici. Osim u Americi, veličina te stranice najčešće je A4. Prelaskom iz analogne knjige (otisnute) u elektroničku verziju knjige mijenja se definicija „stranice”. Stranica je do sada fiksna ploha u koju se smješta tekst i/ili slike i kao takva definira očekivanja i ponašanja korisnika. Ta naučena ponašanja toliko su snažna da su djelomično odredila korisnička ponašanja modernih uređaja koji se drže u ruci (*handheld* – pametni telefoni i tablična računala), poput „listanja” između ekrana. Do razvitka e-knjiga nitko se nije bavio razmišljanjima o tome što takav medij dopušta ili onemogućuje. PDF je format za prikazivanje dokumenata neovisno o softveru, hardveru, operativnom sustavu kojim će se izrađivati, ali i neovisno o tome na čemu će se čitati. Može se kreirati softverom Adobe Acrobat, ali i mnogim drugim softverima koji nisu Adobeovi. U početku je PDF bio namijenjen prikazu statične otisnute stranice, no specifikacija se s vremenom mijenja te danas podržava interaktivne elemente (npr. poveznice, zvuk i video) i strukturirano označavanje sadržaja (npr. XML), što kasnije može koristiti za pristupačnost ili interoperabilnost. PDF je danas jedan od najraširenijih formata koji se može s lakoćom kreirati na širokom spektru softvera, ali i čitati na bilo kojem uređaju. Dokumenti u formatu PDF jednostavno nude većinu atributa papirnatih dokumenata (struktura stranice, napredna grafika i detaljan dizajn, uz navedene elektroničke mogućnosti). PDF je također među prvim formatima koji se može nazvati e-knjigom. Najveća je prednost i istovremeno mana ovog formata njegova fiksna i strogo kontrolirana stranica [28].

Većina uređaja može učitati PDF dokumente, jednako kako ih mnogo softvera može kreirati, ali baš zbog te raznolikosti PDF često bude nepristupačan. Može se učiniti pristupačnijim pravilnim semantičkim označavanjem i uključivanjem alternativnog teksta za grafike. Bez takvog će pristupa PDF biti nepristupačan za tehnologiju mehaničkog iščitavanja [25].

Postoje razni softverski alati koji omogućuju konvertiranje PDF dokumenata u druge formate e-knjiga, međutim često se pojavljuju greške u formatiranju.



Slika 1. Struktura PDF dokumenta

## 2.9. Evaluacija pristupačnosti formata i uređaja

Tiskane se knjige proizvode na dva potpuno različita načina kada se oblikuju za pristupačnosti osobama slabijeg vida ili slijepima. Tiskana je knjiga fiksna u svojim mjerama i veličini pisma pa ne postoji način kako je prilagoditi drugačijim potrebama. Za slijepe postoji Brailleovo pismo za koje se knjige moraju posebno pripremati, ali i zvučne knjige koje je potrebno snimati. Prelaskom u digitalno okruženje, knjige i tehnologije također se prilagođavaju. Konzorcij Daisy (Digital Accessible Information System) formiran je 1996. godine u cilju da omogući prijelaz analognih zvučnih knjiga u digitalne. Daisy je otvoreni sustav čije se digitalne knjige sastoje od audio naracije sadržaja, dokumenta sinkronizacije zvučnih i tekstualnih dokumenata te navigacije. Kreiranje zvučnih knjiga, bilo analognih bilo digitalnih, zahtijeva znatno vrijeme i resurse pa se već dulje pokušava pronaći lakši način za pristupačnost digitalnim sadržajima. Te tehnologije najprije se stvaraju za omogućavanje pristupa sadržaju na internetu. E-knjige predstavljaju velik potencijal za ljude s invaliditetom u smislu pristupa informacijama koje bi im inače bile nedostupne, ako su e-čitači opskrbljeni tehnologijom koja omogućava pristupačnost slijepim i slabovidnim osobama (npr. Screen reader, Braille, Screen magnifiers), a e-knjige oblikovane adekvatnim tehnologijama. Posljednjih godina tehnologija je znatno napredovala u kvaliteti, posebice kvaliteti izgovora, te se proširila i na „obične” korisnike, a ne samo one zakinite za vidne sposobnosti. Naime, nemaju samo slijepci potrebu za pristupačnošću, postoje mnoge situacije u svakodnevnom životu kada je zgodnije poslušati neki sadržaj umjesto čitanja, npr. kada su ruke zauzete nekim alatima ili upravljanjem vozilom, kada svjetlo nije optimalno za čitanje ili pri učenju stranih jezika pri čemu je ključno čuti izgovor. Semantički *web* polagano postaje realnost s HTML5 jezikom koji značenje (semantiku) informacija definira tako da ga razumiju ljudi i strojevi. Semantički *web* na taj način omogućuje i primjenu naprednih *text-to-speech* tehnologija, odnosno mehaničkog iščitavanja sadržaja i pretvaranja svojeg sadržaja u pristupačan. Kako bi *text-to-speech* optimalno funkcionirao moraju biti zadovoljena dva uvjeta: sadržaj mora biti optimalno opisan označnim jezikom na semantički korektan način te uređaj na kojem se sadržaj iščitava mora imati ugrađenu

takvu funkciju. U slučaju iščitavanja sadržaja e-knjiga, sadržaj mora biti pripremljen na adekvatan način i e-čitač mora imati ugrađenu posebnu tehnologiju.

Prema sekciji 508 smjernica *Technology Accessibility Standards & W3C Web Content Accessibility Guidelines 2.0* (WCAG), e-knjiga je pristupačna kada hardver ili softver ima barem jedan način korištenja i dolaženja do informacija za koje nije potreban vid, za one koji slabo vide, koji ne čuju (ili slabo čuju), za koje nije potrebna fina motorika ruku ili simultani pokreti, koji se može upotrebljavati s malom snagom ili dosegom. WCAG 2.0 ima četiri glavna principa:

1. primjetljiv: informacije i komponente korisničkog sučelja moraju biti predstavljene korisnicima na način da su primjetljive
2. operabilan: komponente korisničkog sučelja i navigacija moraju biti operabilne
3. razumljiv: informacije i operacije korisničkog sučelja moraju biti razumljive
4. robustan (jasan): sadržaj mora biti dovoljno jasan da se može sigurno interpretirati s različitim korisničkim posrednicima, uključujući i pomoćnu tehnologiju [30].

Najveća prepreka u pristupačnosti e-knjiga jest kada sadržaj nije oblikovan na bazi teksta. Upotreba slika umjesto teksta onemogućava pomoćnoj tehnologiji da procesira sadržaj tako da ga može obraditi za mehaničko iščitavanje. Također, slike u e-knjigama znatno je teže povećavati i čitati, s obzirom na rezoluciju i samu otežanu tehniku čitanja sadržaja (*zoom&pan*). Vlasnički zatvoreni formati s uključenim DRM-om također kompliciraju pristupačnost te su zaključani na određeni uređaj, koji onemogućava pristup sadržaju ako nema opciju pomoćne tehnologije. Najpristupačniji formati e-knjiga uključuju EPUB 3 i iBooks (ali isključivo na Appleovim uređajima). Najlošije opcije pristupačnosti nudi uređaj Kindle Fire sa svojim vlastitim formatom KF8 koji nema opciju korištenja pomoćnim tehnologijama.

### 3. ANALIZA PRISTUPAČNOSTI E-ČITAČA

Postoje e-čitači u obliku uređaja čija je jedina funkcija čitanje e-knjiga (Kindle i Sony) i multifunkcionalni uređaji na kojima se također mogu čitati e-knjige (iPad). Ali postoje i aplikacije e-čitači koji se instaliraju na željeni uređaj (mobilni telefon, tablično računalo ili osobno računalo). Također postoje i e-čitači bazirani na internetskom pregledniku. Prema tehnologiji prikaza također se dijele na one koji se koriste tehnologijom e-ink i one s LCD ekranima. E-čitači namijenjeni isključivo čitanju e-knjiga uobičajeno su upareni s tehnologijom e-ink. Oni s tehnologijom e-ink, naravno, imaju mnogo dulju izdržljivost baterije te lakše čitanje na direktnoj Sunčevoj svjetlosti nego oni s pozadinski osvjetljenim LCD-om. E-ink prikaz češće je crno-bijeli, tek noviji imaju prikaz boja. LCD e-čitači mogu kristalno jasno prikazati slike svih boja i videozapise visokih rezolucija. Mnogi e-čitači dolaze opremljeni standardnim opcijama ispomoći osobama s invaliditetom. Tipična je mogućnost korištenje tehnologijom *text-to-speech* koja mehanički iščitava tekst. Dobre *text-to-speech* aplikacije dozvoljavaju korisnicima da podese brzinu čitanja, označavaju riječ koja se čita nekom bojom te imaju izbornike i tablice sadržaja koje se mogu poslušati. Ova tehnologija ima još uvijek mnogo mana jer se oslanja na računalno čitanje koje ne zvuči kao ljudsko, te još uvijek nema mnogo podrške izvan engleskoga govornog područja. Ipak omogućava slijepima da pristupe sadržaju koji bi im inače bio nedostupan. Pristupačnost na e-čitačima također uključuju mogućnosti povećanja veličine pisma.

E-čitači Kindle tvrtke Amazon dolaze u dva oblika: e-čitač u obliku uređaja (hardver) i e-čitač u obliku softvera. Uređaji Kindle (svi osim Kindle Fire) imaju funkciju *text-to-speech* koja iščitava sve elemente izbornika i sadržaj. Kindle Fire ne dolazi standardno s opcijom *text-to-speech*. Kindle čita AZW, Mobi, KF8 i PDF, a ne čita EPUB.

E-čitač Barnes&Noble Nook također dolazi u varijantama hardvera i softvera. Nook čita EPUB (s posebnim DRM-om) i nema opcije za pristupačnost.

Apple je razvio softver iBooks za iOS (iPad, iPhone i novija Appleova računala) te podržava formate EPUB, PDF i iBooks. Dolaze s ugrađenim softverom Voice Over za iščitavanje. Trenutačno predstavljaju najpristupačniju tehnologiju na tržištu [31].

Adobe Digital Editions jest desktop e-čitač i kao takav se koristi tehnologijom dotičnog operativnog sustava (uključujući i *text-to-speech*). Namijenjen je za čitanje formata EPUB 2 i EPUB 3 te PDF-a.

#### 4. KREIRANJE E-KNJIGA RAZLIČITIH FORMATA

Oblikovanje e-knjige uvelike je slično kreiranju mrežne stranice. Potrebno je upoznati se sa specifikacijom dotičnog formata te restrikcijama e-čitača kojima je taj format namijenjen. Nije moguće kreirati univerzalni dokument za sve e-čitače, osim u slučaju PDF dokumenta.

Format PDF najlakši je i najbrži za oblikovanje zahvaljujući velikom broju softverskih alata koji to omogućuju, ali i fiksne veličine stranice koja se ne prilagođava uređajima. Također, nikakva posebna predznanja nisu potrebna prilikom kreiranja ovog formata, osim u slučaju semantičkog označavanja sadržaja.

Cjelokupnu specifikaciju EPUB objavio je IDPF na svojoj mrežnoj stranici. Posljednja je specifikacija EPUB 3.01 objavljena 2014. godine, ali sadrži samo manje izmjene od posljednje velike izmjene EPUB 3 iz 2011. godine. EPUB se može kreirati raznim softverskim alatima kao što su Adobe InDesign, Calibre, Apple Pages i Jutoh. Međutim, za kvalitetno oblikovane e-knjige formata EPUB s korektnom semantikom, kreirane prema specifikacijama IDPF-a, potrebna je ili postproizvodnja ili „ručno“ oblikovanje upotrebom uređivača teksta za programiranje. Pa tako postoje alati (npr. Jutoh) za „popravljanje“ semantičke nekorektnosti koja se proizvodi InDesignom, ali i popularni konverteri. Potrebna su posebna znanja, iako se osniva na jezicima HTML, CSS i XML, postoje posebne restrikcije prilikom primjene određenih elemenata i atributa, ali i posebne mogućnosti specifične za e-knjige. Najosnovniji format e-knjige može se kreirati s osnovnim znanjem HTML-a i CSS-a i običnim uređivačem teksta. Iako je specifikacija preporučena 2011. godine, podrška ovisi o e-čitaču te se shodno tome mogu upotrebljavati određena svojstva. Postoje alati koji olakšavaju proizvodnju, npr. EPUBCheck (za validaciju), ePUB Zip/Unzip (za pakiranje na OS-u) i ePUB Pack (za pakiranje u operativnom sustavu Windows).

Iz formata EPUB moguće je proizvesti sve ostale formate e-knjiga raznim dostupnim alatima za konverziju.

Mobipocket se može kreirati alatom Mobipocket Creator, za koji nije potrebno posebno predznanje. Softverom se mogu konvertirati Word, PDF i tekstualni dokumenti ili nadodavati vlastiti kreirani HTML dokumenti.

Amazon ima detaljan dokument koji opisuje način kreiranja formata AZW/KF8 te nudi vlastite alate za kreiranje e-knjiga u tim formatima. Postoji mogućnost upotrebe dodatka za Adobe InDesign. KindleGen je besplatan softver za konverziju iz formata HTML, XHTML ili EPUB. Kindle Previewer Softver grafičko je sučelje koje omogućuje pregledavanje prikaza e-knjige na uređaju Kindle. Amazon specificira način upotrebe dozvoljenih elemenata HTML-a i CSS-a, ali i svoja vlastita nestandardna svojstva. Distribucija i izdavanje Amazonovih formata mogući su isključivo putem Amazona koji e-knjige zaključava svojom DRM shemom [32].

Kreiranje i oblikovanje formata iBooks moguće je isključivo korištenjem Appleova besplatnog softvera iBooks Author. Alat je namijenjen za što jednostavnije kreiranje e-knjige za autore i male izdavače kojima nije potrebno nikakvo posebno predznanje o tehnologijama HTML, CSS ili XML. E-knjige kreirane na ovaj način moraju proći kroz Appleov iTunes Store, platiti naknadu te biti zaštićene vlastitim DRM-om.



## 5. USPOREDBA FORMATA E-KNJIGA PREMA DEFINIRANIM PARAMETRIMA OPTIMALNOGA MATEMATIČKOG SLOGA

TEČNOST (engl. *reflowable*) (prilagodba veličini zaslona, rezoluciji i orijentaciji zaslona) Glavna karakteristika koja konceptualno odvaja i definira e-knjigu u odnosu na otisnutu knjige jest svojstvo da nema fizikalnih karakteristika. E-knjige u tom smislu nemaju veličinu, rezoluciju ni težinu. Smisao e-knjiga jest da je sadržaj dostupan uvijek i na svim uređajima kojima su trenutačni korisnici okruženi, poput tabličnih računala, prijenosnih računala, mobilnih telefona i uređaja namijenjenih za čitanje knjiga (e-čitača). Svojstvo prilagođavanja sadržaja veličini, orijentaciji i rezoluciji zaslona ključno je za pristupačnost sadržaja. Sadržaj se ne prilagođava u smislu smanjivanja ili povećavanja teksta, već aktivnog prelamanja sadržaja prema dostupnoj veličini i rezoluciji zaslona. Prvi koncepti e-knjiga, poput standarda OEBPS i formata Mobipocket, uvidjeli su prednost takvog koncepta koji se rodio s mrežnim tehnologijama, a pristup se nastavio i kod formata EPUB 2 i EPUB 3, nasljednika Mobipocketa formata AZW i KF8 te formata iBooks. Formati Mobi i EPUB 2.1 tečni su u smislu prilagođavanja teksta veličini ekrana. Oba se mogu generalizirati kao zapakirane mrežne stranice. Kao takve, primarna je funkcija koju obnašaju prikazivanje jednostavnih tekstualnih dokumenata, međutim kod bilo kakvog kompliciranijeg prijeloma poput tablica i grafika pokazuju nisku kvalitetu prikaza. Svojevrsan je Amazonov odgovor na EPUB 3 zatvoreni format KF8, koji je sličan (i oba se osnivaju na najnovijim W3C standardima), ali dovoljno različit kako bi produkcija zahtijevala svoje vlastite tehnološke intervencije [28].

I EPUB 3 i KF8 tečni su formati koji dozvoljavaju prilagođavanje teksta veličini ekrana. Međutim, EPUB 3 dozvoljava mnogo više funkcija za kontroliranje prijeloma i stranica i riječi ovisno o jeziku, što KF8 ne dozvoljava te prisiljava tekst da uvijek bude obostrano poravnat [27].

Jedini standard koji ne podražava aktivna prilagođavanja kao osnovno svojstvo jest PDF, čija je povijest, a time i namjena drugačija. PDF ipak spada u kategoriju formata e-knjiga po tome što je elektronički format koji opisuje knjiški sadržaj, ali i svojim drugim karakteristikama (npr. pretraživost, navigacija i interaktivnost). No ne preporučuje se kao primarni format e-knjiga zbog svoje umanjene pristupačnosti (posebno na manjim

zaslonima, poput pametnog telefona) izravno povezane sa sadržajem koji nije tečan. Učitavanjem PDF dokumenta na manjim zaslonima stranica se prikazuje tako da se skalira prema veličini zaslona, no budući da je većina PDF dokumenata pripremljena u veličini A4 ili sl., takav je prikaz premalen za čitanje. Korisnik je tada prisiljen ručno povećati stranicu te navigirati po njoj prstima, što umara i umanjuje sposobnost razumijevanja sadržaja.

PODRŽANOST (engl. *support*) (alati za čitanje) i OTVORENOST (engl. *open*) (povezanost, konverzija i arhiviranje)

Formati e-knjiga mogu biti otvoreni i zatvoreni. Otvoreni formati nisu osmišljeni za neki konkretni uređaj te je njihova specifikacija u potpunosti dostupna, a u slučaju e-knjiga često se osnivaju na nekom postojećem, također otvorenom, mrežnom standardu (poput jezika HTML, XML i CSS). Prednost je otvorenih formata njihova nezavisnost i dostupnost. U ovu kategoriju spadaju OEBPS, EPUB (2 i 3) te donekle i PDF. Zatvoreni ili vlasnički (engl. *proprietary*) formati specifično su kreirani za određeni uređaj ili aplikaciju izvan koje im se ne može pristupiti. Njihova je specifikacija dostupna u onoj mjeri u kojoj to proizvođač dozvoli, a napredak svojstava ovisi o dotičnom proizvođaču. Tako je Mobipocket razvijen za čitač Mobipocket, formati AZW i KF8 za uređaje Kindle i aplikacije za čitanje Kindle, a iBooks za aplikaciju iBooks na Appleovim uređajima.

Podržanost formata ovisi o njihovoj otvorenosti. Otvoreni su formati u konstantnom razvoju zbog svoje ovisnosti o drugim otvorenim mrežnim standardima, pa je tako i moguć razvitak alata za čitanje. Glavni razlog razvoja e-čitača Readium jest da potakne i druge na razvitak novih e-čitača za što više uređaja, što je evidentno iz njihova postojećeg i rastućeg broja. PDF, iako ga je Adobe inicijalno kreirao kao zatvorenog, postao je otvoren 2008. godine kada je specifikacija službeno puštena u javnost. PDF je kreiran da bude neovisan o uređajima i platformama te do danas takav i ostaje.

Međutim, zbog toga što otvoreni formati nisu osmišljeni za neki specifični uređaj, podržanost određenih dijelova specifikacije nije zagarantirana (za razliku od zatvorenih formata). Iako je specifikacija EPUB 3 izašla 2011. godine, još uvijek znatan dio uopće nije podržan ni na jednom e-čitaču, dok su neki dijelovi djelomice ili u potpunosti podržani. Zbog toga je IDPF osmislio mrežnu stranicu epubtest.org na kojoj tko god želi,

prema standardiziranim testovima, prijavljuje koji je dio specifikacije podržan na kojem e-čitaču. Na taj se način uvelike pomaže razvojnim programerima e-knjiga (ali i e-čitača) prilikom oblikovanja e-knjiga. Podržanost kod zatvorenih formata nije upitna jer su ti formati osmišljeni za taj dotičan uređaj. Problem je isključivo kod pristupa kupljenim e-knjigama jer su „zaključane” u taj uređaj ili platformu. Npr. e-knjigu iBooks nije moguće čitati na uređaju koji nije Apple.

Sam pojam otvorenosti nije povezan s pojmom DRM-a (Digital Rights Management). DRM je kodna zaštita koja onemogućuje kopiranje i arhiviranje dokumenata osmišljena zbog zaštite od piratstva. Iako je EPUB otvoreni format, a KF8 zatvoreni, na oba je moguće ugraditi DRM zaštitu, samo što je DRM obvezno uključen na svim Amazonovim i Appleovim e-knjigama, dok je na EPUB-u opcionalan. Zbog prisutnosti DRM sheme e-knjige kupljene putem Amazonove knjižare nije moguće dugoročno pohraniti, već iznajmiti. E-knjige bez DRM zaštite moguće je pohraniti, dijeliti i kopirati te čitati na neograničenom broju uređaja koji taj format podržavaju.

Konvertiranje zatvorenih formata, ovisno o zemlji, nije moguće na legalan način ako e-knjigu sami niste izradili, jer kako bi se e-knjiga konvertirala, obvezno je uklanjanje DRM sheme. Ako ste e-knjigu sami izradili, postoji mogućnost konverzije kroz određene softverske alate poput Calibre, ali su rezultati često nepredvidljivi u smislu prijeloma (Mobipocket, AZW, KF8). iBooks Author može importirati EPUB, inDesign, PDF, Word, Pages i Text, ali eksportirati samo u formate iBooks i PDF. Format EPUB moguće je konvertirati u bilo koji drugi format, ali obrnuti proces nije uvijek moguć, zadovoljavajući ili legalan.

EPUB 3, KF8 i iBooks dozvoljavaju definiranje standardnog fonta, veličine fonta, znakova, udaljenosti riječi i redaka putem CSS-a. Ove stilove korisnik uvijek može pregaziti na svojem uređaju. EPUB 3 dozvoljava kontrolu prijeloma (margina) putem CSS-a, dok je KF8 ne dozvoljava [27].

#### IZRADA (vrijeme, alati, znanje)

Proces izrade e-knjige ovisi o željenom formatu koji se često odabire prema sadržaju same e-knjige, ali i ciljane publike i tržišta kome je namijenjena. Amazonovi se formati kreiraju isključivo putem njihovih vlastitih alata koji zahtijevaju učenje prilikom

zahtjevnijih oblika sadržaja i prijeloma. Prednost je postojanje dodatka (*plug-in*) za InDesign koji je standardni alat struke za prijelom te mogućnost kreacije iz HTML dokumenata. iBooks je vjerojatno najjednostavniji za kreaciju te ima najbržu krivulju učenja zahvaljujući Appleovu softveru iBooks Author koji je pristupačan i jednostavan za upotrebu te nema iznenađenja u izlaznom formatu. PDF je prisutan već toliko godina u struci prijeloma pa nema govora o problemima postojanja potrebnih alata ili znanja. Najveći problem kvalitetne izrade semantičke e-knjige ima format EPUB. Zbog mnoštva dijelova specifikacije koja još nije podržana, razvojni programeri prisiljeni su osmišljavati načine za nove tipove sadržaja. Većina standardiziranih sadržaja koji su prisutni na internetu ne predstavljaju problem, ali do sada u industriji izdavaštva nisu bili potrebni razvojni programeri, već grafički dizajneri. Kreiranje EPUB-a ima još uvijek prepreke u pristupačnim alatima (iako InDesign vrlo brzo napreduje), ali i potpuno novom tipu posla koji podrazumijeva i kodiranje i grafički dizajn prijeloma. Međutim, jednako kako nijedan alat nije osmišljen za kvalitetno kreiranje mrežnih stranica (iako je Adobe Dreamweaver pokušao) te se one i danas kreiraju „ručno“ u uređivačima teksta, zasada je jednaka sudbina EPUB e-knjiga. InDesign može uvelike ubrzati proces, posebice dijelova poput metapodataka i navigacije, no sam sadržaj mora se naknadno ispravljati kako bi bio semantički pravilan. Zbog toga je najbrži i najkorektniji način „ručno“ oblikovanje u uređivaču teksta.

Najčešći način kreiranja e-knjige do danas je ipak konverzija iz PDF-a u Mobi ili EPUB. Takav je proces kompliciran i zahtijeva mnogo više vremena nego da se e-knjiga kreira od početka, zbog mnogo nepredvidivih grešaka koje se događaju, poput neobjašnjivih prijeloma stranica ili nestanka djela teksta [28].

#### MATEMATIČKI SADRŽAJ (prijelom i ažuriranje)

Matematički se sadržaj u e-knjige danas može ugraditi u obliku slike, jezika SVG ili označnog jezika MathML. Označni jezik MathML, semantički optimalan za oblikovanje pristupačnoga matematičkog sadržaja, podržan je u formatima EPUB 3 i iBooks.

Oblikovanjem matematičkog sadržaja MathML-om također se omogućuje ažuriranje te vrste sadržaja, dok je takav sadržaj u obliku slike zauvijek zarobljen u *bitmap* obliku.

Opcije oblikovanja najsigurnije su i optimalne u obliku slike (dakle Mobipocket, AZW i

KF8) i PDF-u, ali tako se gubi opcija tečnosti tog sadržaja. Optimalan matematički slog u e-knjigama podrazumijeva da je oblikovan označnim jezikom koji je dostupan pomoćnim tehnologijama, koji se aktivno i dinamično prilagođava veličini ekrana tako da sadržaj ostaje čitljiv i razumljiv bez obzira na to koji se e-čitač upotrebljava. Oblikovanje i prijelom, u teoriji, u potpunosti su definirani jezikom MathML, a u nastavku ovog rada vidjet će se optimalni načini za postizanje tih mogućnosti.

Zaključuje se da je za izradu optimalnoga dinamičkog matematičkog sloga najbolji format EPUB 3 jer je otvoren (nevezan isključivo za jedan tip e-čitača), omogućuje prilagođavanje matematičkog sadržaja različitim rezolucijama i veličinama e-čitača te omogućuje pristup pomoćnim tehnologijama.

Tablica 1. Usporedba formata e-knjiga prema definiranim parametrima za optimizaciju dinamičkoga matematičkog sloga

<b>format</b>	<b>tečnost</b>	<b>podržanost</b>	<b>otvorenost</b>	<b>izrada</b>	<b>DRM</b>	<b>pristupačnost</b>	<b>matematika</b>
Mobipocket	da	ne	da	da	opcija	da	ne
AZW	da	ne	ne	da	da	da	ne
KF8	da	ne	ne	da	da	ne	ne
EPUB 2	da	da	da	ne	opcija	da	ne
EPUB 3	da	da	da	ne	opcija	da	da
IBA	da	ne	ne	da	da	da	da
PDF	ne	da	da	da	opcija	parcijalno	ne

## 6. KREIRANJE MODELA E-KNJIGE STANDARDOM EPUB 3

Specifikacija EPUB standardni je format za distribuciju i razmjenu digitalnih publikacija i dokumenata. EPUB definira način prikaza, pakiranja i kodiranja strukturiranog i semantički poboljšanog mrežnog sadržaja (uključujući HTML5, CSS, SVG, slike i multimediju) za distribuciju u obliku jedne datoteke. EPUB 3 treća je verzija standarda koja se sastoji od četiri specifikacije:

1. EPUB Publications 3.0 koja definira semantiku i sveobuhvatnu sukladnost za EPUB publikacije
2. EPUB Content Documents 3.0 koja definira profile XHTML, SCG i CSS-a za primjenu u EPUB publikacijama
3. EPUB Open Container Format koja definira format datoteke i model obrade za pakiranje svih datoteka u jednu (ZIP)
4. EPUB Media Overlays 3.0 koja definira format i model obrade za sinkronizaciju audiozapisa i teksta.

Trenutačna verzija standarda jest EPUB 3.0.1 koja je na snazi od lipnja 2014. godine. Specifikaciju je kreirala skupina IDPF (International Digital Publishing Forum) EPUB Working Group čiji je rad u potpunosti otvorenog tipa (eng. *open source*). EPUB je općeprihvaćen kao format za e-knjige, a osmišljen je da podržava sve vrste izdavačkih potreba poput kompleksnih prijeloma, multimedije i interaktivnosti te globalne tipografije. Namjena mu nadilazi isključivo e-knjige, te uključuje i časopise, publikacije za edukaciju te stručne i znanstvene publikacije [33].

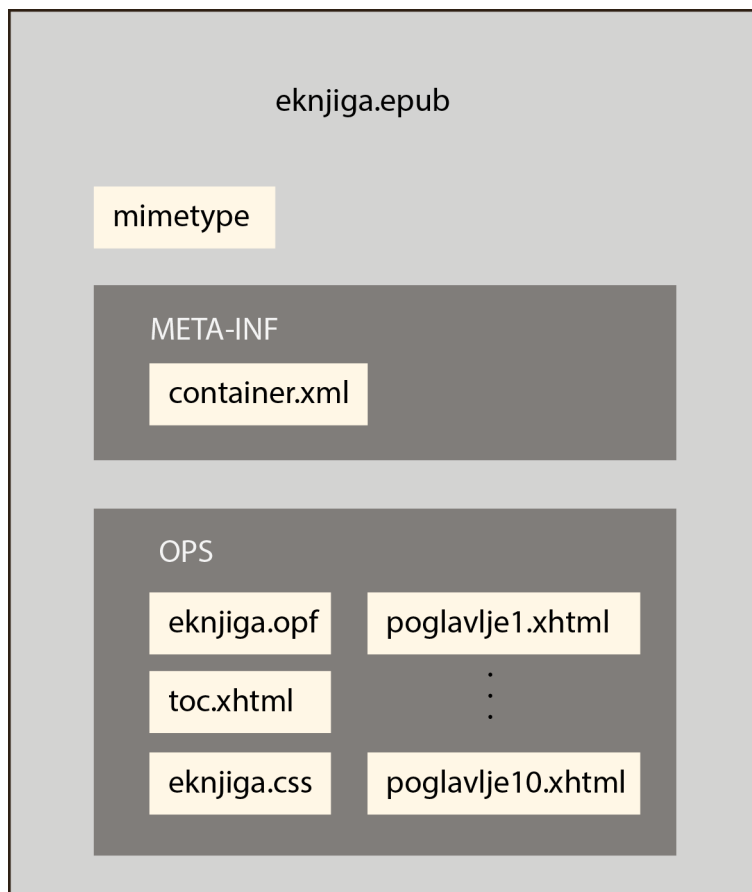
Proces standardizacije započeo je 1998. godine i cilj mu je bio razviti otvoreni standard e-knjige koji se osniva na XML-u te je neovisan o platformi na kojoj se čita. Platforme su postale relevantne nakon pojave uređaja manjih zaslona na kojima se PDF nije mogao čitati kao na desktop računalima. EPUB 2 bio je oblikovan većinom za tekstualne knjige s varijabilnom širinom koja omogućuje sadržaju da se aktivno prilagođava širini zaslona te time pruža opciju čitanja na uređajima različitih veličina. EPUB 3 nastao je kako bi ispravio nedostatke standarda EPUB 2, a sada se osniva na tehnologijama HTML5 i CSS3 dozvoljavajući kreaciju multimedije, kompleksnih prijeloma, matematičkih formula i interaktivnosti, fiksnih prijeloma, nelinearnih dokumenata i funkcionalnosti *text-to-*

*speech*. EPUB se osniva na tehnologiji HTML5 odnosno XHTML5. XHTML5 jest XML inačica specifikacije HTML5, primjenjuje XML način izražavanja te olakšava automatizirano procesiranje dokumenata. Iako EPUB 3 dozvoljava upotrebu jezika JavaScript za skriptirani sadržaj, ipak nije standardiziran, što rezultira različitim razinama podrške na e-čitačima. Do velike je promjene također došlo u oblikovanju navigacijskog dokumenta, koji je u standardu EPUB 2 bio XML dokument ekstenzije .ncx, kompliciran za kreiranje, ali i razumijevanje. EPUB 3 uvodi HTML5 element `<nav>`, jednostavniji za čitanje i interpretaciju [27].

Prvi je korak pri izradi EPUB 3 e-knjige kreacija dokumenata sadržaja koji moraju biti XHTML5 ili SVG dokumenti. Zatim slijedi kreiranje dokumenta *package* kojim se koriste e-čitači kako bi sakupili razne informacije o e-knjizi. U dijelu *manifest* u obliku popisa navode se svi izvori upotrijebljeni prilikom oblikovanja dokumenata sadržaja, uključujući dokumente stila, fontove, multimediju, slike, skripte i XHTML dokumente sadržaja. E-čitači ovim se popisom koriste kako bi utvrdili je li e-knjiga potpuna te koje sve vanjske resurse mora dobiti kako bi prikazali sadržaj. U dokumentu *package* također se nalaze metapodaci koji uključuju informacije o knjizi, poput autora, imena knjige, vremena izdavanja itd. Obavezan je dio i sekcija *spine* koja u obliku popisa navodi standardni slijed čitanja svih zasebnih dokumenata sadržaja. Posljednji je korak prilikom oblikovanja pakiranje svih dokumenata u jednu ZIP arhivu. Posebni dokumenti *mimetype* i *container.xml* moraju biti uključeni u paket kako bi se paket identificirao kao EPUB te kako bi e-čitač znao pronaći dokument *package*. Ovaj se manualni proces tipično ne izvodi svaki put u cijelosti, već postoje softverski alati za sveukupnu ili djelomičnu automatizaciju procesa, ovisno o sadržaju e-knjige.

EPUB djelomično svoje mogućnosti temelji na mrežnim tehnologijama kao što su XHTML5 i SVG 1.1 za tekst, slike, multimediju, matematički sadržaj i grafike; CSS 2.1 i 3 za stilsko oblikovanje i prikaz sadržaja; JavaScript za interaktivnost i automatizaciju skriptiranjem; TrueType i WOFF za primjenu različitih fontova; specijalizirane XML gramatike za jednostavno otkrivanje i procesiranje EPUB-a; SSML/PLS/CSS3 *text-to-speech* za poboljšano glasovno iščitavanje.

Ostale je tehnologije koje se koriste u EPUB knjigama specijalno razvio IDPF [34].



Slika 2. Slika strukture EPUB 3 e-knjige: datoteka *mimetype* sadrži tekst kojim e-čitač prepoznaje EPUB; direktorij META-INF sadrži datoteku *container.xml* koja ukazuje na dokument *package*; direktorij OPS (Open Publication Structure) sadrži sve dokumente sadržaja, dokument *package*, metapodatke, izvore publikacije i njihove lokacije, CSS dokument te tablicu sadržaja (*toc.xhtml*).

Dokument *mimetype* jednostavan je ASCII tekstualni dokument koji operativnom sustavu e-čitača govori na koji je način e-knjiga formatirana. U EPUB e-knjizi uvijek mora imati identičan sadržaj: `application/epub+zip`.

Direktorij META-INF može sadržavati jedan ili više XML dokumenata. *Container.xml* obavezan je dokument koji ukazuje na dokument *package* odnosno na sadržaj e-knjige.

Opcionalni dokumenti mogu sadržavati informacije o enkripciji, DRM-u i sl.

Prilikom ručnog kreiranja EPUB e-knjige važno je da je dokument *mimetype* prvi unutar ZIP arhive. Standardno pakiranje ZIP-om, međutim, postavlja direktorij META-INF na prvo mjesto, a na taj način e-knjiga ne prolazi validaciju. Zbog toga se primjenjuju alati



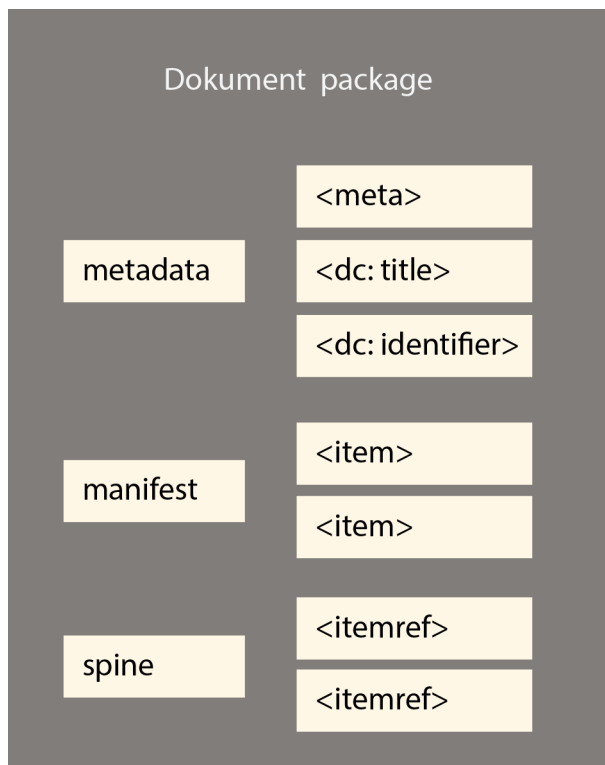
za kreiranje ZIP arhive, na Mac računalima se upotrebljava aplikacija ePub Zip/Unzip, a na računalima s operativnim sustavom Windows ePubPack.

## 6.1. Dokument *package*

Sve temeljne informacije o e-knjizi potrebne e-čitačima za procesuiranje nalaze se u XML dokumentu *package*. Dokument započinje elementom `<package>` s djecom elementima `<metadata>`, `<manifest>` i `<spine>` u sljedećoj strukturi:

```
<package ... version="3.0"  
xmlns="http://www.idpf.org/2007/opf">  
  <metadata xmlns:dc="http://purl.org/dc/elements/1.1/">  
    ...  
  </metadata>  
  <manifest>  
    ...  
  </manifest>  
  <spine>  
    ...  
  </spine>  
</package>
```

Element `<package>` mora imati deklariranu verziju i *namespace*, a može imati i jezik u kojem je pisan i smjer teksta.



Slika 3. Prikaz strukture dokumenta *package*

## 6.2. Metapodaci

EPUB 3 koristi se vokabularom Dublin Core Metadata Element Set (DCMES) za većinu obveznih i opcionalnih metapodataka. Metapodaci pružaju informacije o e-knjizi i o sadržaju knjige krajnjim korisnicima, ali i e-čitačima kako bi olakšali procesuiranje EPUB-a. Zbog potrebe za bogatstvom i raznolikošću metainformacija koje omogućuju kompleksnije funkcionalnosti, specifikacija EPUB 3 dozvoljava i druge vokabulare osim onih osnovnih. Osnovni vokabular uključuje ključne elemente: `<meta>`, `<link>`, `<item>` (metapodaci za elemente u *manifestu*) i `<itemref>` (metapodaci za elemente u *spineu*). Ovi osnovni izrazi pružaju e-čitačima siguran i konzistentan način razumijevanja procesuiranja svakog elementa. Element `<item>` u manifestu, na primjer, informira e-čitač o tome koji dokumenti sadržaja uključuju MathML i/ili JavaScript.

Pogonski metaelement specifikacije EPUB 3 jest `<meta>` koji pruža jednostavan, generički i fleksibilan mehanizam za povezivanje metapodataka s *packageom* i njegovim sadržajem. Broj metaelemenata nije ograničen, a nalaze se u elementu `<metadata>`, prvom djetetu elementa `<package>`. Element `<metadata>` mora obvezno sadržavati tri elementa:

`<dc:identifier>`, `<dc:title>` i `<dc:language>`. Pritom se `dc` odnosi na vokabular DCMES koji mora biti deklariran unutar elementa `<metadata>`:

```
<metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
```

Element `<dc:identifier>` sadrži identifikacijsku oznaku e-knjige (npr. ISBN ili DOI). EPUB može imati i više ovih oznaka, ali samo jedna mora biti i označena atributom `unique-identifier`. Element `<dc:title>` sadrži naslov e-knjige, a `dc:language` specificira jezik kojim je sadržaj e-knjige napisan. Postoje i mnogi drugi korisni, ali opcionalni elementi metapodataka koji upućuju na informacije o kreatoru e-knjige, autoru, datumu kreacije itd. Specifikacija EPUB 3 također nalaže da *package* sadržava metaelement koji bilježi vrijeme i datum izrade. Osnovni element `<metadata>` EPUB e-knjige može se vidjeti na slici 4.

```
<metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
  <dc:title id="title">Moja prva matematika</dc:title>
  <dc:identifier id="mojamat">2014-11-10T09:38:00Z</dc:identifier>
  <dc:language>hr</dc:language>
  <dc:creator>Maja Turcic</dc:creator>
  <meta property="dcterms:modified">2014-11-10T09:38:00Z</meta>
</metadata>
```

Slika 4. Osnovni izgled elementa <metadata> i njegove obvezne djece u EPUB e-knjizi

### 6.3. Elementi za popis izvora publikacije i slijed čitanja – *manifest* i *spine*

Nakon elementa `<metadata>`, sljedeći obvezni dijelovi elementa *package* jesu *manifest* i *spine*. *Manifest* sadrži popis, obilježen elementima `<item>`, svih izvora publikacije, poput dokumenata sadržaja, CSS dokumenta, videozapisa i zvučnih zapisa, fontova i navigacijskog dokumenta. Svrha je ovog popisa da unaprijed informira e-čitač o tipu medija svakog elementa i o tome gdje se nalazi unutar e-knjige. Redoslijed ovog popisa nije važan, ali je važno da je naveden svaki izvor publikacije. Svaki `<item>` ima tri obvezna atributa: `id`, `href` i `media-type`. Jedan `<item>` za dokument sadržaja koji sadrži MathML i skriptirani sadržaj izgleda ovako:

```
<item id="p2" media-type="application/xhtml+xml"
href="poglavlje2.xhtml" properties="mathml scripted">
```

Atribut `id` služi jedinstvenom identificiranju svakog pojedinog elementa koji omogućuje uvrštavanje metapodataka. Atribut `href` definira lokaciju dokumenta, a `media-type` definira tip i format izvora. Neobvezatan atribut jest `properties` u koji se upisuje lista vrijednosti bez zareza koje olakšavaju procesuiranje e-čitaču. Prilikom navođenja dokumenta koji sadrži matematiku i skriptu korisno je navesti vrijednosti `mathml` i `scripted` kako bi e-čitač znao da će se susresti s tim tipovima sadržaja.

*Spine* navodi listu komponenata knjige označene elementima `<itemref>` koji pružaju informaciju standardnog slijeda čitanja e-knjige. Navodi se samo primarni sadržaj koji predstavlja barem jedan logičan slijed čitanja:

```
<spine>
  <itemref idref="p1"/>
  <itemref idref="p2"/>
  ...
</spine>
```

#### 6.4. Navigacijski dokument

Jedna od glavnih prednosti elektroničkih publikacija jest olakšano i brzo navigiranje kroz sadržaj. Dok se EPUB 2 koristio kompleksnim NCX dokumentom, EPUB 3 uvodi novi i razumljiviji EPUB navigacijski dokument baziran na označnom jeziku HTML5. EPUB navigacijski dokument omogućava e-čitačima da čitateljima pruže prikaz specijalizirane tablice sadržaja, ali i kao sadržajni prikaz unutar e-knjige. Specijalizirani prikaz tipično je dostupan u obliku posebne ikone na vrhu ekrana koja pritiskom aktivira prikaz tablice sadržaja u obliku bočne trake ili lebdećeg prikaza koji prekriva trenutačnu stranicu. Ovakav je prikaz omogućen posebnim oznakama u navigacijskom dokumentu koji e-čitači upotrebljavaju za renderiranje, a njegov dizajn ovisi o razvojnim programerima koji su ga oblikovali za pojedini sustav. Ovaj dokument, iako razumljiviji i lakši za kreiranje od starog formata NCX, nije zamišljen da se oblikuje „ručno”. Svi programi kojima se može kreirati EPUB 3 dokument automatski će ga generirati prema sadržaju e-knjige. Strukturu je potrebno poznavati ako autor e-knjige želi kontrolirati prikaz. Navigacijski dokument (slika 5) jest XHTML dokument koji sadrži željeni broj elemenata `<nav>` unutar kojih se gnijezde uređene liste (`<ol>`). Tip navigacijskog dokumenta određuje se atributom `epub:type`, s obzirom na to da postoje tri različita tipa koji se mogu primijeniti. Zbog ograničenja HTML gramatike prilikom definiranja bogatih struktura podataka, posebice prilikom uzimanja u obzir nevizualnih čitatelja kojima je takva struktura važna zbog pristupačnosti, EPUB 3 uključuje atribut `epub:type` koji omogućuje precizno značenje generičkim oznakama koje se zovu semantička modulacija. Bogata semantika podataka je ključna prilikom navigacije, ali i specijaliziranih ponašanja e-čitača. Atribut `epub:type` u slučaju navigacijskog dokumenta definira tip navigacije (npr. `toc` – *table of contents*), ali i mjesto na kojem se taj tip podataka tipično pojavljuje u knjizi (eng. *front matter*) [35] [36].

```
<html xmlns="http://www.w3.org/1999/xhtml" xmlns:epub="http://www.idpf.org/2007/ops">
  <head>
    <title> Moja prva matematika </title>
    <meta charset="utf-8"/>
  </head>
  <body epub:type="frontmatter">
    <nav epub:type="toc" id="toc">
      <h1> Tablica sadržaja </h1>
      <ol>
        <li>
          <a href="poglavlje1.xhtml"> poglavlje 1 </a>
        </li>
        ...
      </ol>
    </nav>
  </body>
</html>
```

Slika 5. Prikaz strukture navigacijskog dokumenta u EPUB 3 e-knjizi



## 6.5. Validacija

Validacija je proces provjere valjanosti kreirane publikacije prema standardima IDPF-a. Prolaženje ovog procesa osigurava potvrdu kreatoru da je ispravno oblikovao EPUB datoteku te da će se sadržaj otvoriti i prikazati na bilo kojem e-čitaču. Validator također upućuje na konkretne pogreške unutar strukture EPUB e-knjige koje mogu biti rezultat loše strukture oznaka. Validator prati pravila EPUB standarda, ali ne osigurava da ta ista pravila i apliciraju svi e-čitači. Stoga je potrebno, posebice pri korištenju neafirmiranim tehnologijama i jezicima poput MathML-a, testiranje na samim uređajima. Alat koji se primjenjuje pri validiranju EPUB 3 datoteka jest EpubCheck. EpubCheck postoji od vremena standarda EPUB 2 te ga je razvio Adobe, ali je njegovo održavanje u obliku *open source* alata preuzeo IDPF. Slobodan je za upotrebu i eventualnu adaptaciju prema potrebi. Alat postoji u dva oblika: kao *online* aplikacija na internetskoj adresi <http://validator.idpf.org/> [37] ili u obliku instalacijskog paketa koji funkcionira lokalno na osobnom računalu putem komandne linije te zahtijeva instaliran paket Java. *Online* validacija ima manu postavljenog limita veličine e-knjige na 10 MB, što može predstavljati problem prilikom kreiranja e-knjiga bogatog sadržaja poput multimedije, velikog broja fontova ili ugrađene JavaScript knjižnice. Lokalno instalirani validator provjerava strukturu i oznake datoteke *package*, konzistenciju referenciranih izvora i drugih grešaka, ali i može generirati izvješće o procjeni koje sadrži greške, upozorenja, ali i razne metapodatke o EPUB-u. Validacija je od koristi i prilikom kreiranja EPUB-a „ručno”, ali i nakon korištenja softverskim alatima koji često kreiraju semantički i označno nekorektne strukture. Prilikom kreiranja matematičkog sadržaja jezikom MathML česta je greška nenavođenje potrebnog i valjanog *namespacea*. S obzirom na to da je EpubCheck strukturalni validator, ne može provjeriti valjanost skriptiranog sadržaja te je JavaScript kod obvezno provjeriti u što je više moguće e-čitača [34].

## 6.6. Analiza softverskih alata za kreiranje EPUB e-knjiga

Najpopularniji te najdostupniji softverski alati uključuju Adobe InDesign, Sigil i iBooks Author.

Sigil je *open source* i besplatan softver za kreiranje e-knjiga u EPUB formatu. Primjenjuje se za oblikovanje, pakiranje i validiranje EPUB e-knjiga. Ima mogućnosti unošenja velikog opsega metapodataka putem jednostavnog sučelja, uređivanja, kreiranja i validiranja pojedinih HTML i CSS datoteka putem koda i generiranja tablice sadržaja. Najveća je prednost Sigila otvaranje i uređivanje postojećih e-knjiga, ubacivanje gotovih HTML datoteka te prikaz koda. Jedina je mana što Sigil nema ugrađenu mogućnost kreiranja ni uređivanja e-knjiga prema standardu EPUB 3, već isključivo prema zastarjelom standardu EPUB 2. Međutim, postoji i dodatak koji omogućuje ponovno procesiranje u format EPUB 3. Rezultat nije optimalan te je potrebno uređivanje kako bi dokument prošao validaciju, no budući da je alat *open-source*, konstantno je prisutno poboljšavanje unutar zajednice. Umetnuti, prethodno kreirani MathML sadržaj Sigil gleda kao HTML kod te je u stanju provjeriti ispravnost oznaka. Sigil time čini kvalitetan alat za kreiranje osnovne ljuske EPUB e-knjige, ali ne kao direktan i pouzdan, posebice pri kreiranju novih EPUB 3 mogućnosti.

iBooks Author besplatan je WYSIWYG (*what you see is what you get*) alat za kreiranje e-knjiga tvrtke Apple. Ovim se alatom može uređivati postojeće EPUB e-knjige, kreirati nove, ali i uvesti već kreirane HTML, Word ili Pages datoteke. Ne dozvoljava ni pregled ni uređivanje na razini koda. Unos metapodataka ograničen je na autora, naslov i jezik. E-knjiga može se izvesti u EPUB 3, PDF ili Appleov format .iba. Kreiranje i uređivanje matematičkih formula moguće je povezivanjem s alatom MathType. Validacija e-knjige nije potrebna jer se dokumenti ne kreiraju „ručnim” kodiranjem. Tablica sadržaja kreira se i uređuje putem jednostavnog sučelja. iBooks Author omogućavanjem kreiranja EPUB 3 datoteka učinio je veliki pomak ka prihvaćanju ovog formata e-knjiga i povećao pristupačnost kreiranja e-knjiga osobama koje nisu vične kodiranju. Nakon uvezene već kreirane EPUB 3 e-knjige ne postoji mogućnost izvoza u format EPUB, nego samo u .iba, PDF ili običan tekst. Međutim, nakon odabira EPUB predloška prilikom kreiranja nove knjige ili kreiranja matematičke formule povezanim alatom MathType ta mogućnost

postoji. Kreirana matematička formula ostaje u jeziku MathML, međutim s nekorektnom semantikom. Alatom iBooks Author može se WYSIWIG načinom kreirati EPUB 3 e-knjiga koja sadrži matematički sadržaj koji nije u obliku slike. Iako se kasnije kod mora doradivati radi označne i semantičke korektnosti, ovaj besplatni alat omogućuje kreiranje EPUB e-knjiga bez predznanja.

Adobe InDesign profesionalni je alat za prijelom koji također sadrži opciju kreiranja odnosno izvoza EPUB 3 e-knjiga. Sadrži opcije detaljnog uređivanja metapodataka te generiranje i oblikovanje tablice sadržaja prema standardu EPUB 3. Adobe InDesign alat je za grafičke dizajnere, stoga ne nudi opciju uređivanja koda ni validacije, ali nudi najviše najpreciznijih opcija dizajna od svih ponuđenih programa. Kao alat za kreiranje osnovne strukture EPUB 3 dokumenata koje mogu uključivati i multimedijske i interaktivne elemente, posebice dizajnerima koji su navikli na njegovo sučelje, može koristiti kao početna točka kreiranja e-knjige. Kreiranje matematičkih formula unutar sučelja InDesigna čak ni uz pomoć alata MathType nije moguće na način da rezultat bude označni jezik MathML. Jedini je način uvrštavanje formula u formatu EPS koje tvore poveznicu na originalan dokument pa je naknadno uređivanje matematičkog sadržaja i dalje moguće. Stoga nije moguće kreirati semantički i označno korektnu EPUB e-knjigu uz pomoć Adobe InDesigna.

Tablica 2. Usporedba popularnih softverskih alata za kreiranje EPUB e-knjiga

	Sigil	iBooks Author	Adobe InDesign
EPUB 3	ne	da	da
metapodaci	da	ne	da
kod i validacija	da	ne	ne
MathML	da	da	ne
besplatan	da	da	ne

Iz tablice 2 može se vidjeti da nijedan alat ne pruža optimalan način kreiranja EPUB 3 e-knjiga označno korektnoga matematičkog sadržaja. Svaki autor EPUB e-knjiga kreira vlastiti način rada prema navikama i preferencijama. iBooks Author učinio je važan korak ka popularizaciji formata EPUB 3 time što je besplatan i jednostavan za upotrebu.

Grafički dizajneri skloni su upotrebi Adobe InDesigna zbog navike, a razvojni programeri radije upotrebljavaju Sigil zbog mogućnosti uređivanja i kontroliranja koda. Neizbježno je da svaki autor poznaje strukturu formata EPUB 3, jer bez obzira na to kojim se alatom koristili, naknadna je dorada na razini koda za sada obvezna.

## 7. STANDARDI UGRAĐIVANJA MATEMATIČKIH FORMULA U E-KNJIGE

Matematički sadržaj oduvijek je predstavljao jedan od kompleksnijih vida prijeloma zbog posebnih tipografskih pravila koja uvjetuju razumijevanje tog tipa sadržaja, posebnih znakova koji se upotrebljavaju prilikom oblikovanja te kompleksnih oblika, veličina i pozicija pojedinih znakova koje matematika zahtijeva. Autori se posebno moraju posvetiti stilu, korištenju, značenju, jasnoći, točnosti i konzistenciji. Moraju se koristiti pravilnom terminologijom i označavanjem te moraju pažljivo pratiti dogovore specijaliziranih polja [38].

Prijelaz u digitalno okruženje sofisticiranih računalnih alata za prijelom olakšao je oblikovanje matematičkog sadržaja; kao na primjer sustav Latex za pripremu dokumenata koji je omogućio da se autori znanstvenih i tehničkih sadržaja ne moraju brinuti o samom izgledu koji je automatski zajamčen samim sustavom, već se brinu isključivo o sadržaju.

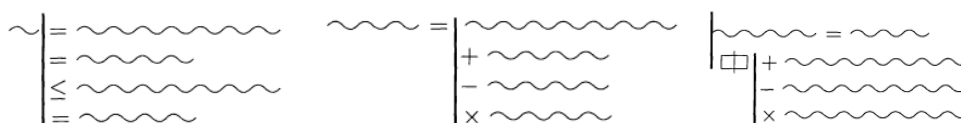
Pojavom interneta potreba za znanstvenim i tehničkim sadržajem ubrzo je naišla na prepreke. S obzirom na to da se standard EPUB 3 većinom oslanja na mrežne standarde, slični su se problemi pojavili prilikom uvrštavanja matematičkih formula u e-knjige. Za razliku od standarda EPUB 2 gdje nije postojala preporuka oko uvrštavanja tog tipa sadržaja u e-knjige zbog nepostojanja bilo tehnologije bilo podrške, u standardu EPUB 3 preporuka je za korištenje označnim jezikom MathML. Prijelaz između ova dva standarda nisu generalno prihvatili izdavači i prodavači te se trenutačno ovo područje i dalje nalazi u nedoumicama oko najboljeg načina rješavanja ovog problema.

Problem nadilazi isključivo prezentaciju informacija te uključuje i pristupačnost sadržaja osobama s invaliditetom, kopiranje i pretraživanje sadržaja te procesiranje sadržaja u svrhu softverske analize ili prebacivanja u alternativni format. Također, potrebno je uzeti u obzir i mogućnost prikaza posebnih matematičkih znakova i simbola za koje je potreban adekvatan font.

Osim posebnih znakova, matematika donosi i cijeli set tipografskih standarda koji se moraju poštivati zbog čitljivosti i razumljivosti sadržaja. *Chicago Manual of Style* navodi kako se matematički sadržaj mora prikazivati tipografskom kvalitetom jednakom tekstu kojim je okružen, posebno se odnosi na skaliranje (formule se moraju skalirati s

tekstom); formule smještene u istoj liniji s tekstom moraju imati identičnu liniju teksta (osnovnu liniju na kojoj „leže” sva slova). Ako je matematička formula predugačka da se prikaže u jednom retku, prijelom mora biti kod operatora (npr. = ili +) ili znaka za relaciju (npr. <), a idući redak mora biti uvučen te poravnat s logičnim mjestom u retku iznad [38]. Knjiga *Mathematics into Type* koju izdaje Američko udruženje za matematiku (American Mathematical Society) detaljnije navodi postupke u slučaju prikaza izraza kao samostalnih objekata (ne unutar tekućeg teksta):

1. izraz se mora lomiti prije operatora, ako nije u zagradama, da čitatelj zna da se izraz nastavlja u idućem retku, a ne da započinje novi izraz
2. poravnanje se radi kod znaka jednakosti ili, ako je jako uvučen, dva četverca od lijevog ruba prvog retka formule
3. ako se formula lomi kod znakova operacija (+, −, ×, /) tada se poravnava s prvim znakom nakon jednakosti (ili dva četverca) [39].



Slika 6. Prikaz tipografskih pravila kompleksnih matematičkih izraza

U slučaju nepodržanosti odabranog formata, sadržaj mora degradirati ili prikazati zamjenski sadržaj tako da bez obzira na odabrani e-čitač čitatelj ima pristup sadržaju. I naravno, sav sadržaj mora biti vjeran specifikaciji formata EPUB 3.

Problem mogu predstavljati i softverski alati dostupni za oblikovanje željenog sadržaja te kontrola samog oblikovanja uz pomoć tih alata. Kompatibilnost je također faktor koji utječe na dobro korisničko iskustvo, a ovisi o odabranoj formi sadržaja. Način oblikovanja utjecat će i na veličinu e-knjige, a količina vremena i kompleksnosti izrade na cijenu finalnog proizvoda.

Sam standard nudi nekoliko različitih mogućnosti pri kreiranju e-knjiga s matematičkim formulama. Prvi način, na koji se većina e-knjiga danas oslanja, u obliku je slika. Slike mogu biti u *bitmap* formatu (JPEG, PNG, GIF) ili vektorskom formatu (SVG). Drugi je način, koji preporučuje IDPF, korištenje označnim jezikom MathML.

## 7.1. Ugrađivanje matematičkog sloga u obliku slika u e-knjige

Uključivanje kompleksnoga matematičkog sadržaja u e-knjige najčešće se izvodi u obliku slika. Slike u *bitmap* formatu podržane su na svim uređajima i u svim formatima e-knjiga. Kreiranje takvog sadržaja ne primorava autore na nove procese rada, već im je omogućeno da postojeći format konvertiraju u sliku. Pa tako autori koji su navikli raditi u okruženju LaTeX, koje je najčešće za izradu znanstvenih publikacija, imaju mogućnost korištenja raznim *online* konverterima, *perl* skriptama (npr. *gladTeX*) koje konvertiraju u slike, ali i mogućnost kreiranja PDF dokumenta (*pdflatex*) iz kojeg je jednostavno kreirati *bitmap* slike. Nerealno je očekivati proces izrade gdje se najprije kreira LaTeX, zatim PDF te na kraju slika koja se ugrađuje u XHTML. Jednostavniji je i izravniji proces upotreba popularnog sustava TeX4ht koji direktno konvertira LaTeX dokument u HTML ili XHTML dokument te dopušta odabir formata slika, bilo *bitmap* bilo SVG, za matematičke formule koje se ne mogu prikazati u formatu HTML.

Za kreiranje SVG slika matematičkog sadržaja, osim konvertera iz formata LaTeX, primjenjuju se standardni softverski alati poput Adobe Illustratora i Inkscapea. Adobe Illustrator, međutim, nije zamišljen kao alat za slaganje matematičkog sadržaja pa je prilično nespretn te je proces dugotrajan. Upotrebom alata Math Type kojem to i jest namjena moguće je kao izlazni format postaviti EPS koji zadrži sva svojstva editiranja teksta jednom kada se unese u Illustrator. Inkscape ima opciju unošenja formula formatu u LaTeX, što ga čini pristupačnijim za kreiranje SVG slika jer je moguće sav posao obaviti u jedinstvenom okruženju.

Najveća je prednost kreiranja matematičkog sadržaja u obliku slika potpuna autorska kontrola. Autor određuje svaki aspekt oblikovanja matematičkih formula sa sigurnošću prikaza na svim uređajima i platformama. Problemi prikaza koji se javljaju nisu u podržanosti samog formata slika, već u načinu renderiranja. Sadržaj u obliku slika skalira se i pozicionira drugačije na različitim uređajima. Određena CSS svojstva tretiraju se drugačije pa se CSS kod i HTML elementi moraju podešavati na specifične nestandardne načine kako bi prikaz bio svugdje jednak. Na primjer, stiliziranje slika za e-čitač iBooks zahtijeva omatanje elementa `<img>` elementom `<div>` kako bi se veličina slike mogla regulirati s postotnim (%) vrijednostima. Upotreba elementa `<div>` u svrhu

prezentacije smatra se semantički netočnom. Prilikom ugrađivanja matematičkog sadržaja koji je okružen tekstem (*inline*) dolazi i do problema pomicanja osnovne linije na kojoj izraz „leži”, što se smatra tipografskom greškom koja otežava čitkost. Rezolucija također može uzrokovati poteškoće s čitkošću, jer ako nije dovoljno velika, slika će imati pikseliziran prikaz, a ako je rezolucija velika, a takvog sadržaja ima mnogo, e-knjiga će biti pretjerano velika. Sadržaj u obliku *bitmap* slika također se veličinom ne prilagođava veličini fonta. Korisnici, naime, imaju mogućnost mijenjanja veličine fonta unutar e-čitača, što predstavlja jedno od osnovnih svojstva pristupačnosti, ali ako je sadržaj „zarobljen” u sliku, neće se skalirati u skladu s fontom. Kada se formule oblikuju kao zasebni blokovi (eng. *display*), u slučaju kompleksnijih izraza te izraza koji su predugački za jedan redak, ako se čita na malom uređaju ili uređaju visoke rezolucije, jedini način pristupa takvom sadržaju jest da korisnik lagano udari (eng. *tap*) na mjesto gdje je slika te tehnikom „*zoom & pan*” (povećavanje s oba prsta te pomicanje slike prstom po ekranu) čita sadržaj. Takav je postupak čitanja zamoran i nečitljiv.

U slučaju vektorskih SVG slika prilagođavanje sadržaja znatno je bolje jer se SVG renderira na drugačiji način. Moguće je skaliranje u skladu s okolnim tekстом, a zahvaljujući SVG svojstvima moguća je i bolja kontrola pozicioniranja u odnosu na osnovnu liniju. SVG je također zbog svoje vektorske prirode mnogo kvalitetniji prilikom povećavanja pa nema nikakvog gubitka informacija te su znakovi uvijek oštri. Međutim, ni u slučaju SVG slika nema govora o prilagođavanju sadržaja prema veličini ekrana. E-čitač će SVG doživljavati kao blok te se formula ne može lomiti prema potrebi. Za razliku od *bitmap* slika, SVG u svojoj specifikaciji ima i opciju ugrađivanja i oblikovanja teksta s elementom `<text>`, što u tom slučaju ne „zaključa” tekst u sliku, već će se i dalje moći selektirati kao tekst i pretraživati. Ipak, ručno kreiranje takvog koda bilo bi nerealno zbog svoje kompleksnosti. Najrealniji tijek bila bi konverzija iz jezika MathML ili Latex, za što postoje razni alati (npr. Python skripta SVG Kit), ali na taj način nema kontrole zadržavanja teksta u obliku slovnih znakova za selekciju ili pretraživanje, a nikako za konverziju u drugi matematički format.

Podržanost SVG-a u e-čitačima prema specifikaciji EPUB 3 obvezna je te je znatno porasla pa se može primjenjivati bez pretjerane zabrinutosti hoće li čitatelj imati pristupa sadržaju.



## 7.2. Usporedba pristupačnosti matematičkog sloga u obliku slika u e-knjigama

Uporabljivost (eng. *usability*) je najočitiji problem pri uvrštavanju matematičkog sadržaja u e-knjige u obliku slika jer ne omogućava maksimalnu efikasnost, učinkovitost ni satisfakciju pri postizanju konkretnog cilja čitatelja, odnosno čitanja matematičkog sadržaja u e-knjigama [40]. Međutim, pristupačnost (eng. *accessibility*) slika općenito, kako na internetu tako i u e-knjigama, predstavlja velik problem, posebice kada slike imaju sadržaj krucijalan za narativ, što je i slučaj s matematičkim formulama. Pojam pristupačnosti često se povezuje isključivo s određenim vrstama invaliditeta, a u slučaju slika očito iskače problem pristupačnosti slijepim i slabovidnim osobama. Međutim, jednak problem imaju i ljudi sa situacijskim nemoćima u trenucima kada su se konstantno prisiljeni koristiti mehanizmima povećavanja zbog malog zaslona ili pokušavaju čitati na izravnoj Sunčevoj svjetlosti [36].

Prilikom strukturiranja podataka e-knjige najvažnije je pripaziti na ispravnost podataka, u smislu da se koriste pravilne oznake i elementi. Ako su podaci pravilno označeni, sustavi za iščitavanje mogu ispuniti svoju namjenu. Uključivanje matematičkih formula u obliku slika krši ovo osnovno pravilo pristupačnosti i ispravnosti podataka. Mehanizam za iščitavanje izgovorit će alternativni tekst koji je pridodan slici, ako je prisutan. Ako alternativni tekst nije uvršten, mehanizam će sliku ignorirati, a čitatelj neće niti znati da je propustio dio sadržaja [41]. Osnovno dodavanje teksta koji opisuje sliku (`<img>`) vrši se putem atributa `alt`. Ako je sadržaj prekompleksan da bi se uvrstio u `alt` opis slike, postoji mogućnost upotrebe WAI-ARIA (Accessible Rich Internet Applications Suite) svojstva `aria-describedby` [42] u kombinaciji s HTML elementima `<figure>` i `<figcaption>`. Primjer osnovnog koda može se vidjeti na slici 7.

```

<figure aria-describedby="opis_slike">
  
  <figcaption>
    Slika 3 – Prva formula
    <details id="opis_slike">
      <summary>detaljan opis</summary>
      <p>
        ...
      </p>
    </details>
  </figcaption>
</figure>

```

Slika 7. Uvrštavanje kompleksnog opisa slike uz pomoć jezika WAI-ARIA i HTML elemenata `<figure>` i `<figcaption>`

Ovim pristupom nailazimo na dva problema. Prvi je semantičko značenje elementa `<figure>`, koji se prema specifikaciji upotrebljava za sekundarni narativ, odnosno za sadržaj koji nije ključan za razumijevanje glavne „priče” te se može čitati kao odvojen i samodostatan, a mehanizam za iščitavanje u tom ga slučaju može preskočiti. Ovakva će definicija ovisiti o situaciji i vrsti matematičkog sadržaja knjige. Drugi je problem sam tekstualni opis matematičkog sadržaja koji često nije linearan poput običnog teksta te način čitanja može u potpunosti promijeniti značenje sadržaja. Manjak standarda za izgovaranje (tekstualno opisivanje) matematičkog sadržaja donekle je riješio sustav MathSpeak koji nudi detaljna matematička gramatička pravila [43]. Za kreiranje takvog sadržaja, međutim, autor je prisiljen naučiti još jedan standard. Primjer:

*x equals y over c plus 1.*

Ova formula može biti protumačena kao:

$$x = \frac{y}{c} + 1 \text{ ili } x = \frac{y}{c+1}$$

Pravilan opis prema sustavu MathSpeak bio bi:

*x equals BEGIN FRACTION y OVER c END FRACTION plus 1.*

Pristupačnost i korisnost SVG slika, zbog vektorske prirode SVG-a, ali i različitosti označnog jezika, bolja je od *bitmap* slika. SVG se može skalirati bez korištenja posebnim alatima za povećavanje i bez efekta pikselizacije. Također se kod može unaprijediti uz pomoć jezika WAI-ARIA, a svaki element može sadržavati `<title>` i `<desc>` (opis) bez opasnosti semantičke nekorektnosti. SVG-u se također može pristupiti i skriptnim jezicima pa su moguće i takve intervencije ako su potrebne u vidu poboljšanja pristupačnosti. Međutim, kao i u slučaju *bitmap* slika, opis sadržaja koji se postavlja u element `<desc>` jest taj koji matematički sadržaj čini pristupačnim te je stoga problem identičan. Primjer osnovnog SVG koda:

```
<svg xmlns:svg="http://www.w3.org/2000/svg">
  <title>Naslov slike</title>
  <desc>
    Opis slike
  </desc>
  ...
</svg>
```

## 8. UGRAĐIVANJE MATEMATIČKOG SLOGA U E-KNJIGE PUTEM JEZIKA MATHML

*Mathematical Markup Language* (MathML) označni je jezik za matematiku koji se osniva na XML gramatici. Godine 2014. postaje službena preporuka i standard kada je uvršten u specifikaciju HTML5. Važan cilj MathML-a, kao obveznog dijela HTML-a, jest da matematički sadržaj više nema potrebe biti slika, čime postaje interaktivan i pristupačan [44].

Standard EPUB 3 podržava podskup jezika MathML koji se odnosi na njegov prikaz – *Presentational MathML*. Ovaj je podskup kreiran kako bi se olakšala i ubrzala podrška MathML-a u sustavima e-čitača te kako bi se promovirala njegova pristupačnost. E-čitači koji funkcioniraju u skladu s preporukom standarda EPUB 3 time moraju pružiti podršku za *Presentational MathML* te ga adekvatno prikazati, moraju moći dinamički generirati alternativni tekstualni sadržaj za potrebe mehaničkog iščitavanja (*text-to-speech*) [45]. Prezentacijski elementi odgovaraju osnovnim tipovima simbola i struktura od kojih se može izgraditi bilo koji tradicionalan matematički zapis. Elementi diktiraju način prikaza, ali su neovisni o mediju u smislu da su oblikovani da sadrže dovoljno informacija za vizualnu i zvučnu interpretaciju. Prezentacijski su elementi namijenjeni da izraze sintaksnu strukturu matematičkog zapisa, poput naslova i paragrafa u standardnom tekstualnom zapisu. Neki se znakovi upotrebljavaju za identifikatore ili operatore koji se tradicionalno prikazuju identično kao neki drugi simbol ili se uopće ne prikazuju. Npr. imaginarno „i” moglo bi se zapisati kao  $i$ , ali bi tada imalo drugačije značenje (značenje varijable), a time i izgovor TTS mehanizma, stoga bi se trebalo primjenjivati `<imaginaryi/>`. Također, entiteti poput `&InvisibleTimes;` i `&ApplyFunction;` obično se ne prikazuju vizualno, ali sadrže važne informacije za izgovor i prijelom te mogu utjecati na značenje izraza [46].

MathML koristi se notacijom Unicode za sve svoje slovne znakove. Za upotrebu ugrađenog MathML zapisa potrebno je primijeniti adekvatni *namespace* unutar elementa `<math>`:

```
<math xmlns="http://www.w3.org/1998/Math/MathML">...</math>
```

Specifikacija opisuje i detaljnu kontrolu prijeloma (prelaska u novi red) unutar matematičkih jednadžbi uz pomoć atributa `linebreak`, `lineleading`, `linebreakstyle` i `linebreakmultchar`, koji, međutim, još uvijek nemaju podršku u preglednicima. Prilikom opisa sadržaja također treba pripaziti da se ne upotrebljavaju imena znakova jer nisu podržani u XHTML5 dokumentima, već njihove decimalne ili heksadecimalne ekvivalente. Na primjer, znak za integral ne smije se opisati HTML imenom `&int`, već heksadecimalnim kodom: `&#x222B;`.

Iako je MathML optimalan način uvrštavanja matematičkih izraza u XHTML i EPUB, najveći problem predstavlja podrška e-čitača koja je različita te se osniva na različitim sustavima. Sustavi koji se baziraju na mehanizmima Gecko i WebKit za prikaz donekle pružaju nativnu (ugrađenu) podršku, dok drugi (npr. Readium) ovise o MathJax Javascript knjižnici [47]. MathJax knjižnica osmišljena je kako bi se premostio jaz podrške MathML-a u preglednicima. Iako prikazuje SVG sliku (u slučaju Readiuma) kada nema podrške preglednika, uvijek ostaje prisutan originalni kod, koji je tada dostupan pretraživačima, ali i dostupan za kopiranje. Pristupačnost je omogućena ugrađenim mehanizmom za povećavanje, ali je i kod dostupan mehanizmima za iščitavanje (TTS) kojima se koriste slijepci [48].

Zbog ovih razlika u podršci očekuju se različiti prikazi u različitim e-čitačima. Također, zbog različitih mehanizama za prikaz pojedina svojstva CSS-a imaju različite razine podržanosti.

Odabir fonta također utječe na prikaz matematičkog zapisa. Prilikom korištenja MathJax knjižnicom ne postoji opcija upotrebe vlastitog fonta jer pruža svoj, ali u drugim slučajevima vrlo je važan odabir fonta jer će o njemu ovisiti podrška svih potrebnih specijalnih znakova koji su tipično nužni u matematičkom sadržaju [34]. *STIX font project* osmišljen je za kreiranje matematičkih zapisa najviše kvalitete i zahtjeva struke te omogućuje besplatnu upotrebu fonta primjerenog za znanstveno i tehničko izdavaštvo [18].

### 8.1. Analiza softverskih alata za kreiranje matematičkih izraza u jeziku MathML

Iako se MathML može kreirati bilo kojim uređivačem teksta, zamišljen je da se oblikuje uz pomoć softverskih alata. Preglednici i e-čitači koji se baziraju na mrežnim pogonima koji nativno podržavaju MathML jesu Firefoxov Gecko i Appleov WebKit. Internet Explorer zahtijeva instalaciju dodatka Math Player, a Chrome korištenje MathJax JavaScript knjižnicom. E-čitač Adobe Digital Editions također podržava MathML. Za kreiranje se mogu primjenjivati različiti softverski alati kao što su FireMath, Wiris i MathType. Također se mogu primjenjivati i alati za konverziju iz LaTeX-a u MathML, primjerice LaTeXML, TeX4ht i MathToWeb. Za testiranje alata za kreiranja MathML-a primjenjuje se Cauchy-Schwartzova nejednakost koja uključuje upotrebu znaka za sumu, velikih zagrada, gornjeg i donjeg indeksa te znaka za nejednakost (slika 8).

$$\left( \sum_{k=1}^n a_k b_k \right)^2 \leq \left( \sum_{k=1}^n a_k^2 \right) \left( \sum_{k=1}^n b_k^2 \right)$$

Slika 8. Cauchy-Schwartzova jednadžba koja se primjenjuje za testiranje alata MathML

MathType je alat koji se obično primjenjuje u kombinaciji s drugim softverima (npr. Word ili InDesign), ali i samostalno. Sučelje mu je jednostavno te nije potrebno mnogo vremena za navikavanje. Zamjerka mu je što je često potrebno upotrebljavati miš za unos predložaka, što usporava rad, no postoji mogućnost povezivanja određenih tipki na tipkovnici s često upotrebljavanim predlošcima (npr. za razlomak). MathType primjenjuje Unicode, ali i dodaje specifične matematičke znakove koji još uvijek nisu uvršteni u Unicode. Nakon kreiranja jednadžbe jednostavnim procesom „kopiraj-zalijepi”, nakon odabranog formata (LaTeX, MathML, TeX...) prikaz ili kod prebacuje se u željeni sustav [49]. MathType dozvoljava preciznu kontrolu pri kreiranju matematičkih izraza u smislu svjesnosti razine oznaka na kojima se autor nalazi (što je posebno važno kod kontroliranja elemenata `<mathrow>` potrebnih za aktivni prijelom). Izraz je semantički i označno korektan te brzo izrađen. Svi posebni znakovi izraženi su uz pomoć heksadecimalnog zapisa. Jedina je zamjerka što prijelom koda s programerskog stajališta

nije korektan te ga je potrebno preoblikovati kako bi ugnježdavanje elemenata bilo jasno te kako bi kasnija intervencija koda bila olakšana. MathType također kreira i elemente `<mstyle>` s atributom `displaystyle="true"`, potrebne, za pravilan prikaz posebnih znakova (poput sume ili integrala) u većoj veličini. Velika mu je prednost povezanost s drugim softverskim alatima za oblikovanje poput InDesigna i Worda te različiti formati izlaza (MathML, LaTeX). MathType se također može upotrebljavati i kao konverter iz formata Latex u MathML i obratno.

Firemath je alat koji je kreirao Firefox kao ekstenziju za preglednik. Sučelje je intuitivno, ali unos se mora obavljati isključivo klikanjem tipke miša, što usporava proces. Prednost je što se kod generira za vrijeme kreacije, no svejedno je ponekad teško shvatiti u kojoj se razini nalazi trenutna pozicija. Generirani je kod lijepo oblikovan i označno točan, iako nema potreban *namespace*, `<mstyle>` sadrži vrijednost u postocima, a ne ključnu riječ i posebni su znakovi izraženi HTML riječima, što je u standardu EPUB zabranjeno. Softver je zamišljen za kreiranje MathML izraza specifično za Firefox pa na taj način i oblikuje.

Wiris je još jedan sličan MathML i LaTeX uređivač teksta s jednostavnim i logičnim sučeljem. Kreiranje ne traje dugo te se istovremeno prikazuje sam kod. Negativna je strana da kod nije prelomljen pa je naknadno uređivanje nemoguće prije nego što se kod uredi. *Namespace* je automatski generiran, primjenjuju se heksadecimalni izrazi za sumu, a upotreba je elemenata `<mrow>` minimalna, što olakšava eventualno dorađivanje. Semantički je i označno točan.

Web Equation je MathML i LaTeX uređivač teksta tvrtke My Script u kojem se izrazi kreiraju crtanjem rukom ili posebnom olovkom na tabličnim računalima. Moguće je i korištenje mišem na osobnom računalu, no nije vrlo precizno. Predstavlja najbrži način kreiranja matematičkog sadržaja jer se ne mora učiti sučelje ni pozicija određenih znakova, već se prirodno piše rukom. Rezultati su precizni, prepoznavanje rukopisa odlično je i semantički točno. Pravilno primjenjuje `<mstyle>` i heksadecimalno izražava znak za sumu (čak i dodaje komentar s imenom znaka). Kreira previše nepotrebnih elemenata `<mrow>` i nema baš mnogo mjesta za pisanje pa se dugačke formule moraju generirati u komadima.

Alternativan je način za kreiranje MathML zapisa primjenom konvertera kao što su TeXZilla i Math To Web. Math To Web konvertira LaTeX u MathML u internetskom pregledniku. Semantički nekorektno kreira element `<math>`, iako matematički sadržaj nije tabličnog tipa, te dodaje mnoštvo nepotrebnih elemenata poput `<mspace>` i *inline* CSS stilova. Pravilno kreira *namespace* i posebne znakove u heksadecimalnom zapisu. Nije teško izvući pravilni dio izraza iz nepotrebne tablice.

TeXZilla Firefox također konvertira iz LaTeXa u MathML i pritom unutar elementa `<annotation>` ostavlja i izraz zapisan u LaTeXu. Kod nije prelomljen te je stoga i nečitak. Znak za sumu izražen je Unicode znakom, što zahtijeva ručnu izmjenu u heksadecimalni zapis potreban za format EPUB. Nužna *namespace* deklaracija pravilno je kreirana i postavljena u kod. Ovaj konverter ne kreira elemente `<math>` koji se mogu primjenjivati za stiliziranje grupacije koda.



## 8.2. Analiza pristupačnosti sadržaja oblikovanog jezikom MathML

Važna je prednost e-knjiga, a time i e-čitača, mogućnost adaptacije sadržaja različitim veličinama ekrana. Sadržaj se reformatira prema dostupnoj veličini zaslona i u slučaju povećanja veličine pisma i promjene orijentacije zaslona [50]. Pristupačnost u vidu adaptacije sadržaja limitirana je ako je sadržaj u obliku slike. U slučaju matematičkog sadržaja, koji je u biti jedan vid tekstualnog sadržaja, pristupačnost je gotovo nepostojeća ili komplicirana za korisnike na mobilnim uređajima. Kada se matematički sadržaj oblikuje uz pomoć označnog jezika MathML, e-čitač procesira sadržaj baš kao i bilo koji drugi tekstualni sadržaj, što znači da je moguće povećavanje veličine pisma, ali i potrebno reformatiranje za prilagodbu veličinama zaslona. Jezik MathML, u kombinaciji s CSS-om, nudi naredbe potrebne za kontrolu prijeloma matematičkih formula, a problemi podržanosti, prilagodbe i moguća rješenja istražuju se u ovom radu. Pristupačnost i korisnost koju MathML također pruža jest dostupnost sadržaja mehanizmima za pretraživanje, za potrebe pretraživanja sadržaja unutar e-knjige, ali i za potrebe interaktivne manipulacije podacima.

Kada je matematički sadržaj uključen u obliku grafike (slike), nedostupan je mehanizmima za iščitavanje. Ostali uobičajeni problemi pristupačnosti digitalnoga matematičkog sadržaja uključuju nemogućnost povećavanja sadržaja po potrebi, nemogućnost manipulacije podacima i nemogućnost strukturiranog oblikovanja sadržaja. Ako se matematički sadržaj oblikuje semantički pravilno, takav sadržaj ima sve uvjete pristupačnosti [51]. Design Science Math Player [52] softver je koji omogućuje iščitavanje matematičkog sadržaja na platformama Windows, u Internet Exploreru i Firefoxu. Na Appleovim sustavima iščitavanje je omogućeno tehnologijom Apple VoiceOver [53]. Iako se e-čitači osnivaju na tehnologijama preglednika, to nije garancija da imaju identične mehanizme pa se ne može zaključiti točno kako će se određeni e-čitač ponašati na temelju toga kako se preglednik ponaša (npr. Safari i iBooks). Ako se upotrebljava MathJax knjižnica za prikazivanje MathML sadržaja, pristupačnost je smanjena jer je *source* (izvorni sadržaj) dostupan samo određenim mehanizmima iščitavanja, iako je povećavanje i kopiranje uvijek dostupno.

Ako sadržaj nije kvalitetno strukturiran, što je često u slučaju konverzije, sadržaj e-knjige neće biti dostupan TTS mehanizmima. Pri oblikovanju sadržaja za pristupačnost važno je informirati korisnika o tipu sadržaja s kojim se susreće te mu omogućiti da određenu sekciju, prema želji, preskoči, što se postiže upotrebom semantički točnih oznaka [54]. Matematički je sadržaj u obliku slike neispravan jer nije grafika, a MathML predstavlja očiti izbor pri biranju semantički pristupačnog načina uvrštavanja matematičkog sadržaja [55].

Digital Accessible Information System (DAISY) način je kreiranja digitalnih zvučnih knjiga za ljude koji žele slušati i navigirati napisani materijal prezentiran u formatu koji se može slušati. Mnogi korisnici DAISY knjiga imaju neku vrstu nemogućnosti čitanja poput sljepila, slabog vida, disleksije i slično. Upotrebom DAISY formata knjige sadržaj je prezentiran sa sustavom navigacije unutar sekvencijalne i hijerarhijske strukture koja se sastoji od (označnog) teksta sinkroniziranog s audiom [56]. Standard DAISY nudi vodilje koje objašnjavaju kako se koristiti označnim elementima za matematičke izraze u DAISY knjigama prema standardu MathML. Sav sadržaj koji prikazuje matematičke izraze trebao bi se oblikovati jezikom MathML, samo se slike ne bi trebale upotrebljavati za prikaz matematičkih izraza jer se ne mogu renderirati sa sintetičkim govorom ili konvertirati u Braille. U tipičnoj primjeni prezentacijskog MathML-a vizualni prikaz matematičkog sadržaja mora biti što sličniji tradicionalnom prikazu poput ručno pisane matematike te nije krucijalno da je označeni sadržaj semantički razumljiv sve dok je vizualni prikaz točan. U slučaju pristupačnih DAISY knjiga kojima pristupaju čitatelji vidnih ili mentalnih oštećenja MathML sadržaj mora biti pogodan za renderiranje putem mehanizama za iščitavanje ili softvera za prijevod u Braille. Ovo znači da je semantičko označivanje sadržaja jednako važno kao i vizualni prikaz. Budući da se MathML sve više prihvaća i ugrađuje u softver, produciranje sadržaja prema specifikaciji MathML osigurava prenosivost sadržaja i konzistentnost prikaza, bez obzira na medij. XSL transformacije i procesuiranje teksta semantički pravilnog sadržaja producirat će predvidljivije rezultate [57].

### 8.3. Analiza i komparacija primjera e-knjiga s ugrađenim matematičkim sadržajem

1. PRIMJER: *BITMAP SLIKE* iz knjige *Theoretical Aerodynamics* (izdavač: Wiley, 2013.)

EPUB knjiga oblikovana je u HTML-u, a matematički je sadržaj ubačen isključivo u obliku *bitmap* slika (format GIF). Zbog ovisnosti o rezoluciji slika, prikaz dramatično degradira prilikom povećavanja slike koje se mora vršiti dvostrukim dodirrom (eng. *double tap*), što smanjuje čitljivost i prisiljava korisnika na prekid tijekom čitanja. Prilikom povećavanja veličine fonta slike ostaju istih dimenzija. Na ovaj način matematički sadržaj nije pristupačan ni slabovidnima koji bi htjeli povećati veličinu pisma ni slijepima, jer zapis u slikama nije dostupan TTS mehanizmima (posebice jer nije uvršten ni opis slika).

Matematički sadržaj se ne može pretraživati niti prebaciti u drugi format. Prijelom je optimalan, iako nije tečan, a na zaslonima manjih dimenzija gdje se slike prilagođavaju veličini zaslona presitan za čitanje. Ažuriranje sadržaja zahtijeva izradu nove *bitmap* slike. Također, veličina datoteke neizbježno je mnogo veća u usporedbi s identičnom knjigom oblikovanom isključivo označnim jezikom, ali je prikaz matematičkog sadržaja zajamčen.

2. PRIMJER: *MATHML MTABLE* iz knjiga *Linear Algebra* (IDPF – EPUB 3 samples) i *The Theory of Heath Radiation* (Infogrid Pacific – sample)

Obje knjige služe kao ogledni primjeri EPUB e-knjiga koje primjenjuju isključivo MathML za matematički sadržaj. Prva knjiga sve matematičke formule koje se nalaze samostalno u retku smješta u MathML element `<math>`, a druga samo one izraze koji su kompleksniji odnosno dulji. Element `<math>` namijenjen je označavanju matrica i tablica te omogućava kontrolu prijeloma matematičkih izraza, slično kako se HTML element `<table>` u prošlosti upotrebljavao za dizajn prijeloma mrežnih stranica [58]. Jednako kao i u *web* dizajnu, tako je i u EPUB e-knjigama važno separirati sadržaj od stila te se nikako ne bi trebalo koristiti označnim elementima za sadržaj koji to i ne predstavlja. Lažno prezentiranje informacija posebice šteti pristupačnosti sadržaja jer čitatelj dobiva lažnu informaciju o samoj strukturi sadržaja. U praktičnom smislu, element `<math>` omogućuje kontrolu prijeloma sadržaja, čak i određenu količinu prilagodbe veličini ekrana. Ako tablice imaju određenu fiksnu veličinu, u aplikaciji iBooks

takav se izraz prikazuje tako da se pojedini slovni znakovi preklapaju i postaju u potpunosti nečitki (posebno kod povećanja veličine fonta). Pri mehaničkom iščitavanju zvučna reprodukcija je usporena, ali omogućena. Potrebno je uzeti u obzir da će slijepa osoba možda takav sadržaj i preskočiti jer će misliti da je riječ o tablici, a ne o matematičkom izrazu. Povećavanje sadržaja omogućeno je iako može polučiti neželjene rezultate poput prelamanja unutar oblikovane ćelije tablice ili već napomenutog preklapanja znakova u aplikaciji iBooks. Pretraživanje i kopiranje teoretski su mogući jer je izraz oblikovan označnim jezikom MathML i koristi se fontom za prikaz znakova. Najveći je nedostatak prebacivanje u druge formate jer će se informacije oko tablice također prenijeti i u drugi jezik. Ako nema podrške jezika MathML u nekom e-čitaču, sadržaj se neće prikazati pravilno. Uvlake, koje se oblikuju kod dugačkih matematičkih izraza koji se prelamaju u više redaka, moguće je ukomponirati u ovakav dizajn jer se prva ćelija na početku svakog retka, započevši s drugim, može ostaviti prazna.

3. PRIMJER: EPUB:SWITCH MATHML, SVG, BITMAP iz knjige *On Optimal Bit Allocation for Classification-Based Source-Dependent Transform Coding* (Hindawi Publishing Corporation)

U ovom primjeru, iako u formatu EPUB 2, primijenjena je opcija zamjenskog (eng. *fallback*) sadržaja metodom *epub:switch*. Element `<epub:switch>` dozvoljava da e-čitač prepozna *namespace* deklaraciju koju podržava i prikaže ga. Na taj se način autori e-knjiga ne moraju brinuti o tome je li neka XML gramatika podržana, već mogu uvrstiti alternative, a e-čitač prikazuje gramatiku kada je prepoznata. Ako ne podržava nijednu, mora postojati zadana vrijednost u XHTML inačici koju će svi podržati sustavi. Proces se može apstrahirati na sljedeći način [34]:

```
<switch>
  <case 1 je li podržan MathML?>
    prikaži MathML
  <case 2 je li podržan SVG?>
    prikaži SVG
  <default>
    prikaži XHTML sadržaj (u ovom slučaju JPG slike)
```

Sustav *switch* postoji još od specifikacije EPUB 2, no pregledavanjem ovog primjera u različitim e-čitačima uočeno je da podrška još uvijek nije optimalna. U nekim se e-čitačima prikazuju sve tri inačice (iBooks, EPUBReader, Calibre), u nekima dvije (IDEAL Reader), a samo se u e-čitačima Adobe Digital Editions i Radium prikazuje samo jedna. Ovakav prikaz u potpunosti onemogućuje čitanje s razumijevanjem. Metoda *epub:switch* idealna je, u slučaju da je podržana, za upotrebu označnih jezika koji još uvijek nemaju općenitu podržanost u e-čitačima, kao osiguranje da se sadržaj prikaže korisniku. Negativna strana metode jest da je potrebno mnogo više truda i vremena za kreiranje ovakve e-knjige, a i veličina takve datoteke višestruko se povećava.

Tablica 3. Usporedba načina ugrađivanja matematičkog sadržaja u EPUB e-knjige

metoda	prikaz	dizajn	aktivni prijelom	pristupačnost	pretraživanje	konverzija	semantika
<i>bitmap</i>	da	da	ne	ne	ne	ne	ne
MathML <math></math>	da	da	da/ne	da	da	ne	ne
<i>epub:switch</i>	višestruki	-	da	da	da	ne	da

## 9. EKSPERIMENTALNA RJEŠENJA PRIKAZA MATEMATIČKOG SLOGA U EPUB E-KNJIZI NA STANDARDNIM E-ČITAČIMA

Preciznije oblikovanje i kontrola prikaza matematičkih formula u EPUB e-knjizi u potpunosti ovisi o e-čitaču i njegovoj tehnologiji prikaza. Svaki e-čitač ima različitu razinu podržanosti svih primijenjenih tehnologija i njihovih kombinacija: MathML, CSS i JavaScript. S obzirom na to da ponašanje e-čitača nije moguće pretpostaviti, pojedini se problemi identificiraju eksperimentalnom metodom. Kreirana EPUB e-knjiga s ispravno oblikovanom matematičkom formulom jezikom MathML testira se na različitim platformama kako bi se identificirali specifični problemi prikaza te zatim individualno riješili kako bi e-knjiga bila pristupačna bez obzira na uređaj koji korisnik odabere.

Testiraju se e-čitači:

1. EPUBReader na Firefoxu (mehanizam za prikazivanje Gecko)
2. iBooks na Appleovim uređajima (mehanizam za prikazivanje WebKit)
3. Readium na Chromeu (mehanizam za prikazivanje Blink)
4. Adobe Digital Editions na Appleovim uređajima (mehanizam za prikazivanje WebKit)
5. IDEAL Group Reader na platformi Android (neutvrđen mehanizam za prikazivanje)
6. Calibre (mehanizam za prikazivanje QT WebKit).

## 9.1. Eksperimentalno rješenje za e-čitač EPUBReader

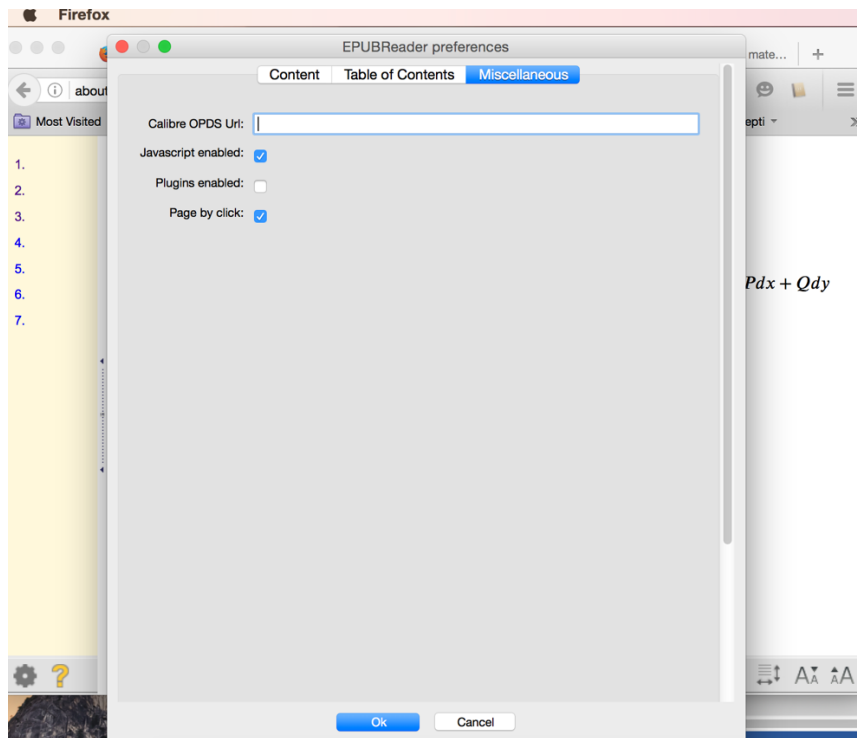
EPUBReader [59] je besplatni dodatak za preglednik Firefox koji omogućava čitanje EPUB e-knjiga u samome pregledniku bez potrebe instalacije dodatnih softvera. Za prikaz se koristi mehanizmom Gecko za prikaz mrežnog sadržaja te se može upotrebljavati na bilo kojem operativnom sustavu na koji se može instalirati preglednik Firefox (Windows, Mac OS X, Linux). Firefox ima ugrađenu (nativnu) podršku prikaza jezika MathML. Slika 9 prikazuje matematičku formulu oblikovanu jezikom MathML unutar EPUB 3 e-knjige bez intervencije CSS-a ili skriptnog jezika. Formula se prikazuje s punom podrškom MathML-a; svi posebni znakovi podržani su zbog specijaliziranog fonta STIX koji je ugrađen u preglednik Firefox. Prilikom promjena veličine prozora preglednika matematička se formula prilagođava dinamičkim prijelomom na adekvatnim mjestima zbog primjene elemenata `<mrow>` za grupiranje bez potrebe za osvježavanjem prozora. Na slici 9 vidi se da uvlaka koja je potrebna zbog razumijevanja i čitkosti kompleksnih matematičkih izraza, a koja se prema specifikaciji izvršava uz pomoć MathML atributa `indentalign` i `indenttarget`, nije aplicirana zbog njihove nepodržanosti.



Slika 9. Prikaz u e-čitaču EPUBReader unutar Firefoxa (Gecko) koji ima nativnu MathML podršku

Problem koji je potrebno riješiti u e-čitaču EPUBReader (Firefox) jest aplikacija uvlake svakog prelomljenog retka u veličini udaljenosti prvog matematičkog operatora od početka izraza. Prilikom prvog pokušaja intervencije skriptnim jezikom JavaScript uočeno je da JavaScript nije uključen kao početna vrijednost u ovom e-čitaču. Jedan od većih dodataka novom formatu EPUB 3, za razliku od njegova prethodnika EPUB 2, jest dodatak podržanosti skriptiranih sadržaja. Taj specifični novi dodatak ima i uvjete korištenja kako bi se osigurala pristupačnost sadržaja. Specifikacija navodi kako je glavni princip skriptiranja *progresivno poboljšanje*: „kada e-čitač renderira dokument bez skriptne podrške ili s isključenom podrškom skriptiranja, sadržaj dokumenta mora zadržati vlastiti integritet, ostati čitljiv korisniku bez gubitaka informacija ili drugih znatnih gubitaka” [60]. Progresivno poboljšanje ne podrazumijeva izbacivanje skriptiranog sadržaja, već skriptiranje nije obvezno. Skriptiranje bi trebalo poboljšati iskustvo kada je dostupno. Potrebno je izbjegavati situaciju gdje primarni narativ ovisi o skriptiranju, odnosno da zahvaljujući skriptiranju određeni dio sadržaja nije dostupan čitatelju. U primjeru kompleksnih matematičkih izraza gdje se uvlaka može aplicirati isključivo skriptnim putem zbog nepodržanosti postojećih atributa, primarni narativ, odnosno sadržaj uvijek je dostupan bez obzira na skriptnu podržanost. Sadržaj uz pomoć skripte dobiva aktivno poboljšanje u smislu čitljivosti takvog specifičnog sadržaja gdje su čitatelji naviknuli na određene tipografske konvencije, istodobno ga ne sakrivajući. U slučaju e-čitača EPUBReader koji nema standardno uključenu podršku skriptnog sadržaja postiže se situacija progresivnog poboljšanja. Ako korisnik samostalno ne uključi podršku skriptiranja, matematički je sadržaj i dalje vidljiv i čitak, ali ako je uključi, pozitivan doživljaj čitanja bit će povećan (slika 10). Jedini je problem u osviještenosti čitatelja o danoj situaciji. U većini situacija čitatelj samo želi čitati svoju e-knjigu te se ne zamara postavkama e-čitača. Bilo bi uputno dati do znanja čitatelju kako da poboljša svoje iskustvo čitanja te dati upute o uključivanju skriptne podrške, ali detaljnije upute nisu moguće jer se postavke razlikuju od e-čitača do e-čitača.





Slika 10. Prikaz postavki skriptiranja e-čitača EPUBReader u Firefoxu

Funkcija JavaScript za aplikaciju uvlake započinje dohvaćanjem matematičkog izraza metodom `getElementByTagName ( )` koja vraća kolekciju elemenata oznake `math` te postavljanjem dohvaćenog u varijablu zbog opetovane upotrebe.

```
var matem = document.getElementsByTagName('math');
```

Najprije je potrebno utvrditi postojanje i udaljenost prvog samostalnog matematičkog operatora. To se čini prolaženjem petljom `for` kroz prvi nivo djece elemente glavnog elementa `<math>` svojstvom `Node.childNodes`. Zatim se dohvaća prvi samostalni operator uspoređivanjem stringa oznake metodom `Element.tagName` sa stringom `<mo>` te sprema u zasebnu varijablu. MathML element `<mo>` predstavlja operator u širokom smislu, osim matematičkih operatora element uključuje i operatore poput zagrada, zareza i točka zareza ili linija koje označavaju apsolutnu vrijednost. Zbog toga je važno prilikom oblikovanja matematičke formule koja se planira prelamati i uvlačiti postaviti prvi operator kao samostalno dijete najvišeg nivoa:

```

<math>
  <mrow> ... </mrow>
  <mo>=</mo>
  <mrow> ... </mrow>
  ...
</math>

```

Na taj se način izbjegava dohvaćanje neželjenih, a prisutnih elemenata `<mo>`.

Horizontalna udaljenost prvog operatora od početka matematičkog izraza potrebna za utvrđivanje i postavljanje veličine uvlake dohvaća se metodom `Element.getBoundingClientRect().left`. Ova metoda vraća veličinu elementa i njegovu relativnu poziciju prema prikaznom području. Svojstvo `left` ove metode dohvaća udaljenost u pikselima od lijevog ruba CSS okvira.

```

//dohvacanje udaljenosti prvog operatora
var firstOperator = [];
for (var i = 0; i < matem[0].childNodes.length ; i++) {
  var child = matem[0].childNodes[i];
  if(child.tagName == 'mo'){
    firstOperator.push(child);
  }
};
var uvlaka = firstOperator[0].getBoundingClientRect().left;

```

Slika 11. Dohvaćanje prvog operatora i njegove udaljenosti od lijevog ruba

Nakon utvrđivanja veličine uvlake započinje dohvat svih elemenata `<mrow>` kako bi se utvrdila mjesta prijeloma te postavila uvlaka. Dohvaćanje se vrši petljom `for` koja prolazi kroz djecu elemente najvišeg nivoa glavnog elementa `<math>` te se uspoređuju njihova imena s imenom oznake `mrow` te se nadodaju u polje `mrows` ako je upit pozitivan metodom `push`.

```

//dohvacanje svih mrow jedinica za prijelom
var mrows=[];

for (var i = 0; i < matem[0].childNodes.length ; i++) {
    var child = matem[0].childNodes[i];
    if(child.tagName == 'mrow'){
        mrows.push(child);
    }
};

```

Slika 12. Dohvaćanje svih elemenata <mrow> zbog apliciranja uvlake

Idući je korak dohvaćanje udaljenosti prvog retka od vrha prikaznog okvira identičnom metodom kao i za dohvaćanje vrijednosti uvlake

`Element.getBoundingClientRect()`, ali uz pomoć svojstva `top`. Element čija se udaljenost najprije provjerava jest prvi član polja `mrows`. Ova vrijednost u pikselima potrebna je kako bi služila kao referentna veličina za usporedbu s ostalim elementima polja `mrows` u svrhu utvrđivanja koji je to element polja koji se prelomio u novi redak, odnosno na kojem se mjestu unutar matematičkog izraza dogodio prijelom kako bi se na tom mjestu mogla aplicirati uvlaka. Zatim se dohvaća udaljenost svih elemenata polja `mrows` te se uspoređuje s udaljenošću prvog elementa polja `mrows`. Ako je udaljenost prvog elementa manja od trenutno dohvaćenog elementa <mrow>, na dotični se element postavlja lijeva margina u vrijednosti prije dohvaćene udaljenosti prvog operatora od lijevog ruba prikaznog područja. Margina se postavlja metodom `element.setAttribute()` postavljanjem atributa `style` vrijednosti `margin-left` na veličinu varijable uvlaka:

```

var resizer = function(){
    resetOffsets();
    var currentY = mrows[0].getBoundingClientRect().top;

    var startIndex=0;
    while(startIndex < mrows.length-1){

        currentY = mrows[startIndex].getBoundingClientRect().top;

        for (i = startIndex; i < mrows.length ; i++) {
            startIndex=i;
            var el=mrows[i];
            var elY;
            elY = getElementY(el);

            if(currentY < elY){
                el.setAttribute('style', 'margin-left:' + uvlaka + 'px');
                break;
            }
        };
    }
};

```

Slika 13. Usporedba udaljenosti pojedinih elemenata <row> po osi y te aplikacija uvlake

Funkcija `resizer` za postavljanje željene uvlake započinje funkcijom `resetOffsets` čiji je cilj resetiranje svih lijevih margina na 0 kako bi se prilikom mijenjanja veličine prikaznog područja (mijenjanjem veličine prozora preglednika) prvotno aplicirane uvlake resetirale i pritom ne bi stvarale rupe unutar matematičkog izraza.

```

var resetOffsets = function(){
    for (var i = 0; i < mrows.length ; i++) {
        var el=mrows[i];
        el.setAttribute('style', 'margin-left:0px');
    };
};

```

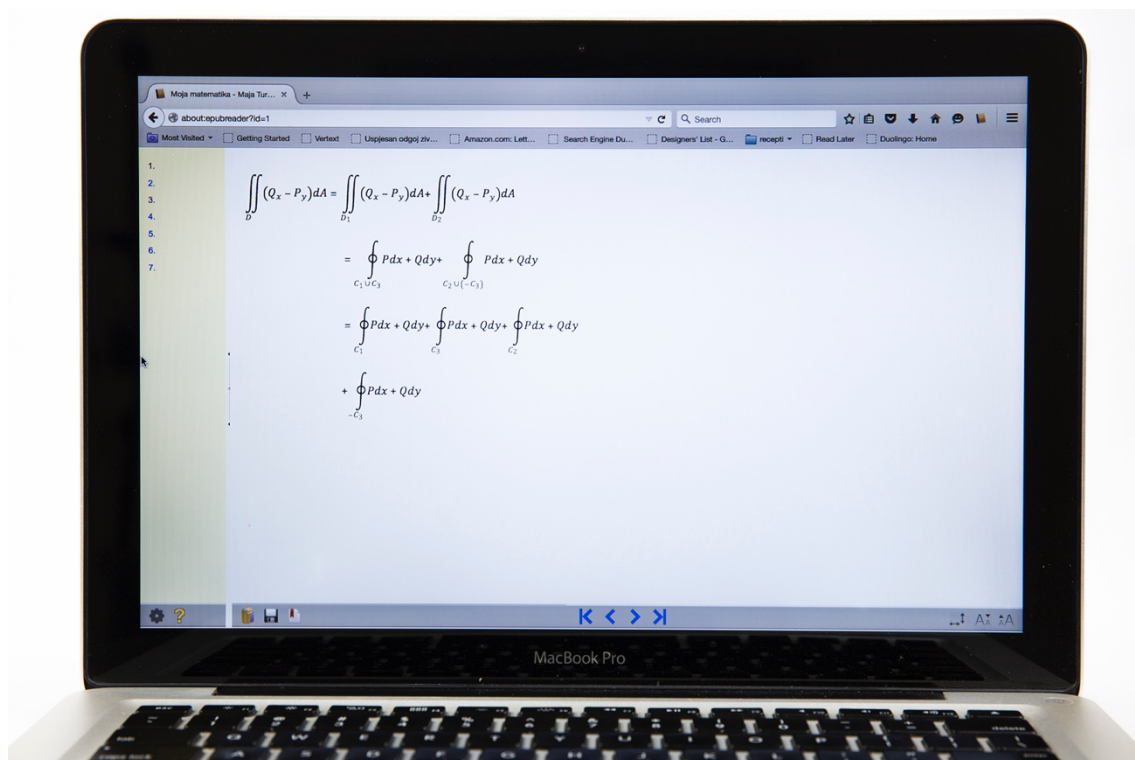
Slika 14. Resetiranje margina na vrijednost 0 px svih elemenata <row>

Funkcija za aplikaciju uvlake poziva se upravljačem događaja prozora `onload`.  
`window.onload = uvlaka;`

Funkcija za postavljanje margine poziva se prilikom mijenjanja veličine prozora preglednika.

```
window.addEventListener("resize", resizer);
```

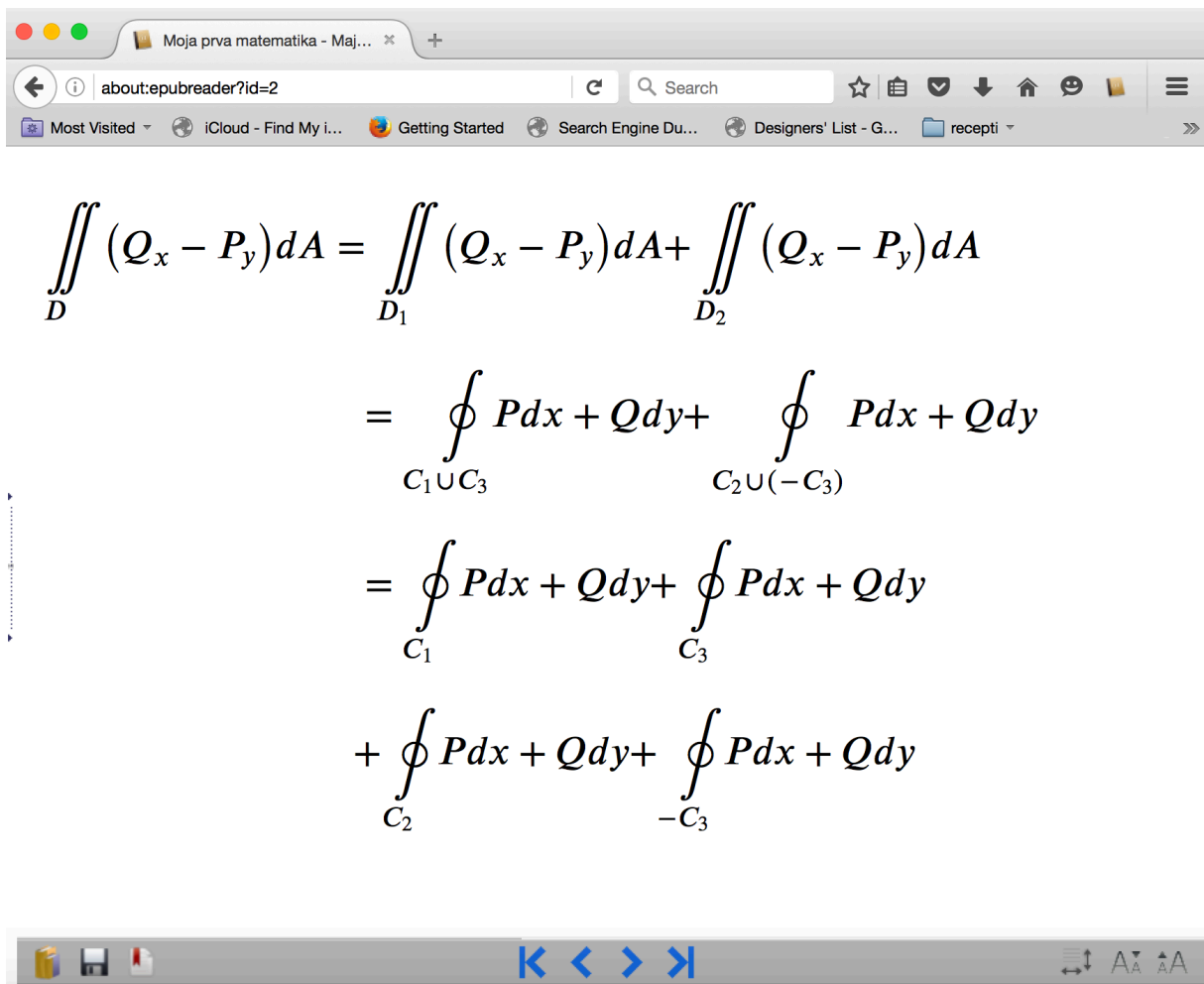
Ovom se funkcijom postiglo da korisnik ne mora osvježiti preglednik (e-čitač) nakon mijenjanja njegove veličine kako bi dobio optimalan prikaz matematičke formule.



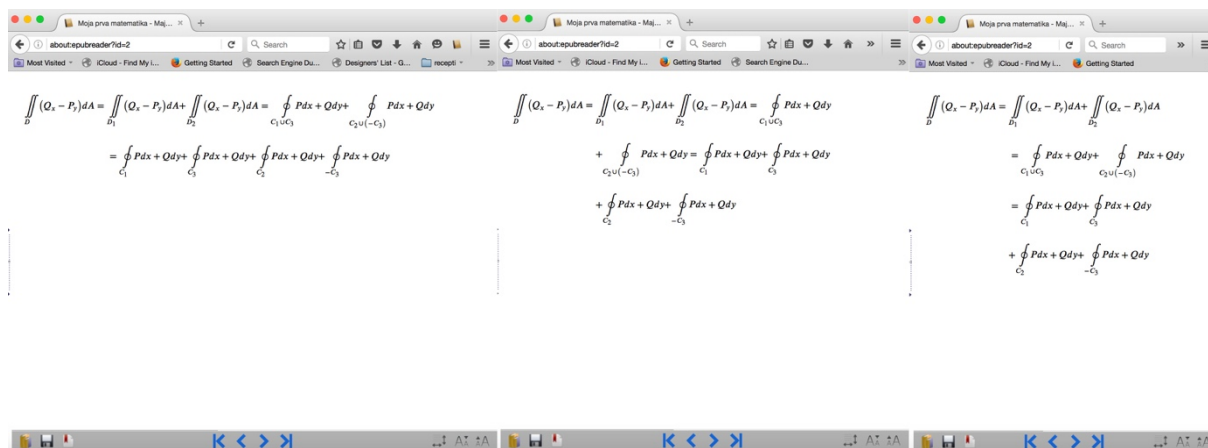
Slika 15. Prikaz EPUB e-knjige u e-čitaču EPUBReader (Firefox) nakon intervencije JavaScriptom. Sadržaj je vidljiv te optimalno tipografski oblikovan aplikacijom uvlake u veličini udaljenosti prvog operatora.

Intervencijom kreirane JavaScript formule postignut je krajnji cilj optimizacije prikaza i tipografskog standarda prilikom oblikovanja kompleksnih matematičkih izraza unutar EPUB e-knjiga za e-čitač EPUBReader u Firefoxu. Sadržaj se u cijelosti pravilno prikazuje zahvaljujući mehanizmu prikaza Gecko koji Firefox primjenjuje za renderiranje mrežnog sadržaja, uključujući jezik MathML. Dinamički se prilagođava veličini prozora odnosno prikaznog područja. Tipografski standardizirana uvlaka svakog idućeg retka aplicirana je

uz pomoć skriptnog jezika JavaScript, koja međutim ne umanjuje pristupačnost ako je korisnik ne uključi. Sadržaj je također dostupan mehaničkom iščitavanju TTS tehnologije dostupne na platformi na kojoj se čita zbog primjene jezika MathML. Matematički se izraz prelama na tipografski optimalnim mjestima i znakovi operatora nalaze se na početku retka primjenom funkcije JS i grupiranjem komada sadržaja elementima `<mathrow>`. Sadržaj se također vrlo jednostavno može povećavati ugrađenom opcijom e-čitača bez degradacije prikaza.



Slika 16. Prikaz matematičkog izraza nakon višestrukog povećanja veličine fonta ugrađenom opcijom e-čitača



Slika 17. Prikaz dinamičkog prijeloma tijekom promjene veličine prozora preglednika

## 9.2. Eksperimentalno rješenje za e-čitač Apple iBooks

E-čitač iBooks aplikacija je tvrtke Apple te se pronalazi na operativnim sustavima OS X i iOS, odnosno na osobnim računalima Mac i mobilnim uređajima iPhone i iPad. Za prikaz upotrebljava mehanizam Safari za prikazivanje mrežnog sadržaja baziran na mehanizmu WebKit.

Prilikom otvaranja oblikovane EPUB e-knjige u e-čitaču iBooks uočeno je da se sadržaj ne prikazuje u cijelosti. Matematička se formula prikazuje do ruba zaslona te je zatim odsječena te ni na jedan način nije dostupna (npr. pomicanjem po zaslonu, jer je pomicanje onemogućeno unutar e-čitača). Razlog je nedostupnosti matematičkog sadržaja još uvijek eksperimentalna metoda CSS kojom se koristi iBooks (odnosno Mobile Safari), u ovome slučaju `flexbox`. CSS svojstvo `flex-wrap` određuje hoće li se objekt `flex` prelamati u nove retke ili će se prikazivati u jednom retku. Inicijalna vrijednost ovoga svojstva postavljena je na `nowrap`, što prisiljava objekt da bude prikazan u jednome retku. Ustanovljeno je da se MathML sadržaj u e-čitaču iBooks prikazuje uz pomoć CSS metode prikazivanja `flexbox` te se jedino na taj način može i ispraviti. Postavljanjem svojstva `flex-wrap` na `wrap` sadržaj se prelama u više redaka.





Slika 18. Prikaz oblikovane EPUB e-knjige bez intervencija CSS-a i JavaScripta u e-čitaču iBooks na uređaju iPhone (iOS)

Nakon intervencije CSS svojstvom prikaz je identičan onome u EPUBReaderu u Firefoxu. Time je dokazano da mehanizam Safari za prikazivanje mrežnog sadržaja nativno podržava jezik MathML. S obzirom na to da je problematika koja se odnosi na apliciranje adekvatnih uvlaka zbog čitljivosti identična onoj u EPUBReaderu, rješavanju problema pristupilo se s već kreiranom JavaScript formulom. E-čitač iBooks podržava skriptiranje te nije potrebna nikakva dodatna korisnička intervencija kako bi JavaScript bio uključen. Funkcija kreirana za Firefox, međutim, nije kreirala željene uvlake u sadržaju. Nakon eksperimentiranja s postojećom funkcijom unutar e-čitača iBooks otkriveno je da iBooks ne prepoznaje, odnosno ne podržava metodu `getBoundingClientRect()`. Otkriveno je da se udaljenost prvog operatora može dohvatiti metodom `offsetLeft`, a usporedba elemenata `<math><mathrow>` uz pomoć metode `offsetTop`. Metoda `offsetLeft` dohvaća udaljenost elementa od gornjeg lijevog kuta njegova elementa roditelja, a svojstvo `offsetTop` dohvaća udaljenost elementa od vrha elementa roditelja. Ostatak funkcije zadržava postojeću logiku i algoritam te ostvaruje željene rezultate: matematički se sadržaj prelama kod operatora, aplicira se uvlaka zbog

čitljivosti tog tipa sadržaja, sadržaj se dinamički prelama ovisno o veličini dostupnog prostora na zaslonu uređaja, a sadržaj je pristupačan (TTS i povećanje veličine pisma).



Slika 19. Prikaz rješenja aplicirane funkcije JavaScript za iBooks na platformi iOS

Zbog univerzalnosti primjene kreirane funkcije za uvlake, adaptirana funkcija pokušava se aplicirati unutar EPUBReadera. Optimalna bi solucija bila kreacija funkcije koja bi djelovala na svim e-čitačima. Međutim, metode `offsetTop` i `offsetLeft` nisu podržane u EPUBReaderu pa funkcija adaptirana za iBooks ne može djelovati univerzalno. Zbog toga se na početak funkcije postavlja upit za podržanost pojedine metode koja se zatim po potrebi aplicira. Izmjena, odnosno primjena potrebne metode ostvaruje se JavaScript izjavom `switch` koja obavlja različite akcije prema zadanim uvjetima.

```

var method;

function getElementY (element){
    var y;
    switch(method){
        case 'bounds':
            y=element.getBoundingClientRect().top;
            break;
        case 'offset':
            y=element.offsetTop;
            break;
    }
    return y;
}

function uvlaka (){

    var matem = document.getElementsByTagName('math');
    if(matem.offsetTop == undefined){
        method='bounds';
    }else{
        method='offset';
    }
}

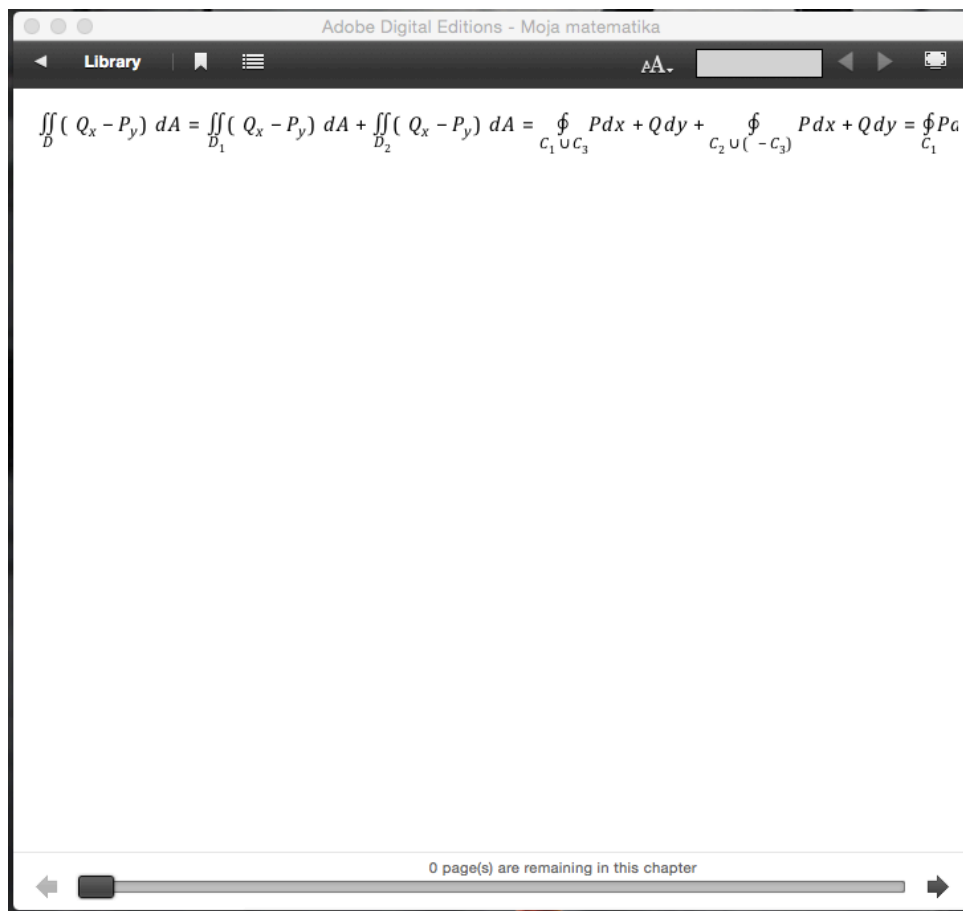
```

Slika 20. Funkcija koja obavlja upit podržanosti metode `offsetTop` izjavom `switch` te shodno primjenjuje potrebnu metodu za dohvaćanje udaljenosti

Podržanost svojstva, poput podržanosti skriptiranja, u e-čitačima ne može se generalizirati zbog kompleksnosti ekosustava koji su pojedine tvrtke kreirale. Zbog toga je potrebno, prilikom kreiranja e-knjige specifičnih zahtjeva, ispitati i testirati podržanost pojedinih svojstva, bez obzira na to što specifikacija pojedinog proizvođača tvrdi da je neko generalno svojstvo podržano. U ovom konkretnom slučaju e-čitači EPUBReader i iBooks podržavaju JavaScript skriptiranje unutar sadržaja e-knjige, ali nije moguće primijeniti jedinstvenu metodu dohvaćanja udaljenosti pojedinog MathML elementa od ruba zaslona (odnosno izvornog HTML elementa). Time je oblikovanje optimiziranog sadržaja e-knjiga prema korisničkim zahtjevima kompleksan posao koji nije moguće automatizirati sve dok tehnologije e-čitača ne postanu usuglašene.

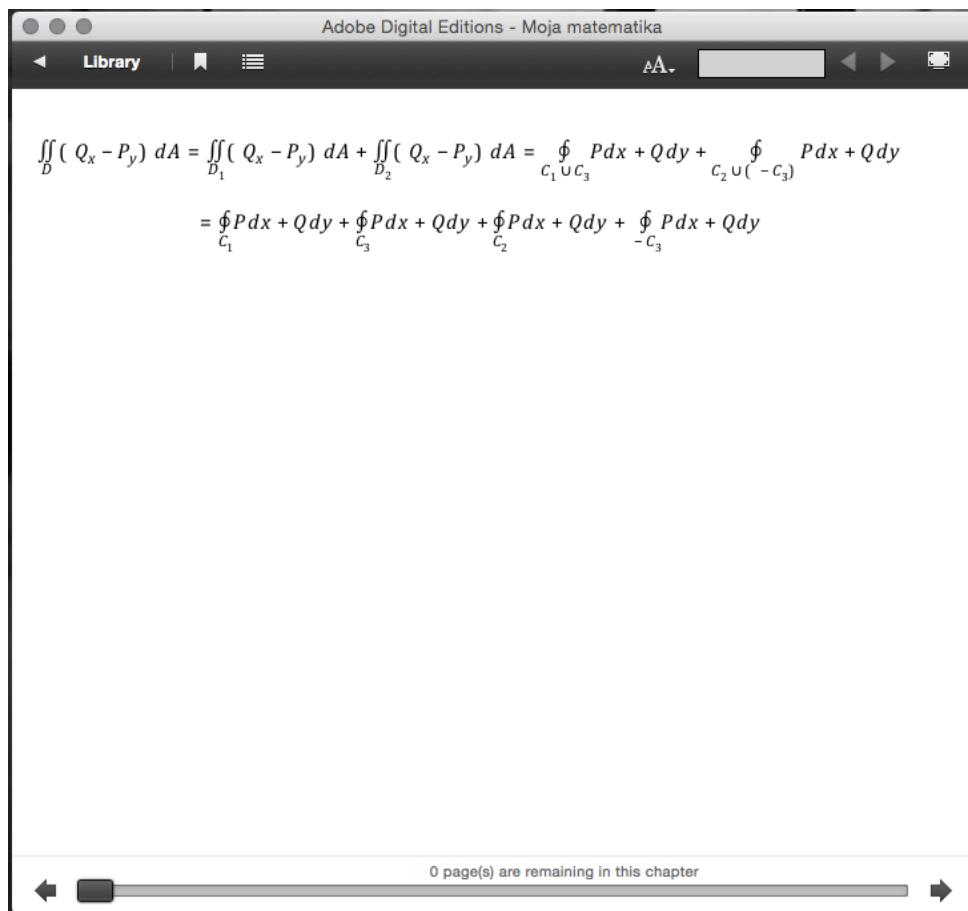
### 9.3. Eksperimentalno rješenje za e-čitač Adobe Digital Editions

Adobe Digital Editions besplatni je e-čitač tvrtke Adobe namijenjen za sve platforme: OS X, iOS, Android i Windows. Prvenstveno je razvijen za čitanje EPUB e-knjiga pa samim time ima veliku podršku specifikacije EPUB 3. Tvrtka Adobe, kao vodeća kompanija u grafičkoj industriji, prihvatila je format EPUB za softver za prelamanje knjiga Adobe InDesign. Danas omogućuje svojim softverskim alatima i kreiranje i čitanje e-knjiga u formatu EPUB 3. S obzirom na to da je Adobe Digital Editions e-čitač namijenjen za više platformi, koristi se različitim mehanizmima prikazivanja jezika MathML. Ovaj je e-čitač testiran na operativnom sustavu Mac OS X koji upotrebljava mehanizam Safari za prikazivanje sadržaja. Prilikom otvaranja pripremljene e-knjige unutar e-čitača Adobe Digital Editions verzije 4.01, u slučaju bez intervencije CSS-om ili JavaScriptom, matematička se formula ne prelama, ali se prikazuju svi potrebni znakovi. S obzirom na to da se matematički sadržaj ne prelama, čitatelju nije omogućen pristup sadržaju u cijelosti jer je akcija *scroll* tipično isključena u e-čitačima. Inicijalni je problem identičan onome u e-čitaču iBooks, kako je bilo i očekivano, s obzirom na mehanizam za prikazivanje. Testiranje je ipak nužno zbog unikatnosti pristupa svakog e-čitača prema specifičnim značajkama koje razvija.



Slika 21. Prikaz e-knjige prije intervencije CSS-om i JavaScriptom u e-čitaču Adobe Digital Editions 4.01 na platformi Mac OS X

Nakon otvaranja matematičkog sadržaja oblikovanog CSS-om i JavaScriptom, apliciranom funkcijom prethodno pripremljenom za EPUBReader i iBooks, sadržaj se prilagođava veličini zaslona u vidu prijeloma na adekvatnim mjestima te se aplicira željena veličina uvlake. S obzirom na to da se Adobe Digital Editions koristi mehanizmom Safari (WebKit) za prikazivanje sadržaja, zaključuje se da je funkcija kreirana za iBooks u ovome slučaju adekvatna. Pristupačnost je također omogućena u vidu povećavanja veličine fonta i mogućnosti pristupa TTS mehanizma na dotičnom operativnom sustavu. Time je ostvarena optimizacija prijeloma matematičkog sloga unutar EPUB e-knjiga na e-čitaču Adobe Digital Editions na platformi Mac OS X.

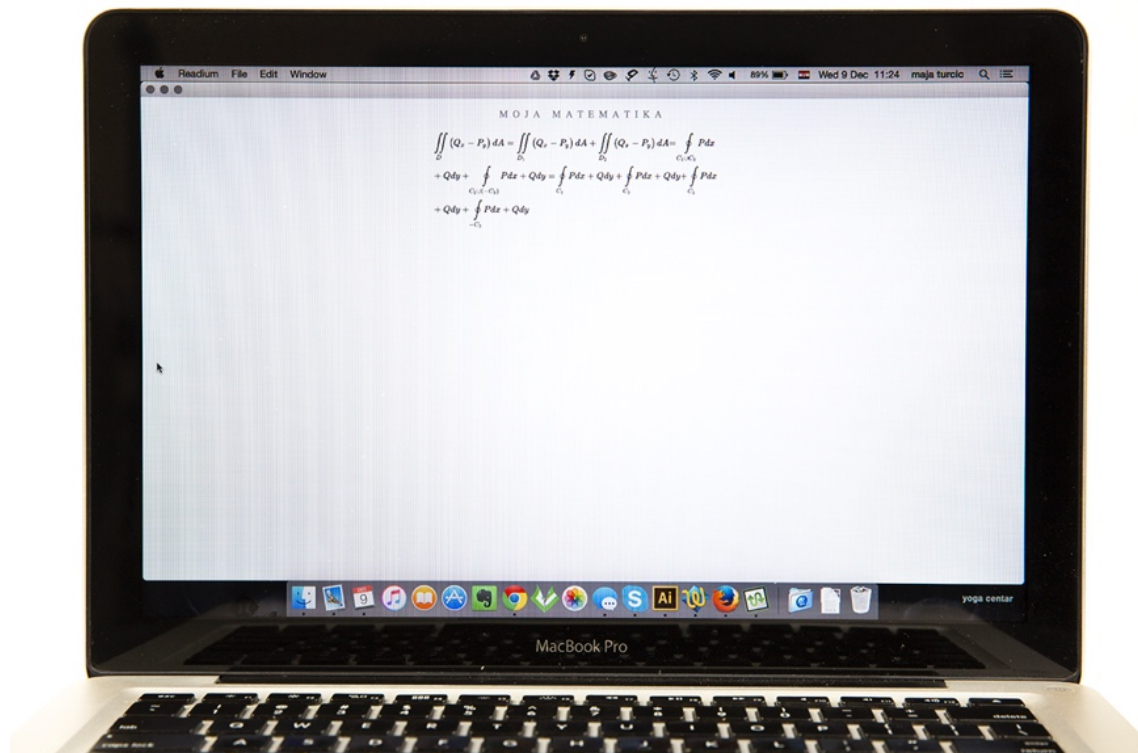


Slika 22. Prikaz e-knjige u e-čitaču Adobe Digital Editions 4.01 na platformi Mac OS X nakon intervencije CSS-om i funkcijom JavaScript

#### 9.4. Eksperimentalno rješenje za e-čitač Radium verzije 2.16.X

Radium [61] je tehnologija otvorenog koda namijenjena ubrzanju prihvaćanja formata EPUB 3 unutar industrije digitalnog izdavaštva. Glavni je cilj projekta Radium proizvesti EPUB e-čitač koji poštuje specifikaciju, a može se primjenjivati u preglednicima, aplikacijama preglednika ili ugraditi kao SDK (Software Development Kit) u nativnu aplikaciju. Projekt ima cilj pružiti funkcionalni EPUB e-čitač za upotrebu na internetu i postoji u dva oblika: kao Radium CloudReader i ekstenzija za preglednik Chrome. Dostupan je u obliku otvorenog koda kako bi osigurao razvitak projekta i popularizirao format EPUB. Radium cilja na usuglašenost posljednje specifikacije EPUB i kao testna platforma za nove EPUB module. Radium je referentni sustav i pogon, ali ne i komercijalni e-čitač.

Radium je testiran na platformi Mac OS X kao ekstenzija preglednika Chrome. Prilikom otvaranja e-knjige u Radiumu vidljivo je da se matematički sadržaj prikazuje te da se prelama u novi redak prema potrebi s obzirom na dostupan prostor prozora aplikacije. Sadržaj se lomi prije matematičkih operatora, što dokazuje uspješnost oblikovanja sadržaja grupiranjem uz pomoć elemenata `<math><mathrow>`.



Slika 23. Prikaz matematičke formule bez intervencije CSS-a ili JavaScripta u e-čitaču Readium u pregledniku Chrome na platformi Mac OS X

Preglednik Chrome nativno ne podržava jezik MathML. Stoga Readium, kako bi zadovoljio zahtjeve specifikacije, omogućuje čitanje matematičkog sadržaja oblikovanog jezikom MathML uz pomoć JavaScript knjižnice MathJax.

MathJax može renderirati matematičke izraze na tri načina: koristeći se HTML-om i CSS-om za prikaz, koristeći se SVG-om ili koristeći se nativnom podrškom preglednika. Ovi su prikazi implementirani s izlaznim procesorima HTML-CSS, SVG i NativeMML. Readium upotrebljava SVG izlazni procesor za prikaz matematičkih izraza uvrštenih jezikom MathML. SVG izlazni procesor upotrebljava tehnologiju Scalable Vector Graphics za prikaz matematičkih izraza. SVG je podržan u svim većim preglednicima, uključujući Chrome. SVG način prikaza visoke je kvalitete i brz, stoga se dobro prikazuje u preglednicima. Mana ovog načina prikaza jest da ne se koristi prednostima fonta STIX za prikaz matematičkih znakova pa ima pristup samo onim znakovima koji se nalaze u mrežnim fontovima. Druga velika mana SVG prikaza jest da jednom kada je matematički izraz renderiran i prelomljen postaje fiksne veličine pa se ne prelama dinamički prilikom promjene veličine prozora.



Izlazni procesori HTML-CSS i SVG implementiraju MathML automatsko prelamanje teksta u novi redak prema specifikaciji. S obzirom na to da prelamanje zahtijeva dodatno procesiranje i može usporiti prikaz, često je automatski isključeno. U testiranoj verziji Radiuma automatski je prijelom uključen, kao što se vidi na slici 15. Automatsko prelamanje teksta pri korištenju SVG prikazom također se može aktivirati konfiguracijom MathJaxa uz pomoć naredbe:

```
MathJax.Hub.Config({
  SVG: { linebreaks: { automatic: true, },
  }
});
```

Svi se posebni matematički znakovi prikazuju zato što MathJax podržava sedam različitih fontova od kojih autor može birati koji će upotrijebiti te dolaze u paketu s JavaScript knjižnicom. Za SVG prikaz MathJax preuzima podatke putanja koji odgovaraju dotičnom fontu. Izlazni procesor SVG neće se izravno koristiti fontovima, već izvedenim SVG podacima putanja kako bi iscrtao putanje potrebnih znakova. Sam je prikaz matematičkih izraza, dakle, izveden u obliku SVG krivulja, a ne znakova fonta. Za apliciranje uvlake na retke koji se lome potrebno je upoznati se s Document Object Modelom (DOM) koji MathJax generira u opciji SVG izlaza. Na slici 16 vidi se kako MathJax kreira SVG elemente `<g>` za grupiranje sadržaja. Ovi elementi odgovaraju MathML elementima `<mathrow>` kreiranima za grupiranje i prelamanje matematičkog sadržaja, što olakšava manipulaciju tim elementima s pomoću JavaScripta.



dohvaća se uz pomoć regularnih izraza JavaScript objekta. Pozicije svih SVG elemenata `<g>` zapisane su u atributu `transform = „translate(x,y)“`, stoga se regularni izraz primjenjuje za pregledavanje atributa `transform` prvog operatora, dohvaća string vrijednost piksela `x` pozicije `translate` te ga postavlja u svoju varijablu koja će se kasnije upotrebljavati za apliciranje uvlake.

```
var operator = window.document.getElementById('firstOperator');
var poz = operator.getAttribute('transform');
var parts = /translate\(\s*(\s*,)+\s*,(\s*,)+\s*\)/.exec(poz);
var xPoz = parts[1];
```

Slika 25. Dohvaćanje prvog operatora i regularni izraz s pomoću kojeg se dohvaća udaljenost prvog operatora od ruba prozora

Zatim se kreira petlja `for` kojom se puni polje s elementima `<g>` kojima će se postaviti upit pozicije te aplicirati uvlaka.

```
var drugiRedovi = svg[0].firstChild.childNodes;
var govi = [];
for (var i = 0; i < drugiRedovi.length ; i++) {
    var child = drugiRedovi[i];
    if(child.tagName == 'g'){
        govi.push(child);
    }
};
```

Slika 26. Prikaz petlje koja dohvaća sve retke matematičkog izraza

Elementi redova dohvaćaju se kao prva djeca SVG elementa te se spremaju u polje pod uvjetom da se element zove `<g>`.

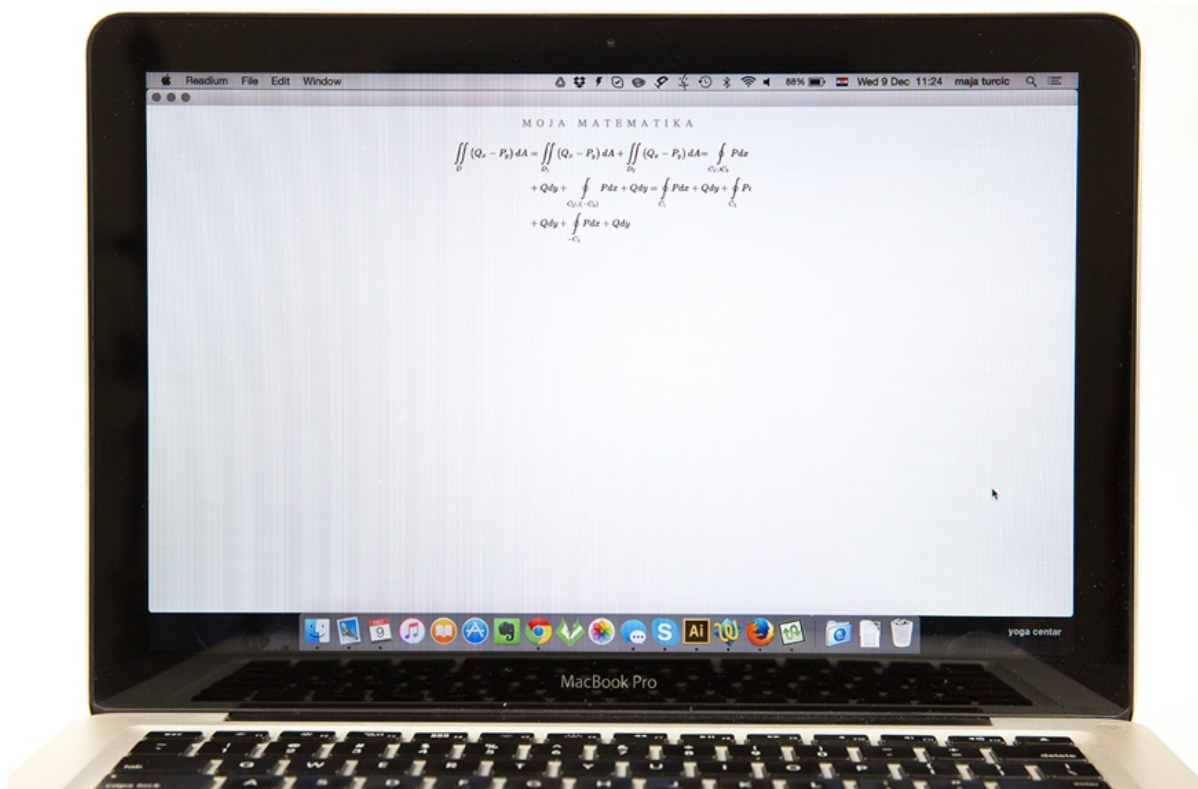
Sada su dohvaćeni svi potrebni elementi te vrijednost uvlake koja se želi primijeniti.

Posljednja petlja u funkciji uvlake započinje se prolaženjem kroz polje sakupljenih elemenata `<g>`, preskačući prvi jer se na njega uvlaka ne aplicira. Elementima se dohvaća njihova pozicija iz atributa `transform` kako bi se mogla izvršiti usporedba udaljenosti po osima `x` i `y` od ruba ekrana. Ako je `x` pozicija elementa jednaka nuli, a `y`

pozicija različita od nule (što znači da se nalazi u drugom, trećem ili inom retku) na taj se element <g> postavlja atribut transform s vrijednošću pomicanja po osi x u vrijednosti prethodno dohvaćene udaljenosti prvog operatora.

```
for (var j = 1; j < govi.length; j++) {
    var koo = govi[j].getAttribute('transform');
    var xy = /translate(\s*(\s*,)+)[,](\s*,)+/.exec(koo);
    var X = xy[1],
        Y = xy[2];
    if(X!=0 && Y!=0){
        govi[j].setAttribute('transform', 'translate('+xPoz+', '+Y+')');
    }
};
```

Slika 27. Prikaz petlje koja utvrđuje pozicije pojedinih grupa elemenata, uspoređuje ih te aplicira novu željenu poziciju



Slika 28. Prikaz formule nakon apliciranja JavaScript funkcije u e-čitaču Readium u pregledniku Chrome

Nakon otvaranja e-knjige s pripremljenom matematičkom formulom te pozivanjem JavaScript funkcije sadržaj je vizualno dostupan te se prelama prilagođavanjem dostupnom prostoru zaslona. Također je vidljivo da se sadržaj prelama na adekvatnim mjestima, gdje svaki redak započinje s operatorom, a sadržaj se ne lomi na tipografski nelogičnim mjestima. Uvlaka veličine udaljenosti prvog operatora aplicira se na početku svakog idućeg retka.

Mijenjanjem veličine zaslona ili izmjenom veličine fonta sadržaj se ne prilagođava automatski zbog toga što je potrebno da MathJax ponovno renderira SVG prikaz formule. Nakon što se prozor osvježi, formula se opet optimizira. Ovo je rješenje stoga problematično jer korisnik ne zna za potrebu ponovnog učitavanja sadržaja radi optimizacije prikaza.

Pristupačnost samog sadržaja putem TTS mehanizma omogućena je jer, iako MathJax matematički sadržaj pretvara u jezik SVG, također zadržava izvorni MathML zapis u kodu koji je dostupan mehanizmima mehaničkog iščitavanja.

## 9.5. Eksperimentalno rješenje za IDEAL Group Reader na platformi Android i e-čitač Calibre na platformi Windows i OS X

IDEAL Group Reader jest e-čitač namijenjen za platformu Android. Njegove su glavne karakteristike da podržava EPUB 3 format e-knjiga, omogućuje mehaničko iščitavanje teksta, vizualno naglašava tekst koji se iščitava, prikazuje i glasovno iščitava MathML sadržaj, znatno povećava pristupačnost sadržaja, omogućuje zapisivanje bilješki, podržava video i audio sadržaje unutar EPUB e-knjiga te podržava interaktivnost [62]. Najveća je prednost ovog e-čitača da omogućuje iščitavanje MathML sadržaja, što povećava pristupačnost ove vrste sadržaja slijepim i slabovidnim osobama. IDEAL Group Reader testiran je na tabličnom računalu Samsung Galaxy Tab 2 i platformi Android.



Slika 29. Prikaz matematičke formule na e-čitaču IDEAL Reader Group na tabličnom računalu Samsung Galaxy Tab 2 prije intervencije skriptnim jezikom

Nakon otvaranja EPUB e-knjige matematičkog sadržaja na tabličnom računaru Samsung Galaxy vidljivo je da se matematički znakovi prikazuju, što znači da je jezik MathML podržan, ali da se matematička formula ne prelama s obzirom na dostupnu širinu zaslona odnosno prozora aplikacije. Rješenje stoga podrazumijeva intervenciju koja će prisiliti matematičku formulu na prijelom i apliciranje tipografske uvlake radi razumijevanja sadržaja. Prijelom se realizira konfiguriranjem MathJax knjižnice na početku XHTML dokumenta.

```
<script type = 'text/javascript'>
window.MathJax = {
  "HTML-CSS": { linebreaks: { automatic: true, width: "70%" },
                displayAlign: "left"
              }
};
</script>
```

Slika 30. Prikaz konfiguracije MathJax knjižnice kod izlaznog prikaza „HTML-CSS”

MathJax knjižnica nudi opcije konfiguriranja prijeloma matematičkog sadržaja s unaprijed definiranim naredbama. IDEAL Group Reader primjenjuje izlazni prikaz HTML-CSS za jezik MathML, što je i primarni izlazni prikaz MathJax knjižnice. Proizvodi prikaz visoke kvalitete, koristi se mrežnim fontovima, ali zahtijeva više vremena za renderiranje od nativnog prikaza. Objekt `linebreaks` kontrolira ponašanje izlaznog procesora HTML-CSS čija je početna vrijednost `false`. Automatsko se prelamanje stoga mora uključiti postavljanjem na vrijednost `true`. Širina se postavlja na 70 % širine kontejner elementa kako se sadržaj nakon apliciranja uvlake ne bi prikazivao izvan zaslona. Nakon konfiguracije prijeloma redaka, međutim, sadržaj se ne prikazuje zbog redosljeda učitavanja. Naime, prilikom učitavanja dokumenta e-čitač prepozna jezik MathML te automatski najprije učitava MathJax knjižnicu, a tek zatim njegovu konfiguraciju prijeloma redaka. Zbog toga je potrebno naknadno učitati MathJax knjižnicu jer je potrebno najprije konfigurirati izlaz, a tek zatim knjižnicu, kako bi se sadržaj prikazao i prelomio:

```

<script type="text/javascript"
src="http://cdn.mathjax.org/mathjax/latest/MathJax.js?config
=MML_HTMLorMML">
</script>

```

Naknadno učitavanje MathJax knjižnice usporava rad e-čitača jer se prilikom otvaranja svake stranice mora učitati cijela MathJax knjižnica. Postoji mogućnost optimiziranja smanjivanjem knjižnice na nužne dijelove, ali ta je optimizacija izvan istraživanja ovog rada.

Zatim se pristupa rješavanju uvlake funkcijom JavaScript. Ponovno je potrebno upoznati se sa specifičnom DOM strukturom ovog izlaznog modela.

```

▼<span class="math" id="MathJax-Span-1" role="math" style="width: 48.561em; display: inline-block;">
▼<span style="display: inline-block; position: relative; width: 46.698em; height: 0px; font-size: 104%;">
▼<span style="position: absolute; clip: rect(6.253em 1046.7em 13.404em -999.997em); top: -7.809em; left: 0.003em;">
▼<span class="mrow" id="MathJax-Span-2">
▼<span style="display: inline-block; position: relative; width: 46.698em; height: 0px;">
▼<span style="position: absolute; clip: rect(2.587em 1046.7em 6.133em -999.997em); top: -4.144em; left: 0.003em;">
▼<span class="mrow" id="MathJax-Span-3">
▶<span class="mstyle" id="MathJax-Span-4">...</span>
▶<span class="mrow" id="MathJax-Span-9" style="padding-left: 0.183em;">...</span>
▶<span class="mi" id="MathJax-Span-20" style="font-family: STIXGeneral-Italic; padding-left: 0.183em;">+</span>
  <span class="mi" id="MathJax-Span-21" style="font-family: STIXGeneral-Italic;">A</span>
</span>
<span class="mo" id="firstOperator" style="font-family: STIXGeneral-Regular; padding-left: 0.303em;">=</span>
▶<span class="mrow" id="MathJax-Span-23" style="padding-left: 0.303em;">...</span>
<span class="mo" id="MathJax-Span-45" style="font-family: STIXGeneral-Regular; padding-left: 0.243em;">+</span>
▶<span class="mrow" id="MathJax-Span-46" style="padding-left: 0.243em;">...</span>
<span class="mo" id="MathJax-Span-68" style="font-family: STIXGeneral-Regular; padding-left: 0.303em;">=</span>
▶<span class="mrow" id="MathJax-Span-69" style="padding-left: 0.303em;">...</span>
<span class="mo" id="MathJax-Span-89" style="font-family: STIXGeneral-Regular; padding-left: 0.243em;">+</span>
▶<span class="mrow" id="MathJax-Span-90" style="padding-left: 0.243em;">...</span>
<span class="mo" id="MathJax-Span-115" style="font-family: STIXGeneral-Regular; padding-left: 0.303em;">=</span>
▶<span class="mrow" id="MathJax-Span-116" style="padding-left: 0.303em;">...</span>
<span style="display: inline-block; width: 0px; height: 4.15em;"></span>
▶<span style="position: absolute; clip: rect(2.587em 1020.38em 6.073em -999.997em); top: -0.478em; left: 0.003em;">...</span>
</span>

```

Slika 31. Prikaz DOM strukture matematičke formule uz pomoć MathJax knjižnice s izlazom HTML-CSS

Najprije je potrebno dohvatiti prvi operator u matematičkoj formuli te očitati njegovu udaljenost od lijevog ruba zaslona.



```

var uvlaka = function()
{

    var mat = window.document.getElementsByClassName('math');
    var firstOperator = mat[0].getElementsByTagName('span')[3].childNodes;
    var spanovi = [];

    for (var i = 0; i < firstOperator[0].childNodes.length ; i++) {
        var child = firstOperator[0].childNodes[i];
        if(child.className == 'mo'){
            spanovi.push(child);
        }
    };
    var x = spanovi[0].offsetLeft;

    var drugiRedovi = mat[0].getElementsByTagName('span')[3].childNodes;
    var polje =[];
    for (var j = 1; j < drugiRedovi.length; j++){

        var dete = drugiRedovi[j];
        polje.push(dete);
    }

    polje[0].firstChild.setAttribute('style', 'margin-left:' + x + 'px');

    if (polje[1])
        {polje[1].firstChild.setAttribute('style', 'margin-left:' + x + 'px');};

};

```

Slika 32. Prikaz funkcije JavaScript za apliciranje uvlake na platformi Android

DOM se u ovom slučaju razlikuje od onoga kreiranog u Radiumu zbog drugačijeg izlaznog modela. Matematički se sadržaj nalazi u elementu klase `math` koji se dohvaća naredbom `getElementsByClassName` i sprema u polje. Elementi koji su grupirani oznakama `<math>` u MathML-u se u izlazu HTML-CSS nalaze u elementima `<span>`. Prvi se operator dohvaća sakupljanjem elemenata `<span>` prve grupacije matematičke formule i njihove djece elemenata:

```

var firstOperator =
mat[0].getElementsByTagName('span')[3].childNodes;

```

Nakon dohvaćanja adekvatnih elemenata `<span>`, uz pomoć petlje `for` pretražuje se polje i provjerava ime klase elementa. Ako klasa nosi ime `mo`, sprema se u varijablu.

Udaljenost se dohvaća metodom `offsetLeft`:

```

for (var i = 0; i < firstOperator[0].childNodes.length ; i++) {
    var child = firstOperator[0].childNodes[i];
    if(child.className == 'mo'){
        spanovi.push(child);
    }
};
var x = spanovi[0].offsetLeft;

```

Slika 33. Prikaz dohvaćanja udaljenosti prvog operatora za IDEAL Group Reader

Varijabla `x` sada sadrži udaljenost prvog operatora od lijevog ruba zaslona. Zatim se dohvaćaju redovi koji su prelomljeni kako bi se na njih mogla aplicirati uvlaka. Uvlaka se aplicira JavaScript intervencijom u CSS svojstvo `margin-left`:

```

polje[0].firstChild.setAttribute
('style', 'margin-left:' + x + 'px');

```

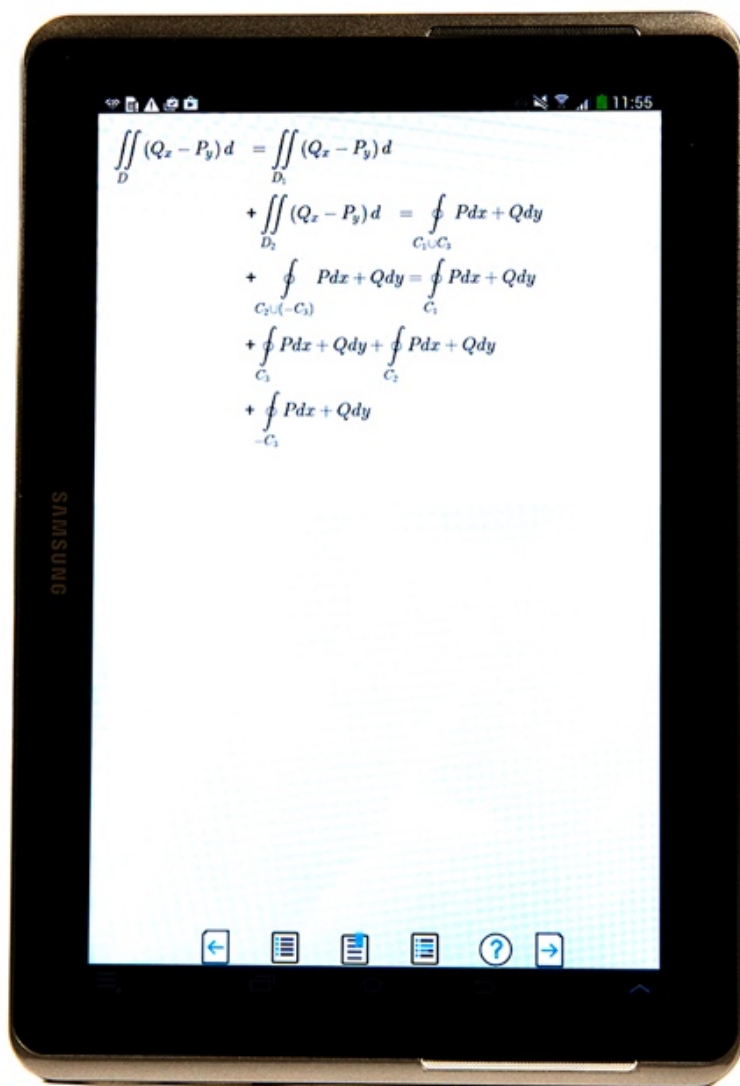
Ako postoji više redova, na njih se također aplicira lijeva margina iste vrijednosti:

```

if (polje[1])
{polje[1].firstChild.setAttribute
('style', 'margin-left:' + x + 'px');

```

Interveniranjem u konfiguraciju MathJax knjižnice i modificiranom funkcijom JavaScript postignuto je zamišljeno rješenje za prijelom kompleksne matematičke formule u EPUB e-knjizi na platformi Android. Ovo se rješenje također može aplicirati na bilo kojem e-čitaču koji se koristi MathJax knjižnicom s izlazom HTML-CSS. Ova je tvrdnja provjerena i dokazana na e-čitaču Calibre na platformi Windows.



Slika 34. Prikaz matematičke formule u e-čitaču IDEAL Group Reader na platformi Android nakon intervencije konfiguracijom MathJax knjižnice i funkcije JavaScript

Nakon intervencije, matematički je sadržaj dostupan korisniku vizualno i prema tipografskim pravilima struke, ali je i dostupan putem TTS tehnologije jer je zadržan izvorni MathML kod koji ovaj e-čitač ima funkciju mehanički iščitati. Prilikom mijenjanja veličine zaslona nakon što je stranica knjige učitana sadržaj se neće prilagoditi bez osvježavanja prikaza, ali to u ovom slučaju ne predstavlja problem jer korisnici rijetko mijenjaju orijentaciju uređaja prilikom čitanja e-knjiga, a na mobilnim uređajima je to jedini način mijenjanja veličine zaslona.

## 10. EKSPERIMENTALNI REZULTATI I ANALIZA

Semantički ispravno jezikom MathML oblikovana kompleksna matematička formula ugrađena je u EPUB 3 e-knjigu i testirana na e-čitačima različitih mehanizama za prikaz mrežnog sadržaja na različitim platformama. Ukupno je testirano šest e-čitača na operativnim sustavima Windows, Mac OS X, iOS i Android. Eksperimenti su se vodili u uvjetima ugrađenog (nativnog) prikaza jezika MathML, ali i putem MathJax JavaScript knjižnice. Svojstva koja su se testirala su: podrška jezika MathML koja opisuje prikazuje li se sadržaj oblikovan jezikom MathML na e-čitaču; izlazni prikaz koji uvjetuje pristupačnost takvog sadržaja; pristupačnost u vidu dostupnosti sadržaja TTS mehanizmima te performanse ugrađenih TTS mehanizama pri iščitavanju matematičkog sadržaja i mogućnosti povećavanja veličine sadržaja; pretraživost u obliku kopiranja određenih dijelova matematičke jednadžbe te izvođenja pretrage kopiranih znakova ugrađenim mehanizmom za pretragu; tipografski prijelom koji optimalno za čitljivost i razumijevanje prelama matematičke formule te dinamički prijelom koji aktivno prilagođava prijelom sadržaja dostupnoj veličini zaslona ili prozora.

## 10.1. Analiza podrške MathML-a

Eksperimentalnom metodom izrade modela EPUB e-knjige matematičkog sadržaja utvrdilo se da svi testirani e-čitači prikazuju MathML sadržaj unutar EPUB 3 e-knjige. Suprotno pretpostavci, nije potrebno ugraditi specijalni font koji sadrži sve matematičke simbole. Izlazni prikaz odnosno renderiranje sadržaja ovisi o e-čitaču koji se upotrebljava. E-čitači upotrijebljeni za testiranje izrađenog modela na različite načine renderiraju jezik MathML ovisno od platformi odnosno mehanizmu za prikaz mrežnog sadržaja. Stoga se prema renderiranju matematičkog sadržaja e-čitači mogu podijeliti u dvije osnovne grupacije: oni koji jezik MathML prikazuju nativno i putem MathJax JavaScript knjižnice. Nativni prikaz omogućuju mehanizmi za prikaz mrežnog sadržaja Gecko (Firefox) i WebKit (Safari). Mehanizmi za prikaz mrežnog sadržaja koji ne prikazuju jezik MathML nativno već se koriste MathJax knjižnicom za prikaz jesu Blink (Chrome) i QT WebKit (Calibre). E-čitači koji MathML prikazuju nativno kao izlazni prikaz imaju MathML, međutim to ne unificira specifične probleme prijeloma matematičkog sadržaja.

## 10.2. Analiza metode kreiranja semantički pravilnoga matematičkog sloga

Prilikom kreiranja MathML sadržaja važan je odabir softverskog alata. Softverski alat utječe na semantičku korektnost sadržaja, što utječe na kvalitetu konverzije u druge formate, na način na koji će TTS mehanizmi čitati sadržaj i grupiranje sadržaja. Pri kompleksnim matematičkim izrazima utvrđena je nužnost grupiranja dijelova matematičkih formula. Grupiranje se izvršava prilikom oblikovanja matematičkih formula u jeziku MathML uz pomoć elemenata `<mathrow>` i omogućuje prijelom matematičke formule na tipografski optimalnim mjestima. Utvrđeno je da je na svim e-čitačima na kojima je otvoren model e-knjige čiji je matematički sadržaj grupiran elementima `<mathrow>` ispunjen cilj prijeloma na tipografski optimalnim mjestima, odnosno na mjestima operatora ili jednakosti. Pokazalo se da, iako jezik MathML sadrži attribute potrebne za detaljnu i specifičnu kontrolu prijeloma sadržaja, ti atributi još uvijek nisu podržani ni kod jednog e-čitača te se ne mogu primjenjivati u praksi. Eksperimentalnom je metodom utvrđeno da svi testirani e-čitači, bez obzira na mehanizam prikazivanja, prikazuju sve potrebne matematičke znakove, iako se u oblikovanom modelu e-knjige ne primjenjuje za tu svrhu kreiran font STIX. Ako bi postojala potreba za ugradnjom fonta STIX u direktorij EPUB-a, to bi znatno povećalo njegovu veličinu, posebno zbog potrebe ugradnje kompletne familije pisma, ali i različitih formata fonta nužnih za funkcioniranje na različitim platformama.

### 10.3. Dinamičko traženje točke prijeloma u matematičkom slogu

Dinamički prijelom podrazumijeva prilagođavanje matematičkog sadržaja različitim zaslonima, odnosno različitim veličinama dostupnih prozora e-čitača. Prilikom učitavanja modela e-knjige primijećena su tri načina prikaza kompleksnoga matematičkog sadržaja s obzirom na prijelom. Prvi, optimalan način, s obzirom na podršku jezika MathML, prikazuje EPUBReader. EPUBReader i Readium prikazuju matematički izraz cjelovito, prelamaju formulu ovisno o veličini dostupnog prozora na tipografski optimalnim mjestima, bez stilske ili skriptne intervencije. E-čitač Calibre također prelama matematički sadržaj, ali na proizvoljnim mjestima, bez obzira na grupirani sadržaj te ne zadovoljava tipografski preduvjet čitljivosti ovog tipa sadržaja. Drugi način prikaza jest da se matematički sadržaj ne prelama, prikazan je do ruba prozora e-čitača. Takav sadržaj može i ne mora biti dostupan korisniku. U slučaju e-čitača iBooks i Adobe Digital Editions sadržaj koji se nalazi izvan prozora uopće nije vizualno dostupan čitatelju. Sadržaj je na platformi Mac OS postao dostupan CSS intervencijom zbog Safarijeve implementacije modela prikazivanja *flexbox*. Treći način prikaza, viđen na e-čitaču IDEAL Group Reader, jest da se sadržaj također ne prelama, ali je dostupan klizanjem (eng. *scroll-pan*) udesno. Problem prijeloma u ovom je slučaju riješen konfiguriranjem MathJax knjižnice. Prijelom matematičkog sadržaja koji se renderira putem MathJax knjižnice u primjerima može biti standardno uključen (Readium i Calibre) ili isključen (IDEAL Group Reader). Osnovni preduvjet korisnosti e-knjige jest da njezin sadržaj, bez obzira na to je li tekstualni, matematički, slikovni itd., bude vizualno dostupan. Problem nedostatka prelamanja matematičkog sadržaja odnosno vizualne nedostupnosti matematičkog sadržaja u e-knjigama riješen je intervencijom jezikom CSS ili konfiguriranjem MathJax knjižnice.

#### 10.4. Oblikovanje dinamičkoga matematičkoga prijeloma

Uvjet važan za čitljivost i razumijevanje kompleksnog matematičkoga sadržaja jest i pravilno tipografsko oblikovanje koje, osim prijeloma na adekvatnim mjestima, podrazumijeva i apliciranje uvlake u veličini prostora do prvog znaka jednakosti ili prvog važnog znaka matematičke operacije. Jezik MathML sadrži atribute za ovu finu kontrolu prijeloma, ali nijedan e-čitač za sada ne pruža podršku tim mogućnostima. Samim time slijedi da ni u jednom e-čitaču nije automatski aplicirana takva uvlaka.

Kontrolu do kojeg se znaka mjeri udaljenost ima kreator matematičkoga sadržaja, a takvu odluku vjerojatno nikada neće moći donijeti nijedan softver, već će uvijek ovisiti o specifičnom primjeru. Manipulacija te udaljenosti u primjeru se vršila postavljanjem tog označnog elementa slobodnim, odnosno prvim djetetom glavnog elementa `<math>`, okruženog ostalim elementima `<mathrow>`. Također mu se i dodijelio unikatni imenovan atribut `id` kako bi se u određenim situacijama mogao dohvatiti. Ovakva intervencija zahtijeva od kreatora sadržaja osnovno poznavanje sintakse i strukture jezika MathML. Dohvaćanje udaljenosti u svim se rješenjima vrši uz pomoć skriptnog jezika JavaScript. Određeni e-čitači nemaju standardno uključenu opciju JavaScripta, kao što je u primjeru EPUBReader, što može umanjiti korisnost ovog pristupa rješavanju problema uvlake. Funkcija kreirana za apliciranje uvlake za sve e-čitače ima identičan algoritam:

1. dohvati element do kojeg se mjeri udaljenost
2. izmjeri udaljenost od lijevog ruba prozora do dohvaćenog znaka
3. dohvati drugi red/redove matematičkoga izraza
4. apliciraj uvlaku dohvaćene udaljenosti.

Kreiranje funkcije ovisi o mehanizmu za prikaz mrežnog sadržaja, specifičnije o: podržanim metodama, DOM-u i izlaznom prikazu. U slučaju nativnog prikaza MathML-a manipulacija elementima jednostavnija je i izravnija jer se Document Object Model izgledom poklapa s originalnim kodom jezika MathML. Za nativni prikaz koji se pronalazi u e-čitačima EPUBReader, iBooks i Adobe Digital Editions (Mac OS) potrebna je prilagodba funkcije izmjenom JavaScript metode dohvaćanja udaljenosti prvog operatora zbog različite podržanosti u mehanizmima Gecko i WebKit. U slučajevima prikaza putem MathJax knjižnice funkcija ovisi o izlaznom prikazu te sukladno kreiranom



DOM-u. Tada je potrebno proučiti kreirani DOM i prilagoditi algoritam. Prvi se operator u ovim slučajevima najjednostavnije dohvaća putem postavljenog atributa `id`, što nije nužno u primjerima nativnog prikaza MathML-a. Radium je jedini testirani e-čitač koji se koristi izlaznim prikazom SVG putem MathJax knjižnice i zbog toga je potrebno upotrebljavati regularne izraze kako bi se dohvatila udaljenost prvog operatora i SVG attribute prilikom apliciranja uvlake. IDEAL Group Reader koristi se izlaznim prikazom HTML putem MathJax knjižnice pa u ovom slučaju nema potrebe za primjenom atributa `id` ni regularnih izraza, udaljenost se dohvaća identičnom JavaScript metodom kao i kod mehanizma WebKit za mrežni prikaz, a uvlaka se aplicira putem CSS svojstva.

Nakon što je riješen problem dinamičkog i tipografskog prijeloma, uočeno je da se u e-čitačima koji se koriste MathJax knjižnicom za prikaz matematičkog sadržaja aktivno prilagođavanje sadržaja prilikom izmjene veličine dostupnog prozora (primjerice okretanjem orijentacije uređaja) zadane izmjene neće realizirati nakon što se učitala stranica s matematičkom formulom i primijenila funkcija JavaScript. Na tim je e-čitačima potrebno ponovno osvježiti stranicu te prisiliti uređaj da ponovno renderira sadržaj kako bi se dogodile željene izmjene na sadržaju. Na e-čitačima koji nativno prikazuju MathML osvježavanje nije potrebno, već se prilagođavanje događa u realnom vremenu.

## 10.5. Pristupačnost matematičkog sloga u e-knjigama

Pristupačnost matematičkog sadržaja u e-knjigama testirala se kroz četiri definirana principa pristupačnosti smjernica *Web Content Accessibility Guidelines*:

1. primjetljiv
2. operabilan
3. razumljiv
4. robustan.

Prema prvom principu koji nalaže da su informacije primjetljive testirana je promjena veličine matematičkih znakova i simbola u različitim e-čitačima. Eksperimentalnom je metodom utvrđeno da se na svim testiranim e-čitačima matematički sadržaj odnosno veličina fonta može povećavati putem za to predviđenog izbornika standardiziranog u svim e-čitačima. Nadalje, s obzirom na to da je matematički sadržaj oblikovan uz pomoć označnog jezika MathML, a ne putem *bitmap* slika, dostupan je tehnologijama *text-to-speech*, ali i procesiranju u druge jezike poput Braillea ili LaTeXa. Dostupni su različiti softverski programi i internetske aplikacije čija je namjena konvertiranje jezika MathML u neku drugu vrstu zapisa. Iako MathJax knjižnica ima izlazni prikaz u obliku HTML-a ili SVG-a i time mijenja strukturu DOM-a, originalni MathML zapis uvijek je zadržan u svojem izvornom obliku u svim e-čitačima.

Drugi princip operabilnost nalaže da se sadržajem može navigirati i da je pretraživ. Navigacija je unutar sustava e-čitača osigurana putem navigacijskog izbornika, dok je lokalno navigiranje osigurano zbog obgrljenosti matematičkog sadržaja elementom `<math>` pa se takav sadržaj u slučaju slušanja može jednostavno pronaći ili preskočiti. Pretraživost se testirala kopiranjem specifičnih dijelova matematičke formule te pretragom putem za to predviđenog alata unutar e-čitača. S obzirom na to da je sadržaj oblikovan označnim jezikom, autorski se maksimalno osigurala ova mogućnost, međutim e-čitači još uvijek uglavnom ne pružaju podršku za ovu mogućnost. Jedino e-čitač Calibre dozvoljava kopiranje i pretraživanje posebnih matematičkih simbola, poput simbola za integral. EPUBReader dozvoljava pretragu svih ostalih znakova, ali ne i simbola za integral, dok ostali e-čitači ne dozvoljavaju ni selekciju dijelova formule.

Treći je princip razumljivosti i čitljivosti sadržaja dokazan u svim e-čitačima putem dinamičkog i tipografskog prijeloma osiguranog putem CSS i JavaScript intervencije. Kako bi matematičke formule bile razumljive i čitljive, potrebno je osigurati njihov prikaz u cijelosti te optimizirati njihov prijelom prema standardima tipografske struke.

Četvrti princip robusnosti testiran je provjerom kompatibilnosti modela e-knjige na različitim uređajima. Model e-knjige izrađen je prema trenutnim standardima EPUB 3 koji podrazumijevaju korištenje označnim jezikom MathML prilikom oblikovanja matematičkog sadržaja. Utvrđeno je da je nakon intervencije CSS-om i funkcijom JavaScript sadržaj kompatibilan na svim testiranim e-čitačima. Dostupnosti sadržaja TTS tehnologijama pokazala se djelomično ispunjenom. E-čitači koji imaju ugrađeni TTS mehanizam iščitavaju sadržaj, ali još uvijek nisu u svojoj standardnoj verziji osposobljeni za čitanje matematičkog sadržaja. Readium jedini uopće ne iščitava sadržaj, dok ostali iščitavaju, ali ne tako da to čitatelj može razumjeti (bez čitanja znaka za sumu, jednakost, integral i ostalih specifičnih znakova). Za iščitavanje specijalnoga matematičkog sadržaja potrebno je instalirati posebne softvere koji su tome namijenjeni, koji prilikom izrade ovog rada nisu bili dostupni autoru. Međutim, ako specifikacija softvera tvrdi da iščitava MathML sadržaj, model ove e-knjige sigurno je kompatibilan.

Tablica 4. Eksperimentalni rezultati usporedbe e-čitača prema definiranim parametrima optimizacije matematičkog sloga u e-knjigama

e-čitač	Platforma	Podrška MathML-a	Izlazni prikaz	Pristupačnost	Pretraživost	Tipografski prijelom	Dinamički prijelom
EPUBReader	PC/Mac	nativno (Gecko)	MathML	da	djelomično	da	da
iBooks	Mac / iOS	nativno (Mobile Safari)	MathML	djelomično	ne	da	da
Readium	PC/Mac	MathJax	SVG	djelomično	ne	da	djelomično
Adobe Digital Editions	Mac	nativno (Safari)	MathML	djelomično	ne	da	da
IDEAL Group Reader	Android	MathJax	HTML	da	ne	da	djelomično
Calibre	PC/Mac	MathJax	SVG/HTML	djelomično	da	da	djelomično

## 11. ZAKLJUČAK

Postavljeni uvjet kreiranja optimalnoga matematičkog sloga u e-knjigama podrazumijeva njegovu semantičku ispravnost, tečnost i prijelom prema tipografskim standardima matematičke struke. Procesu kreiranja optimalnoga matematičkog sadržaja za e-knjige pristupilo se najprije analizom i usporedbom formata e-knjiga (2) te definiranjem vrijednosti parametara koje je potrebno ispuniti. Utvrđeno je da svi danas prisutni formati potječu iz ranijeg standarda OEBPS (Open eBook Publication Structure), ali njihov razvoj teče u drugačijim smjerovima, prema ciljevima pojedine tvrtke koja njima upravlja. Prvi uvjet koji je bilo potrebno zadovoljiti bio je analizom naći optimalni format otvorenog tipa zbog transparentnosti kreiranja koda te pristupa istome.

Prvo se je pristupilo analizi formata Mobipocket (2.1) koji je sličan formatu EPUB koji Amazon otkupljuje i drži zatvorenog, sadržaj prikazuje tečno, ali isključivo podržava prikaz teksta i slika. Moguće ga je konvertirati u druge formate s nekontroliranim rezultatima. U ovome formatu nije se moguće koristiti matematičkim označnim jezikom, što onemogućuje korištenje pomoćnim tehnologijama. Sljedeći format u analizi, AZW (Amazon Kindle book format) (2.2), identičan je formatu Mobipocket s dodatkom enkripcije u svrhu zaštite od piratstva. Format KF8 (Kindle Format 8) vlasnički je, zatvoren i tečan format tvrtke Amazon, nema mogućnost zvučne reprodukcije sadržaja niti podržava matematički sadržaj u obliku označnog jezika MathML (Mathematical Markup Language) koji bi osigurao semantičku točnost. Konverzije su moguće u druge formate također uz tipografski i semantički nepredvidive rezultate. U formatima AZW i KF8 oblikovanje sadržaja tipografski je izrazito limitirano. Format EPUB (2.4, 2.5, 2.6) standard je industrije koji je otvoren, tečan, dostupan pomoćnim tehnologijama, lako se konvertira u druge formate, osnovan na otvorenim mrežnim tehnologijama i XML-u (Extensible Markup Language), tekst se oblikuje putem CSS-a (Cascading Style Sheets), podržava matematički označni jezik i multimediju. Format iBooks također je vlasnički format tvrtke Apple s enkripcijskom zaštitom. Fluidan je, podržava TTS tehnologije (*text-to-speech*) i matematički slog, ali je čitanje moguće isključivo na Appleovim uređajima, a distribucija putem Appleova dućana. Format PDF (Portable Document Format) (2.8) ne

može se primjenjivati za kreiranje tečnoga matematičkoga sloga zbog toga što nije fluidan te time onemogućuje čitanje na malim zaslonima.

Analizom je potvrđeno da je uvjet pristupačkoga matematičkoga sadržaja njegova semantička točnost koja omogućuje korištenje pomoćnim tehnologijama. Sadržaj u obliku slike ne smatra se pristupačnim. Utvrdilo se da je EPUB 3 najpristupačniji format (2.9). Analizom pristupačnosti e-čitača (3) zaključilo se da će, bez obzira na način kreiranja sadržaja ili formata, prikaz i pristupačnost ovisiti o odabranom uređaju. Analizom načina kreiranja različitih formata (4) zaključeno je da je zatvorene formate znatno jednostavnije i brže proizvesti zbog softverskih alata koje dostavljaju tvrtke vlasnice, ali bez kontrole semantike. Za semantički optimalan sadržaj potrebno je „ručno“ kodiranje koje je dugotrajnije, ali precizno.

Pri uspoređivanju formata e-knjiga (5) definirali su se parametri koji se prate: tečnost, podržanost, otvorenost, izrada, prijelom i ažuriranje matematičkoga sadržaja. Tečnost podrazumijeva prilagodbu sadržaja veličini sadržaja, rezoluciji i orijentaciji zaslona te je kao takva obvezna za optimalan matematički slog. Podržanost se odnosi na količinu prisutnih alata za čitanje, a otvorenost na povezanost formata s nekim sustavom, njegovu konverziju i arhiviranje. Vlasnički su formati zatvoreni, vezani za određeni alat ili platformu te im je arhiviranje onemogućeno zahvaljujući DRM (Digital Rights Management) shemi. Izrada u slučaju semantički točno kreiranog sadržaja za sada zahtijeva znatno dulje vrijeme i znanje nego kod vlasničkih formata. Jedini format koji omogućuje kreaciju tečnog i semantičkoga matematičkoga sadržaja koji je otvoren, pristupačan i podržan jest EPUB 3. S obzirom na trenutačnu neprisutnost adekvatnih softverskih alata za brzo i jednostavno kreiranje EPUB 3 e-knjige kreiranje je opisano u šestom poglavlju. Opisani su obvezni konstruktori EPUB e-knjige: dokument *package* (6.1), metapodaci (6.2), elementi za popis izvora publikacije i slijed čitanja (6.3), navigacijski dokument i proces obveznog validiranja (6.5).

Usporedbom popularnih softverskih alata za kreiranje EPUB e-knjiga (6.6) utvrđeno je da nijedan alat ne pruža optimalan način kreiranja EPUB e-knjige semantički korektnoga matematičkoga sadržaja te da se određeni alati mogu primjenjivati kao pomoć, ali je „ručno“ kodiranje neizbježan dio procesa izrade.

U sedmom poglavlju postavljeni su problemi ugrađivanja matematičkog sadržaja u EPUB e-knjigu, koji osim što uključuju specifičnu tipografsku prezentaciju koja omogućuje razumijevanje, uključuju i pristupačnost sadržaja, kopiranje, pretraživanje, procesiranje i prebacivanje u druge formate (matematičke i knjižne). Ustanovljeno je da ugradnjom matematičkog sadržaja u obliku vektorske ili *bitmap* slike (7.1) sadržaj ne zadovoljava uvjete optimalnoga matematičkog sloga. Zbog pristupačnosti matematičkog sadržaja potrebno je posvetiti pažnju prilikom strukturiranja podataka (7.2). Nakon što je utvrđeno da je MathML optimalan način ugradnje matematičkog sadržaja u e-knjige, utvrđen je proces izrade i oblikovanja (8) te se ustanovio i optimalan softverski alat za taj zadatak (8.1). Potvrđena je hipoteza da je matematički označni jezik uvjet za pristupačnost matematičkog sadržaja u e-knjigama te da u drugim opcijama nije moguća pretraživost ili dostupnost pomoćnih tehnologija (8.2). Komparacijom dostupnih primjera s tržišta koji ugrađuju matematički sadržaj u obliku *bitmap* slika, semantički netočno opisanoga matematičkog sadržaja ili uz pomoć elementa `<switch>` utvrđeno je da nijedna od te tri dostupne metode ne udovoljava uvjetima optimalnoga matematičkog sloga. Dokazana je hipoteza da optimalni matematički slog može oblikovati isključivo označnim jezikom MathML.

U eksperimentalnom dijelu rada odabrano je šest e-čitača na različitim operativnim sustavima koji se koriste različitim mehanizmima za prikazivanje mrežnog sadržaja za testiranje EPUB e-knjige s MathML sadržajem. Dokazano je da dinamički matematički slog ne ovisi samo o odabranome formatu e-knjige i načinu ugrađivanja matematičkog sloga već i o odabranome e-čitaču u kojem se e-knjiga renderira. Pokazana su rješenja za pojedine e-čitače uz pomoć jezika CSS čija je intervencija bila nužna kod e-čitača iBooks (9.2) i JavaScript funkcija za apliciranje tipografske uvlake. U slučajevima primjene MathJax knjižnice za prikaz jezika MathML bila je nužna intervencija konfiguracije MathJax knjižnice u cilju prelamanja matematičkog sloga (9.4). Dokazana je tvrdnja da bez intervencije skriptnim jezikom JavaScript apliciranje tipografske uvlake koja osigurava čitljivost pri dinamičkom prijelomu nije moguće.

U rezultatima i analizi eksperimentalnog dijela (10) uspoređena su testirana svojstva podrške prikaza jezika MathML te izlazni prikaz koji uvjetuje pristupačnost, pretraživost, tipografski prijelom i dinamički prijelom u svrhu kategorizacije testiranih e-čitača.

Eksperimentalno je utvrđena važnost semantički korektno označenoga matematičkog sadržaja (10.2) na koju utječe alat izrade. Dokazano je da semantika utječe na grupiranje sadržaja koje osigurava korektan tipografski prijelom i način na koji TTS mehanizmi čitaju sadržaj. Svi odabrani e-čitači pružaju podršku jezika MathML nativno ili putem MathJax knjižnice. Dinamički prijelom matematičkog sadržaja (10.3) koji podrazumijeva prilagođavanje matematičkog sadržaja veličini zaslona ili prozora e-čitača postignut je na tri načina: automatski u e-čitaču, putem CSS-a ili konfiguracijom MathJax knjižnice. Tipografski prijelom (10.4) ključan je element pri razumijevanju kompleksnih matematičkih izraza i nije ga moguće ostvariti jezikom MathML koji sadrži sve potrebne elemente i atribute u tu svrhu, već je postignut kreiranjem funkcije JavaScript zasebno za svaki e-čitač. Eksperimentalnom metodom dokazana je hipoteza da se dinamički matematički slog oblikovan prema tipografskim pravilima može postići primjenom označnog jezika MathML i JavaScript intervencije. Pristupačnost matematičkog sadržaja testirala se kroz četiri osnovna principa pristupačnosti (10.5). Na kraju je tablično prikazana kategorizacija testiranih e-čitača.

Ovom disertacijom ukazano je na važnost i potrebu postojanja tipografski oblikovanoga matematičkog sadržaja unutar e-knjiga koji je dostupan pomoćnim tehnologijama. Usprkos postojećim specifikacijama, tehnološki razvitak e-čitača još uvijek zaostaje u podršci, djelomično zbog monopola velikih tvrtki koje upravljaju tržištem. Jednako kako internetski preglednici koče razvitak *web* dizajna i prisiljavaju dizajnere na kreiranje trenutačnih rješenja, e-čitači koče puni potencijal pristupačnih i tipografski oblikovanih e-knjiga. U ovoj se disertaciji rješavaju trenutačni problemi matematičkog sloga u e-knjigama čija rješenja neće funkcionirati dugoročno, ali dokazuju da su već sada moguća i potrebna. Tehnologija e-čitača ubrzano se razvija, a format EPUB sve je priznatiji kao standard struke. Pristupačnost sadržaja nije samo trenutačno popularan pojam već tehnološki prioritet i obveza kreatora sadržaja. Potrebno je pravilnom upotrebom postojećih specifikacija zahtijevati podršku od kreatora e-čitača u cilju rješavanja problema pristupačnosti i standardnoga tipografskog oblikovanja.



## 12. LITERATURA

- [1] Warren, J. W. Innovation and Future of e-books. *The International Journal of the Book* 6(1), 2009, 83-94
- [2] De Meester, B. et al. A digital-first authoring environment for enriched e-Books using EPUB 3. *Information Services & Use* 34.3-4 (2014): 259-268
- [3] Kim, J., Hong, E. J., Jeon, G. I. Development of a Braille E-Book for the Blind using E-Book Standard. *Journal of KIISE: Computing Practices and Letters* 17.6 (2011): 369-377.
- [4] Ginev, D., Miller B. R., Oprea S. E-books and Graphics with LATEXml\*. *Intelligent Computer Mathematics: CICM 2014 Joint Events: Calculemus, DML, MKM, and Systems and Projects 2014, Coimbra, Portugal, July 7-11, 2014. Proceedings* 8543 (2014): 427.
- [5] Shotton, D. Semantic publishing: the coming revolution in scientific journal publishing. *Learned Publishing* 22.2 (2009): 85-94.
- [6] Garrish, M. *What is EPUB 3?*. O'Reilly Media, Inc., 2011.
- [7] Marinai, S., Marino, E. and Soda, G. Conversion of PDF books in ePub format. *2011 International Conference on Document Analysis and Recognition*. IEEE, 2011.
- [8] Choi, J., Lee, Y., Kim, K. An HTML5-based Interactive E-book Reader. *International Journal of Software Engineering and Its Applications* 8.2 (2014): 67-74.
- [9] Sandhu, P. *The MathML Handbook*, Charles River Media. Inc., Hingham, Massachusetts (2003).
- [10] Lee, K., Guttenberg, N. and McCrary V. Standardization aspects of eBook content formats. *Computer Standards & Interfaces* 24.3 (2002): 227-239.

- [11] Pinkney, A. J., Bagley, S. R. and Brailsford, D. F. No need to justify your choice: pre-compiling line breaks to improve eBook readability. *Proceedings of the 2013 ACM symposium on Document engineering*. ACM, 2013.
- [12] Tai, M. T., Lin, C. K., Ke, Z. R. and Chen, P. Y. A Study of E-Book Readability. *Applied Mechanics and Materials*. Vol. 262. Trans Tech Publications, 2013.
- [13] Rubens, P. *Science and technical writing: A manual of style*. Routledge, 2002.
- [14] Krantz, S. G. *Handbook of typography for the mathematical sciences*. CRC Press, 2000.
- [15] Marinai, S., Quiriconi, S. Displaying chemical structural formulae in ePub format. *Proceedings of the 2012 ACM symposium on Document engineering*. ACM, 2012.
- [16] Kleinfeld, S. *HTML5 for Publishers*. O'Reilly Media, Inc., 2011.
- [17] W3C – World Wide Web Consortium (2013), CSS Fonts Module 3, dostupno na: <http://www.w3.org/TR/css3-fonts/>, posijećeno: 15.08.2016.
- [18] STIX Font Project (2016) STIX Fonts, dostupno na: <http://www.stixfonts.org/>, posijećeno: 15.08.2016.
- [19] Park, B., Case study: how to use typography of digital publishing on the Smart Devices. *A Treatise on The Plastic Media 16.3* (2013): 99-108.
- [20] Pinkney, A. J., Bagley, S. R., Brailsford, D. F. Reflowable documents composed from pre-rendered atomic components. *Proceedings of the 11th ACM symposium on Document engineering*. ACM, 2011.

[21] Lin, X., Gao, L., Tang, Z., Lin, X., Hu, X. Mathematical formula identification in PDF documents. *2011 International Conference on Document Analysis and Recognition*. IEEE, 2011.

[22] Moon, H. S. The Production of Interactive App Books using Digital Publishing Technique. *Journal of Digital Design* 13.2 (2013): 441-449.

[23] Nakajima, T., Shinohara, S., Tamura, Y. Typical functions of e-Textbook, implementation, and compatibility verification with use of ePub3 materials. *Procedia Computer Science* 22 (2013): 1344-1353.

[24] Giannetti, F. Paginate dynamic and web content. *Proceedings of the 11th ACM symposium on Document engineering*. ACM, 2011.

[25] Junus, S. R. E-books and E-readers for Users with Print Disabilities. *Library Technology Reports* 48.7 (2012): 22-28.

[26] Mobipocket Developer Center (2008) What is Mobipocket fileformat?, dostupno na: <http://www.mobipocket.com/dev/article.asp?BaseFolder=prcgen&File=mobiformat.htm>, posijećeno: 15.08.2016.

[27] Bläsi, C., Rothlauf, F. On the interoperability of eBook formats. *Johannes Gutenberg-Universität Mainz–Germany*. <http://wi.bwl.uni-mainz.de/publikationen/InteroperabilityReportGutenbergfinal07052013.pdf> (2013).

[28] McIlroy, T. Ebook formats are a mess—here's why. *Learned Publishing* 25.4 (2012): 247-250.

[29] , International Digital Publishing Forum (2016) EPUB 3 Accessibility Guidelines - Text-to-Speech Overview, dostupno na: <https://idpf.github.io/a11y-guidelines/content/tts/overview.html>, posijećeno: 15.08.2016.

[30] Petri, K. Accessibility Issues in E-books and E-book Readers. *No shelf required 2* (2012): 35-60.

[31] Junus, S. R. E-books and E-readers for Users with Print Disabilities. *Library Technology Reports* 48.7 (2012): 22-28.

[32] Amazon Inc. (2016) Amazon Kindle Publishing Guidelines, dostupno na: <https://kindlegen.s3.amazonaws.com/AmazonKindlePublishingGuidelines.pdf>, posijećeno: 15.08.2016.

[33] International Digital Publishing Forum (2016) EPUB 3 Overview, dostupno na: <http://www.idpf.org/epub/30/spec/epub30-overview.html>, posijećeno: 15.08.2106.

[34] Garrish, M., Gylling, M. *EPUB 3 best practices*. O'Reilly Media, Inc., 2013.

[35] International Digital Publishing Forum (2016) EPUB 3 Accessibility Guidelines - The pub:type attribute, dostupno na: <https://idpf.github.io/a11y-guidelines/content/semantics/epub-type.html>, posijećeno: 15.08.2016.

[36] Garrish, M. *Accessible EPUB 3*. O'Reilly Media, Inc., 2012.

[37] International Digital Publishing Forum (2016) EPUB Validator (beta), dostupno na: <http://validator.idpf.org>, posijećeno: 15.08.2016.

[38] Grossman, J., ur. *The Chicago manual of style*. University of Chicago Press, 1993.

[39] Swanson, E., O'Sean, A. A., Schleyer, A. T. *Mathematics into type*. American Mathematical Soc., 1999.

[40] Jokela, T., Iivari, N., Matero, J., Karukka, M. The standard of user-centered design and the standard definition of usability: analyzing ISO 13407 against ISO 9241-11. *Proceedings of the Latin American conference on Human-computer interaction*. ACM, 2003.

[41] WebAIM (2014) Designing for Screen Reader Compatibility, dostupno na: <http://webaim.org/techniques/screenreader/>, posijećeno: 15.08.2016.

[42] The World Wide Web Consortium (2016) ARIA1: Using the aria-describedby property to provide a descriptive label for user interface controls, dostupno na: <https://www.w3.org/TR/WCAG20-TECHS/ARIA1.html>, posijećeno: 15.08.2016.

[43] GH LLC (2006) MathSpeak Initiative, dostupno na: <http://www.gh-mathspeak.com/>, posijećeno: 15.08.2016.

[44] The World Wide Web Consortium (2015) W3C MathML 3.0 Approved as ISO/IEC International Standard, dostupno na: <https://www.w3.org/2015/06/mathmlpas.html.en>, posijećeno: 15.08.2016.

[45] International Digital Publishing Forum (2016) EPUB Content Documents 3.0 Embedded MathML, dostupno na: <http://www.idpf.org/epub/30/spec/epub30-contentdocs.html#sec-xhtml-mathml>, posijećeno: 15.08.2016.

[46] W3C – The World Wide Web Consortium (2014) Mathematical Markup Language (MathML) Version 3.0 2nd Edition - Presentation Markup, dostupno na: <https://www.w3.org/TR/MathML3/chapter3.html>, posijećeno: 15.08.2016.

[47] MathJax Consortium (2016) MathJax, dostupno na: <https://www.mathjax.org>, posijećeno: 15.08.2016.

[48] Cervone, D. MathJax: a platform for mathematics on the Web. *Notices of the AMS* 59.2 (2012): 312-316.

[49] Topping, P. Using MathType to create TeX and MathML equations. *Proceedings of the 1999 TEX Annual Meeting, TUGBoat*. Vol. 20. No. 3. 1999.

[50] Bazargan, K. TEX as an eBook reader. *ArsTEXnica* (2009): 40.

[51] IMS Global Learning Consortium Inc. (2016) IMS Guidelines for Developing Accessible Learning Applications, dostupno na:  
<http://www.imsglobal.org/accessibility/accessiblevers/sec11.html>, posijećeno:  
15.08.2016.

[52] Design Science (2016) MathPlayer, dostupno na:  
<http://www.dessci.com/en/products/mathplayer/>, posijećeno: 15.08.2016.

[53] Apple Inc. (2016) VoiceOver for OS X, dostupno na:  
<https://www.apple.com/accessibility/osx/voiceover/>, posijećeno: 15.08.2016.

[54] Lenzi, V. B., Leporini, B. Investigating an accessible and usable ePub book via VoiceOver: a case study. *Human Factors in Computing and Informatics*. Springer Berlin Heidelberg, 2013. 272-283.

[55] Nazemi, A., Murray, I., Mohammadi, N. Mathspeak: An audio method for presenting mathematical formulae to blind students. *2012 5th International Conference on Human System Interactions*. IEEE, 2012.

[56] The DAISY Consortium (2016) DAISY consortium: DAISY Digital Talking Book, dostupno na: <http://www.daisy.org/daisypedia/daisy-digital-talking-book>, posijećeno: 15.08.2016.

[57] The DAISY Consortium (2012) MathML in DAISY 3 Structure Guidelines, dostupno na: <http://www.daisy.org/z3986/structure/SG-DAISY3/part2-math.html>, posijećeno: 15.08.2016.

[58] Kerer, C., Kirda, E. Layout, content and logic separation in web engineering. *Web Engineering*. Springer Berlin Heidelberg, 2001. 135-147.

[59] EPUBReader (2009) EPUBReader, dostupno na: <http://www.epubread.com/en/>, posijećeno: 15.08.2016.

[60] International Digital Publishing Forum (2016) EPUB 3 Accessibility Guidelines: Progressive Enhancement, dostupno na: <https://idpf.github.io/a11y-guidelines/content/script/pe.html>, posijećeno: 15.08.2016.

[61] Radium foundation (2016) Radium, dostupno na: <http://radium.org/>, posijećeno: 15.08.2016.

[62] Google (2016) IDEAL Group Reader, dostupno na: <https://play.google.com/store/apps/details?id=org.easyaccess.epubreader&hl=en>, posijećeno: 15.08.2016.

### 13. POPIS SLIKA I TABLICA

Slike:

Slika 1. Struktura PDF dokumenta

Slika 2. Slika strukture EPUB 3 e-knjige: datoteka *mimetype* sadrži tekst kojim e-čitač prepoznaje EPUB; direktorij META-INF sadrži datoteku *container.xml* koja ukazuje na dokument *package*; direktorij OPS (Open Publication Structure) sadrži sve dokumente sadržaja, dokument *package*, metapodatke, izvore publikacije i njihove lokacije, CSS dokument te tablicu sadržaja (*toc.xhtml*).

Slika 3. Prikaz strukture dokumenta *package*

Slika 4. Osnovni izgled elementa `<metadata>` i njegove obvezne djece u EPUB e-knjizi

Slika 5. Prikaz strukture navigacijskog dokumenta u EPUB 3 e-knjizi

Slika 6. Prikaz tipografskih pravila kompleksnih matematičkih izraza

Slika 7. Uvrštavanje kompleksnog opisa slike uz pomoć jezika WAI-ARIA i HTML elemenata `<figure>` i `<figcaption>`

Slika 8. Cauchy-Schwartzova jednadžba koja se primjenjuje za testiranje alata MathML

Slika 9. Prikaz u e-čitaču EPUBReader unutar Firefoxa (Gecko) koji ima nativnu MathML podršku

Slika 10. Prikaz postavki skriptiranja e-čitača EPUBReader u Firefoxu

Slika 11. Dohvaćanje prvog operatora i njegove udaljenosti od lijevog ruba

Slika 12. Dohvaćanje svih elemenata `<mrow>` zbog apliciranja uvlake

Slika 13. Usporedba udaljenosti pojedinih elemenata `<mrow>` po osi *y* te aplikacija uvlake

Slika 14. Resetiranje margina na vrijednost 0 px svih elemenata `<mrow>`

Slika 15. Prikaz EPUB e-knjige u e-čitaču EPUBReader (Firefox) nakon intervencije JavaScriptom. Sadržaj je vidljiv te optimalno tipografski oblikovan aplikacijom uvlake u veličini udaljenosti prvog operatora.

Slika 16. Prikaz matematičkog izraza nakon višestrukog povećanja veličine fonta ugrađenom opcijom e-čitača

Slika 17. Prikaz dinamičkog prijeloma tijekom promjene veličine prozora preglednika



Slika 18. Prikaz oblikovane EPUB e-knjige bez intervencija CSS-a i JavaScripta u e-čitaču iBooks na uređaju iPhone (iOS)

Slika 19. Prikaz rješenja aplicirane funkcije JavaScript za iBooks na platformi iOS

Slika 20. Funkcija koja obavlja upit podržanosti metode `offsetTop` izjavom `switch` te shodno primjenjuje potrebnu metodu za dohvaćanje udaljenosti

Slika 21. Prikaz e-knjige prije intervencije CSS-om i JavaScriptom u e-čitaču Adobe Digital Editions 4.01 na platformi Mac OS X

Slika 22. Prikaz e-knjige u e-čitaču Adobe Digital Editions 4.01 na platformi Mac OS X nakon intervencije CSS-om i funkcijom JavaScript

Slika 23. Prikaz matematičke formule bez intervencije CSS-a ili JavaScripta u e-čitaču Radium u pregledniku Chrome na platformi Mac OS X

Slika 24. Prikaz Dokument Object Modela MathML sadržaja u e-čitaču Radium

Slika 25. Dohvaćanje prvog operatora i regularni izraz s pomoću kojeg se dohvaća udaljenost prvog operatora od ruba prozora

Slika 26. Prikaz petlje koja dohvaća sve retke matematičkog izraza

Slika 27. Prikaz petlje koja utvrđuje pozicije pojedinih grupa elemenata, uspoređuje ih te aplicira novu željenu poziciju

Slika 28. Prikaz formule nakon apliciranja JavaScript funkcije u e-čitaču Radium u pregledniku Chrome

Slika 29. Prikaz matematičke formule na e-čitaču IDEAL Reader Group na tabličnom računalu Samsung Galaxy Tab 2 prije intervencije skriptnim jezikom

Slika 30. Prikaz konfiguracije MathJax knjižnice kod izlaznog prikaza „HTML-CSS”

Slika 31. Prikaz DOM strukture matematičke formule uz pomoć MathJax knjižnice s izlazom HTML-CSS

Slika 32. Prikaz funkcije JavaScript za apliciranje uvlake na platformi Android

Slika 33. Prikaz dohvaćanja udaljenosti prvog operatora za IDEAL Group Reader

Slika 34. Prikaz matematičke formule u e-čitaču IDEAL Group Reader na platformi Android nakon intervencije konfiguracijom MathJax knjižnice i funkcije JavaScript

Tablice:

Tablica 1. Usporedba formata e-knjiga prema definiranim parametrima za optimizaciju dinamičkoga matematičkoga sloga

Tablica 2. Usporedba popularnih softverskih alata za kreiranje EPUB e-knjiga

Tablica 3. Usporedba načina ugrađivanja matematičkoga sadržaja u EPUB e-knjige

Tablica 4. Eksperimentalni rezultati usporedbe e-čitača prema definiranim parametrima optimizacije matematičkoga sloga u e-knjigama

## 14. PRILOZI

## 14.1. SADRŽAJ DOKUMENTA *PACKAGE.OPF*

```
<?xml version="1.0" encoding="utf-8"?>
<package version="3.0" xml:lang="en" unique-
identifier="mojamat" xmlns="http://www.idpf.org/2007/opf">
  <metadata xmlns:dc="http://purl.org/dc/elements/1.1/">
    <dc:title id="title">Moja matematika</dc:title>
    <dc:identifier id="mojamat">2014-11-
10T09:38:00Z</dc:identifier>
    <dc:language>hr</dc:language>
    <dc:creator>Maja Turcic</dc:creator>
    <meta property="dcterms:modified">2014-11-
10T09:38:00Z</meta>
  </metadata>
  <manifest>
    <item id="css" href="style.css" media-type="text/css"/>
    <item id="p1" media-type="application/xhtml+xml"
href="poglavlje1.xhtml" properties="mathml"/>
    <item id="p2" media-type="application/xhtml+xml"
href="poglavlje2.xhtml" properties="mathml scripted"/>
    <item id="p3" media-type="application/xhtml+xml"
href="poglavlje3.xhtml" properties="mathml scripted"/>
    <item id="p4" media-type="application/xhtml+xml"
href="poglavlje4.xhtml" properties="mathml scripted"/>
    <item id="p5" media-type="application/xhtml+xml"
href="poglavlje5.xhtml" properties="mathml scripted"/>
    <item id="p6" media-type="application/xhtml+xml"
href="poglavlje6.xhtml" properties="mathml scripted"/>
    <item id="p7" media-type="application/xhtml+xml"
href="poglavlje7.xhtml" properties="mathml scripted"/>
    <item id="nav" media-type="application/xhtml+xml"
href="toc.xhtml" properties="nav"/>
  </manifest>
  <spine>
```

```
<itemref idref="p1"/>
<itemref idref="p2"/>
<itemref idref="p3"/>
<itemref idref="p4"/>
<itemref idref="p5"/>
<itemref idref="p6"/>
<itemref idref="p7"/>
</spine>
</package>
```

## 14.2. KOD MATEMATIČKE FORMULE U JEZIKU MATHML

```
<?xml version="1.0" encoding="utf-8"?>
<html xml:lang="en-us" lang="en-us"
xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <meta charset="utf-8"/>
    <title>Primjer matematičke formule u EPUB3</title>
  </head>
  <body>
    <math xmlns="http://www.w3.org/1998/Math/MathML"
id='matem'>
      <mrow>
        <mstyle displaystyle='true'>
          <munder>
            <mo>&#x222C;</mo>
            <mrow><mi>D</mi></mrow>
          </munder>
        </mstyle>
        <mrow>
          <mo>(</mo>
          <mrow>
            <msub><mi>Q</mi><mi>x</mi></msub>
            <mo>&#x2212;</mo>
            <msub><mi>P</mi><mi>y</mi></msub>
          </mrow>
          <mo>)</mo>
        </mrow>
        <mi>d</mi><mi>A</mi>
      </mrow>
      <mo id="firstOperator" linebreak='nobreak'>=</mo>
      <mrow>
```

```

<mstyle displaystyle='true'>
  <munder>
    <mo>⋮</mo>
    <mrow>
      <msub><mi>D</mi><mn>1</mn></msub>
        </mrow>
    </munder>
  </mstyle>
  <mrow>
    <mo>(</mo>
    <mrow>
      <msub><mi>Q</mi><mi>x</mi></msub>
      <mo>⋮</mo>
      <msub><mi>P</mi><mi>y</mi></msub>
    </mrow>
    <mo>)</mo>
  </mrow>
  <mi>d</mi><mi>A</mi>
</mrow>

<mrow>
  <mo>+</mo>
  <mstyle displaystyle='true'>
    <munder>
      <mo>⋮</mo>
      <mrow>
        <msub><mi>D</mi><mn>2</mn></msub>
          </mrow>
      </munder>
    </mstyle>
  <mrow>
    <mo>(</mo>

```

```

        <mrow>
            <msub><mi>Q</mi><mi>x</mi></msub>
            <mo>#x2212;</mo>
            <msub><mi>P</mi><mi>y</mi></msub>
        </mrow>
        <mo>)</mo>
    </mrow>
    <mi>d</mi><mi>A</mi>
</mrow>

```

```

<mrow>
    <mo>=</mo>
    <mstyle displaystyle='true'>
        <munder>
            <mo>#x222E;</mo>
            <mrow>

```

```

<msub><mi>C</mi><mn>1</mn></msub>
            <mo>#x222A;</mo>

```

```

<msub><mi>C</mi><mn>3</mn></msub>
            </mrow>

```

```

        </munder>

```

```

    </mstyle>

```

```

    <mi>P</mi><mi>d</mi><mi>x</mi><mo>+</mo><mi>Q</mi><mi>d</mi>
</mi><mi>y</mi>
    </mrow>

```

```

<mrow>
    <mo>+</mo>
    <mstyle displaystyle='true'>
        <munder>
            <mo>#x222E;</mo>

```



<mrow>

<msub><mi>C</mi><mn>2</mn></msub>

<mo>#x222A;</mo>

<mo> (</mo>

<mo>#x2212;</mo>

<msub><mi>C</mi><mn>3</mn></msub>

<mo>)</mo>

</mrow>

</munder>

</mstyle>

<mi>P</mi><mi>d</mi><mi>x</mi><mo>+</mo><mi>Q</mi><mi>d</mi><mi>y</mi>

</mrow>

<mrow>

<mo>=</mo>

<mstyle displaystyle='true'>

<munder>

<mo>#x222E;</mo>

<mrow>

<msub><mi>C</mi><mn>1</mn></msub>

</mrow>

</munder>

</mstyle>

<mi>P</mi><mi>d</mi><mi>x</mi><mo>+</mo><mi>Q</mi><mi>d</mi><mi>y</mi>

</mrow>

<mrow>

```

        <mo>+</mo>
        <mstyle displaystyle='true'>
            <munder>
                <mo>&#x222E;</mo>
                <mrow>

<msub><mi>C</mi><mn>3</mn></msub>
                </mrow>
            </munder>
        </mstyle>

        <mi>P</mi><mi>d</mi><mi>x</mi><mo>+</mo><mi>Q</mi><mi>d
</mi><mi>y</mi>
        </mrow>

    <mrow>
        <mo>+</mo>
        <mstyle displaystyle='true'>
            <munder>
                <mo>&#x222E;</mo>
                <mrow>

<msub><mi>C</mi><mn>2</mn></msub>
                </mrow>
            </munder>
        </mstyle>

        <mi>P</mi><mi>d</mi><mi>x</mi><mo>+</mo><mi>Q</mi><mi>d
</mi><mi>y</mi>
        </mrow>

    <mrow>
        <mo>+</mo>
        <mstyle displaystyle='true'>

```

```

<math display="block">
\frac{C^3}{C^3}


$$Pdx + Qdy$$


```

## 15. ŽIVOTOPIS I POPIS OBJAVLJENIH RADOVA

Maja Turčić rođena je 1982. godine u Zagrebu, gdje je 2000. godine završila srednju školu International Baccalaureate. Diplomirala je na Grafičkom fakultetu Sveučilišta u Zagrebu 2006. godine s temom „SVG animacija teksta” pod mentorstvom prof. Žiljka. Radno iskustvo započinje na Tehničkom veleučilištu u Zagrebu 2006. godine na poziciji asistenta na predmetima Računalna tipografija, Grafički programski jezici, Interaktivno programiranje na *webu* te Pretražnici i navigacija na *webu*. Praktično iskustvo grafičkog dizajna prikuplja od 2006. godine volontirajući godinu dana u studiju za dizajn Parabureau. Predavačica postaje 2012. godine te drži predavanja na predmetima Oblikovanje *web*-stranica i Grafički programski jezici te osniva novi predmet Oblikovanje e-literature.

Osim što je znanstvenim radovima sudjelovala na šest međunarodnih znanstvenih konferencija, objavila je i dva znanstvena rada u znanstvenim časopisima i poglavlje u knjizi. Također je zanima unaprjeđenje metoda predavanja, stoga je sudjelovala u tri programa Erasmus.

Popis radova:

1. Turčić, Maja; Janković, Mario. Kako Digital Rights Movement šteti e-knjigama // TISKARSTVO & DIZAJN 2013 / Žiljak Vujić, Jana (ur.). Zagreb: FS, FotoSoft, 2013. 143-147 (predavanje, međunarodna recenzija, objavljeni rad, znanstveni).
2. Turčić, Maja; Predanić, Stipe. Pripremanje Web sadržaja za tisak (na primjeru HTML form elementa putem Processinga) // TISKARSTVO 2011 / Žiljak, Vilko (ur.). Zagreb: FS, FotoSoft, 2011. 89-94 (predavanje, međunarodna recenzija, objavljeni rad, znanstveni).
3. Turčić, Maja; Koren, Tajana; Rudolf, Maja. Geometrical measurements of diacritical dcroat sign // Annals of DAAAM for 2010 & Proceedings of the 21st International DAAAM Symposium „Intelligent Manufacturing & Automation: Focus on Interdisciplinary Solutions” 20-23rd October 2010, Zadar, Croatia / Katalinić, Branko (ur.). Vienna, Austria: DAAAM Internatinal Vienna, 2010. 1327-1328 (predavanje, međunarodna recenzija, objavljeni rad, znanstveni).

4. Stanić Loknar, Nikolina; Rudolf, Maja; Turčić, Maja. Pixel Manipulation in Security Graphics // 11th International design conference: Design 2010 / Žiljak, Vilko; Milčić, Diana (ur.). Zagreb: Grafički fakultet, Sveučilište u Zagrebu, 2010. 1947-1952 (predavanje, međunarodna recenzija, objavljeni rad, znanstveni).
5. Turčić, Maja; Rudolf, Maja; Uglješić, Vesna; Stanić Loknar, Nikolina. Uvođenje novih tipografskih elemenata na Web // Tiskarstvo 2010 / Žiljak, Vilko (ur.). Zagreb: Fotosoft, 2010. 5-10 (predavanje, međunarodna recenzija, objavljeni rad, znanstveni).
6. Rudolf, Maja; Stanić Loknar, Nikolina; Turčić, Maja; Koren, Tajana. Raster elements in securities typography // Proceedings of the 10th International Design Conference (Design 2008) Workshop: Design of Graphic Media / Žiljak, Vilko (ur.). Zagreb; Glasgow: Faculty of Mechanical Engineering and Naval Architecture; Design Society, 2008. 1475-1479 (predavanje, međunarodna recenzija, objavljeni rad, znanstveni).
7. Pap, Klaudio; Turčić, Maja; Rudolf, Maja. Dinamička izrada PDF dokumenata // Tiskarstvo 2010, Digitalni sustavi u tiskarstvu / Žiljak, Vilko (ur.). Zagreb: FotoSoft d.o.o., 2010. 61 (predavanje, međunarodna recenzija, sažetak, znanstveni).
8. Turčić, Maja; Rudolf, Maja; Pavlović, Tomislav. Individualizirani sadržaj PDF knjige kroz Processing // Znanstveno-stručni skup Tiskarstvo 09: knjiga sažetaka. Zagreb, 2009. (predavanje, međunarodna recenzija, sažetak, znanstveni).
9. Turčić, Maja; Rudolf, Maja. Individualni slovni znakovi vektorske grafike u XML-u (Primjene na karakteristične znakove hrvatske abecede u SVG-u) // Tiskarstvo 08 / Žiljak, Vilko (ur.). Zagreb: FotoSoft d.o.o., 2008. 20-20 (predavanje, međunarodna recenzija, sažetak, znanstveni).
10. Rudolf, Maja; Turčić, Maja; Pavlović, Tomislav. Interaktivne aplikacije na Web-u (SVG) // Zbornik radova sa znanstveno-stručnog skupa Tiskarstvo 08 / Žiljak, Vilko (ur.). Zagreb: FotoSoft, 2008. 21-21 (predavanje, međunarodna recenzija, sažetak, znanstveni).
11. Rudolf, Maja; Turčić, Maja. Tipografija u grafičkim programskim jezicima (sa primjenom u Processingu) // Tiskarstvo 08 / Žiljak, Vilko (ur.). Zagreb: FotoSoft d.o.o., 2008. 22-22 (predavanje, međunarodna recenzija, sažetak, znanstveni).

12. Turčić, Maja; Koren, Tajana; Rudolf, Maja. Digital character design of dcroat mark // Daaam International Scientific Book 2011 / Katalinić, Branko (ur.). Vienna: DAAAM International Vienna, 2011.
13. Turčić, Maja; Žiljak, Vilko; Žiljak-Stanimirović, Ivana. Individual Stochastic Screening for the Development of Computer Graphics. // Acta graphica. 22 (2011), 3-4; 69-78 (članak, znanstveni).
14. Turčić, Maja; Uglješić, Vesna; Rajković, Ivan; Koren, Antun. Design and positioning of diacritical marks in latin typefaces. // Acta graphica. 22 (2010), 3-4; 5-15 (članak, znanstveni).