Web aplikacija za izradu vektorske grafike

Mihalić, Ivan

Undergraduate thesis / Završni rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:045543

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-20**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA

Stručni studij

WEB APLIKACIJA ZA IZRADU VEKTORSKE GRAFIKE

Završni rad

Ivan Mihalić

Osijek, 2020 godina.

Sadržaj

1. UVOD 1
1.1 Zadatak završnog rada 2
2. USPOREDBA WEB APLIKACIJA ZA IZRADU VEKTORSKE GRAFIKE 3
3. RAČUNALNA GRAFIKA 6
3.1 Povijest 6
3.2. Razlika između rasterske i vektorske grafike 7
4. WEB APLIKACIJA ZA IZRADU VEKTORSKE GRAFIKE 9
4.1. Korištene tehnologije 9
4.1.1. SVG 9
4.1.2. HTML 10
4.1.3. CSS 10
4.1.4. JavaScript 10
4.1.5. XML format 11
4.1.6. Visual Studio Code 11
4.2. Dizajn aplikacije 12
4.3. Opis aplikacije 12
4.4. Mogućnosti aplikacije 19
5. ZAKLJUČAK 28
LITERATURA 29
POPIS UPOTREBLJENIH OZNAKA 31
SAŽETAK 32
ABSTRACT 33
ŽIVOTOPIS 34

1. UVOD

U današnje vrijeme je sve veća potreba za korištenje vektorskih grafičkih programa jer vektorska grafika ima mnogo prednosti u odnosu na druge grafičke programe koji koriste rastersku grafiku. Vektorska grafika se sastoji od točaka ili koordinata koje su povezane linijama i krivuljama koje se nazivaju putanje. Neke od najvažnijih prednosti koje vektorska grafika omogućuje korisnicima su to što korisnik može neograničeno skalirati vektorsku sliku bez gubitka kvalitete slike za razliku od rasterske grafike koja gubi na kvaliteti nakon skaliraja. Vektorska grafika je skalabilna što omogućuje korisnicima da lako mogu smanjivati ili povećavati grafičke elemente u bilo kojoj veličini. Vektorske datoteke imaju manju veličinu datoteke jer u sebi sadrže hrpu koordinata umjesto složenih informacija poput tisuća ili čak milijuna piksela na digitalnoj fotografiji.

Vektorska grafika se može uređivati pomoću kôda što omogućuje korisnicima da sami mogu uređivati vektorsku sliku pomoću vlastitog kôda. Još jedna od prednosti vektorske grafike što vektorska grafika može biti interaktivna. Grafički elementi mogu biti interaktivni na web stranici ili se mogu mjenjati ovisno o interakciji korisnika. Vektorska grafika se koristi za izradu logo-a, geometrijskih oblika, logotipa, crteža. plakata, 3D prikazivanja i animacija. Programi za vektorsku grafiku često se koriste za stvaranje ilustracija visoke razlučivosti za upotrebu na webu, u igrama i drugoj multimediji. Korisniku se omogućuje da stvara i mijenja vektorske slike te ih pohranjuje u vektorske grafičke formate kao što su PDF, SVG, WMF (engl. *Windows Metafile*).

U drugom poglavlju ovoga rada usporedit ćemo dvije slične web aplikacije i opisati njihove mogućnosti. U trećem poglavlju ovoga rada se ukratko opisuje općenita povijest računalne grafike i razlika između rasterske i vektorske grafike. U četvrtom poglavlju je objašnjeno koja je zadaća same aplikacije, opisuju se tehnologije korištene za izradu ove aplikacije, prikazan je dizajn aplikacije, njen opis te mogućnosti rada same aplikacije.

Objašnjeno je način na koji je web aplikacija napravljena i kako ju korisnik treba koristiti. Uz to su i dodane slike kako bih se lakše objasnio napisani određeni kôd te je prikazano kako konačna web aplikacija izgleda. Web aplikacija za izradu vektorske grafike je napravljena pomoću programskog jezika Scalable Vector Graphics (SVG) koji se može koristiti za crtanje vektorske grafike, a temelji se na jeziku XML. Korišten je programski jezik HTML, CSS i JavaScript.

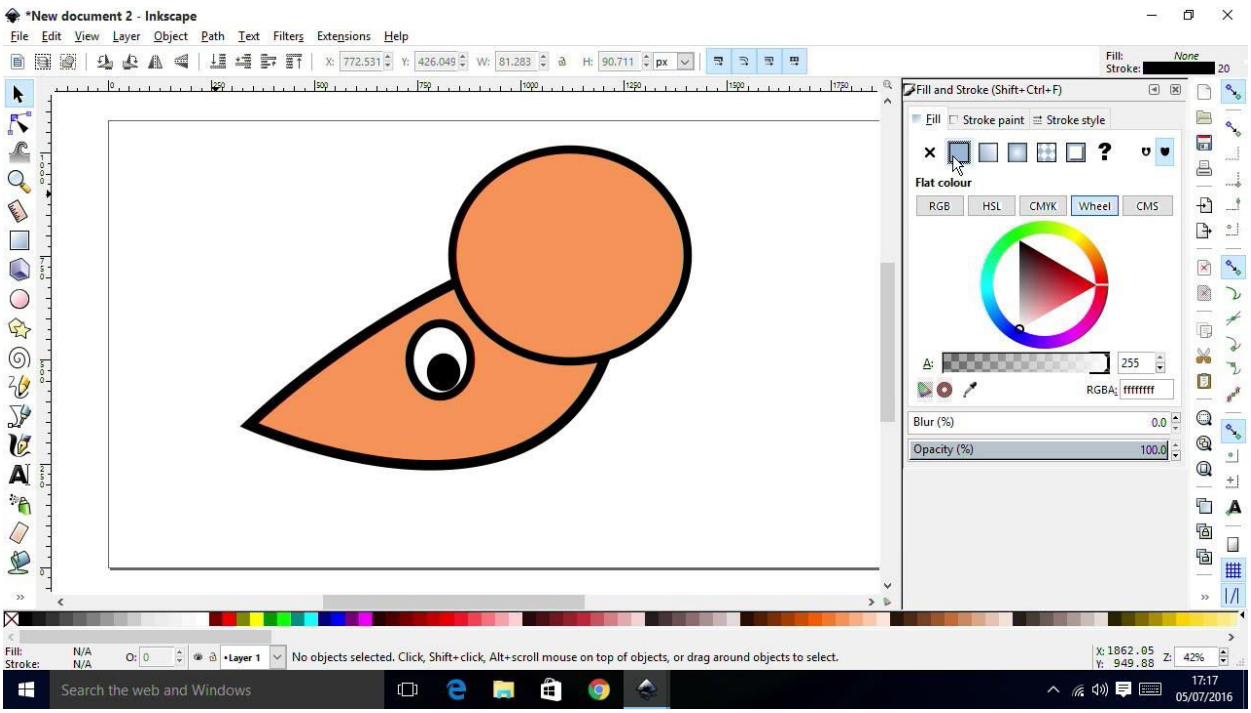
Web aplikacija se pokreće u web pregledniku Google Chrome, ali može se pokrenuti i u drugim web preglednicima.

1.1 Zadatak završnog rada

Cilj ovog završnog rada je opisati i izraditi web aplikaciju koja će korisniku omogućiti izradu osnovnih vektorskih grafičkih elemenata te ih korisnik potom može uređivati po želji. Za programiranje se koristio Visual Studio Code.

2. USPOREDBA WEB APLIKACIJA ZA IZRADU VEKTORSKE GRAFIKE

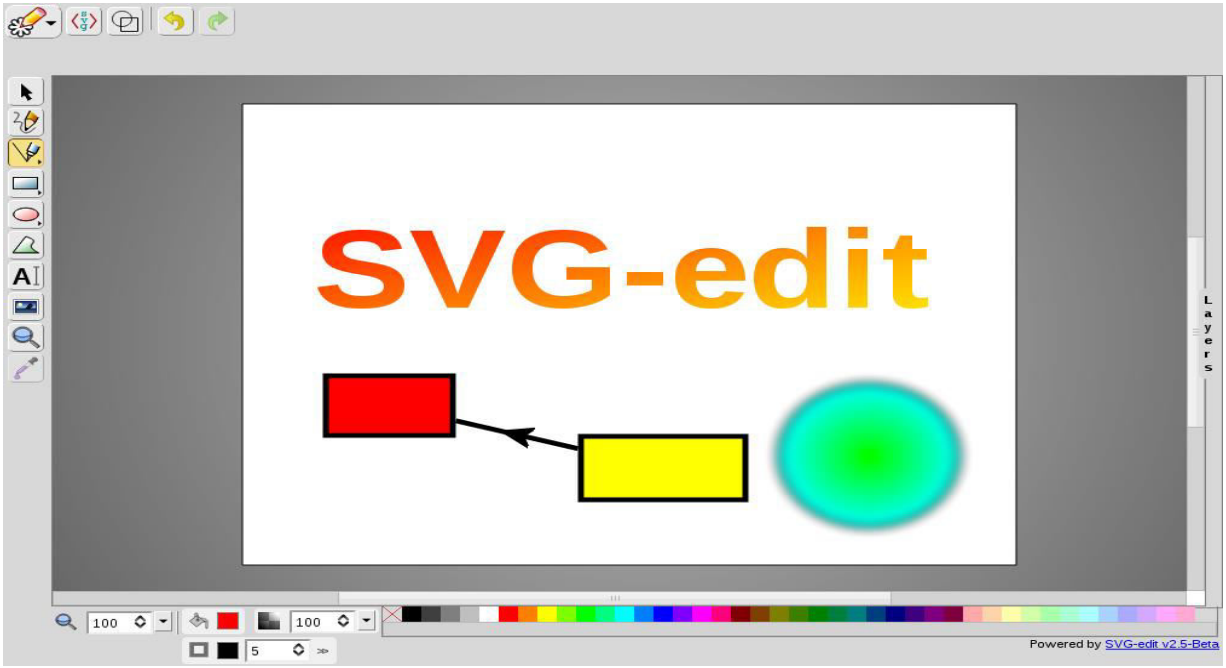
Postoji jako puno različitih web aplikacija za izradu vektorske grafike. U ovom poglavlju usporedit ćemo dva slična grafička programa za izradu vektorske grafike. Programi za crtanje vektorske grafike koriste se za stvaranje i uređivanje vektorske grafike. Grafičkim dizajnerima pruža se mogućnost stvaranja preciznih i skalabilnih slika. Jedan od najpoznatijih grafičkih programa je Inkscape. Inkscape je profesionalni vektorske grafički program koji se može pokrenuti na operacijskim sustavima kao što su Windows, Linux, Mac OS, Ubuntu te je u cijelosti preveden na hrvatski. Inkscape koristi W3C-ov otvoreni SVG standard kao izvorni format i besplatan je program otvorenog kôda [1].



Slika 2.1. *Prikaz Inkscape programa*

Inkscape koriste profesionalni dizajneri i amateri širom svijeta za stvaranje raznovrsne grafike, kao što su ilustracije, ikone, logotipi, dijagrami, karte i web grafike. Na primjer, tisuće slika na Wikipediji kreiraju se pomoću Inkscape-a. Inkscape pruža sofisticirane alate za crtanje vektorskih grafika, koji su u potpunosti usporedivi sa standardnim grafičkim programima. Prema slici 2.1. vidi se prikaz Inkscape programa koji sadrži mnogobrojne funkcije, jednostavno sučelje, višejezičnu podršku i mogućnost proširenja programa. Korisnici mogu prilagoditi Inkscape-ovu funkcionalost raznim dodacima. Inkscape može uvesti i izvesti razne formate,

uključujući SVG, PDF, EPS (engl. *Encapsuled PostScript*), CDR (engl. *CorelDraw*) VSD (engl. *Visio*) i AI (engl. *Adobe Illustrator*). Inkscape može izvorno otvoriti većinu rasterskih formata JPEG, PNG i GIF kao bitmape, ali može izvesti samo PNG bitmape. Inkscape ne namjerava izvesti druge rasterske formate osim PNG-a, jer je ovo trenutno najprikladniji format za vektorsku grafiku. Zato što je otvoreni format koji koristi kompresiju bez gubitaka i obično ga podržavaju svi grafički programi [2].



Slika 2.2. *Prikaz SVG-edit programa*

Svg-edit je pouzdani besplatni SVG grafički program otvorenog kôda za izradu vektorske grafike koji se pokreće pomoću JavaScripta. Aplikacija koristi samo HTML5, JavaScript, SVG, GIF, TIFF i CSS te je nedostatak što nema funkcionalnosti na strani poslužitelja, ali zato nije potrebna bilo kakva dodatna obrada na strani poslužitelja [3]. SVG-edit je program otvorenog kôda što znači da korisnik sam može preuzeti program i sam izraditi vlasititu verziju mijenjanjem izvornog kôda. SVG-edit može raditi u svim web preglednicima. SVG-edit je jednostavan grafički program za izradu mnogo značajki. Prema slici 2.2. vidi se prikaz SVG-edit programa koji koristi značajke poput izrade linije, elipse, poligona, pravokutnika, pretvaranje oblika u putanju, poravnanje, zumiranje, tekst koji se može stilizirati. Svg-edit omogućuje jednostavno pretvaranje SVG datoteke u formate kao što su WMP (engl. *Windows Media Format*), JPEG, PNG, BMP (engl. *Bitmap*), GIF i TIFF [4].

Sada kada se bolje razumije kako prethoda dva grafička programa rade može se primjetiti da se radi o vrlo dva slična grafička programa koji su pouzdani, besplatni i otvorenog kôda za izradu vektorske grafike. Inkscape grafički program se više koristi za profesionalne poslove, ali se može koristiti i za manje poslove, obične korisnike. Razlog tome je što Inkscape nudi puno više značajki za uređivanje vektorske grafike te je podržan na četiri strana jezika (engleski, njemački, poljski, švedski) dok je SVG-edit podržan za samo engleski jezik. SVG-edit se koristi više za manje poslove i obične korisnike. U usporedbi s ovom web aplikacijom može se primjetiti da je više slična SVG-edit grafičkom programu jer je jednostavna web aplikacija koja korisnicima omogućuje izradu osnovnih grafičkih oblika te nudi mogućnost njihovog uređivanja i spremanje izrađene slike u SVG formatu. Sva tri grafička programa rade na principu SVG te pružaju korisniku mogućnost za izradu vektorske grafike, ali i da korisnici mogu spremiti izrađenu sliku u različitim formatima.

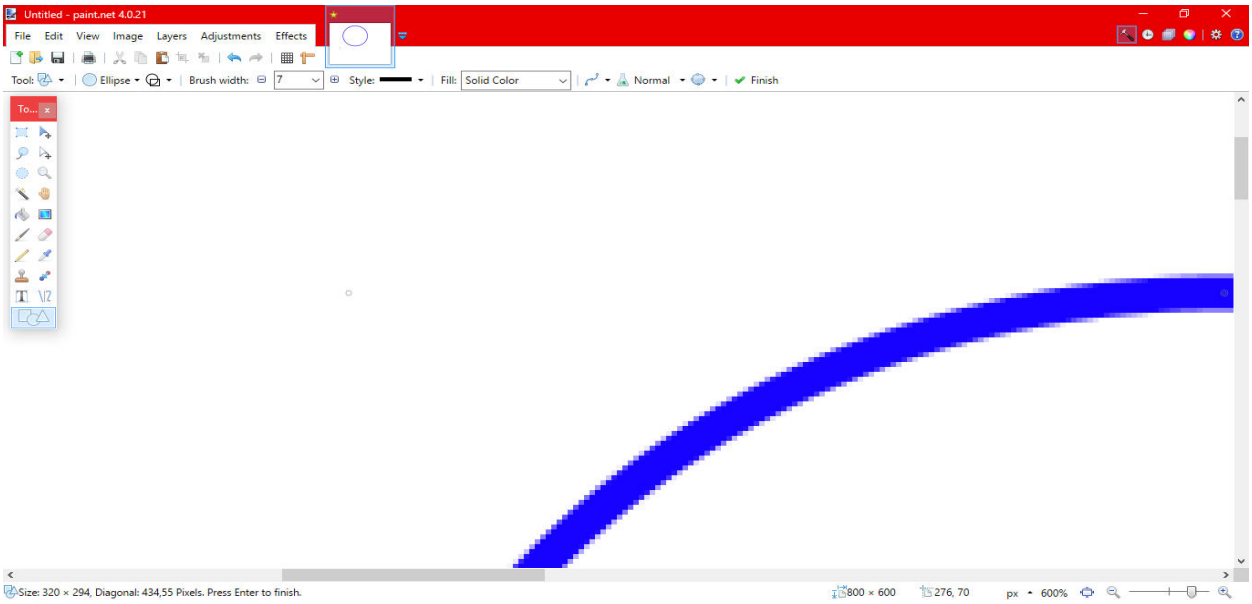
3. RAČUNALNA GRAFIKA

3.1 Povijest

Računalna grafika ima široku namjenu i služi za različite stvari kao što je izrađivanje arhitektonskih crteža, grafičkog dizajna, crtanje slika na računalu. Nacrtana slika predstavlja digitalni podatak. Slika se može uređivati na različite načine kao što je promjena veličine slike, rotiranje, promjena boja i još mnogo toga. Slika se može spremiti, poslati drugom korisniku, isprintati. Sve te mogućnosti su omogućene jer manipuliramo s digitalnim podatkom. Računalna grafika ima svoju namjenu i u medicini. Kada doktor pokaže neki dio tijela pomoću slike to je zapravo računalni grafički prikaz nacrtan pomoću velikih količina podataka.

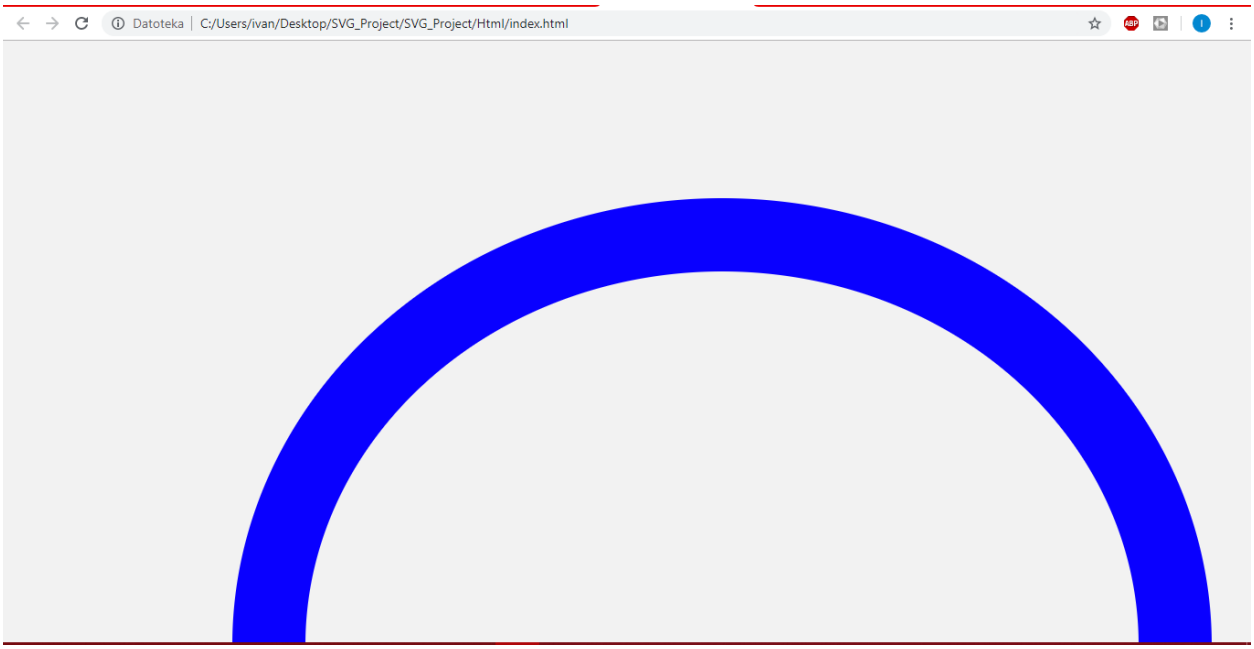
Prvi CAD (engl. *Computer aided design*) pojavio se 1959. godine koji je razvijen od strane IBM-a (engl. *International Business Machines*) i General Motors-a za stvaranje automobila na računalnom ekranu. Ovo se smatra jednim od najvećih postignuća u dizajniranju automobila. Student MIT-a (engl. *Massachusetts Institute of Tehnology*) Steve Russell 1961. godine je napravio prvu računalno grafičku igru na mini računalu koja se zove Spacewar. Ivan Sutherland 1963. godine razvija Sketchpad u kojem se slike mogu crtati na računalu. 1970. godine razvijaju se Bézierove krivulje koje postaju jedan od najvažnijih alata u vektorskoj grafici. 1985. godine Microsoft objavljuje prvu verziju osnovnog programa za crtanje rasterske grafike koji postaje jedan od najpopularnijih programa. Adobe Photoshop 1990. godine obavljuje svoju prvu verziju te postaje jedan od najpopularnijih svjetskih paketa za izradu grafičkog dizajna. Student sveučilišta Illinois Marc Andreessen razvija Mosaic, prvi web preglednik koji prikazuje tekst i slike jedni pored drugih, što je izazvalo veliki interes za internet. Konzorcij za W3C 1999. godine započinje razvoj SVG, temeljen na XML-u koji može pružati slike visoke kvalitete. Apple 2007. godine proizvodi *iPhone* i *iPod Touch* proizvode s grafičkim sučeljima s zaslonima osjetljivim na dodir [5].

3.2. Razlika između rasterske i vektorske grafike



Slika 3.1. *elipsa u paintu*

Prema slici 3.1. vidi se prikaz približene nacrtane elipse u paintu odnosno nacrtanu u programu za izradu rasterske grafike.



Slika 3.2. *elipsa izrađena u web aplikaciji za izradu vektorske grafike*

Prema slici 3.2. vidi se prikaz približene elipse izrađene u web aplikaciji za izradu vektorske grafike. Za razliku od rasterske grafike koja sliku pohranjuje kao pravokutnu mrežu točaka

piksela (engl. *pixel*), vektorska grafika sliku pohranjuje pomoću poligona. Poligoni su dvodimenzionalna geometrijska tijela omeđena konačnim brojem dužina. Glavna prednost takvih slika je da ih se može beskonačno uvećati bez gubitka kvalitete. Kao što se može jasno primjetiti razlika između slike 3.1. i slike 3.2. tako je dokazana tvrdnja da uvećanjem slike u vektorskoj grafici slika ne gubi na kvaliteti.

4. WEB APLIKACIJA ZA IZRADU VEKTORSKE GRAFIKE

Grafički dizajn je umjetnost sa svrhom. To uključuje kreativni i sustavni plan za rješavanje problema ili postizanje određenih ciljeva, pomoću slike, simbola ili čak riječi. To je vizualna komunikacija i estetski izraz koncepata i ideja koristeći različite grafičke elemente i alate. Zadaća web aplikacije je omogućiti korisniku da izabere osnovni vektorski grafički element po njegovoj želji te korisnik ima mogućnosti izrađivati vektorsku grafiku na površini koja je namjenjena za crtanje. Korisnik ima mogućnosti izrađivati više grafičkih elemenata na istoj površini za crtanje.

Svaki grafički element korisnik može uređivati po želji što uključuje promjenu boju linije, promjenu debljine linije, ispuna bojom unutar grafičkog elementa te brisanje nactanog grafičkog elementa. Korisnik nakon što izradi vektorsku grafiku ima mogućnost preuzeti sliku na svoje računalo u SVG formatu. Web aplikacija prikazuje grafičke elemente na engleskom jeziku te ukoliko korisnik želi uređivati grafički element također je sve napisano na engleskom jeziku što omogućuje korištenje aplikacije za veći dio populacije u svijetu.

4.1. Korištene tehnologije

4.1.1. SVG

Scalable Vectors Graphics (SVG) je format vektorske slike na bazi XML-a za dvodimenzionalnu grafiku s podrškom za interaktivnost i animaciju. SVG specifikacija je otvoreni standard koju je razvio World Wide Web Consortium (W3C) od 1999. godine [6]. SVG slike i njihova ponašanja definirani su u XML tekstualnim datotekama. Što znači da se mogu pretraživati, kreirati i uređivati s bilo kojim uređivačem teksta, kao i pomoću softvera za crtanje. SVG se može prikazati u različitim web preglednicima kao što su Google Chrome, Mozilla Firefox, Internet Explorer, Microsoft Edge, Safari i Opera. SVG datoteke obično imaju ekstenziju .svg, a pošto se temelje na XML standardu može ih se čitati i uređivati u tekstualnom obliku. Dodatna prednost korištenja XML formata je da se njime može upravljati pomoću JavaScripta, što omogućuje animiranje SVG slike i interakciju s korisnikom, na isti način kao i HTML dokument. Slike u SVG formatu su obično manje od slika u rasterskim formatima (.jpg, .png, .bmp) i mogu se efikasno komprimirati bez gubitaka isto kao i tekst (komprimirana .svg slika koristi ekstenziju .svgz).

4.1.2. HTML

HTML je najosnovniji opisni jezik koji služi za izradu web stranica. Timothy Berners-Lee ga je definirao 1990. godine. U HTML upotrebljavamo oznake (engl. *tags*) kao što su <head>, <title>, <body>, <header>, <footer>, <p>, i mnoge druge koje definiraju izgled i dojam podataka u HTML-u. Svaki *tag* ima početnu i završnu oznaku. Na primjer, <head> označava početak, a </head> označava kraj elementa. Ime elementa unutar oznake ne razlikuje velika i mala slova [7].

4.1.3. CSS

CSS je među osnovnim opisnim jezikom koji je razvijen od strane W3C specifikacije. Standardiziran je u web preglednicima i koji opisuje kako se elementi trebaju prikazivati na zaslonu, papiru, govoru i u drugim medijima. Služi za uređivanje i izgled same web stranice. Pomoću CSS-a može se uređivati promjena boje, veličine, razmaka sadržaja ili dodavanje animacija i drugih ukrasnih značajki. Najčešće se u CSS-u uređuje tekst kao što je postavljanje fonta, podebljanost, kurziv, razmak između redaka i slova, sjene i druge značajke teksta [8].

4.1.4. JavaScript

JavaScript je najpoznatiji kao interpretativni skriptni jezik za web stranice iako ga koriste i mnoga druga okruženja koja nisu preglednici poput Node.js, Apache CouchDB i Adobe Acrobat. JavaScript je dinamički programski jezik koji je prvotno napravljen da omogući interaktivnost web stranicama. Služi programerima za izradu skalabilnih, interaktivnih web aplikacija [9]. JavaScript može dodati novi HTML na stranicu, promijeniti postojeći sadržaj, izmjeniti stilove, reagirati na radnje korisnika mišom, postavljanje pitanja korisniku. U HTML dokumentu implementira se između dva *taga* koji se zovu <script>...</script>. Mnogi ga povezuju s programskim jezikom Java, ali JavaScript je poptuno samostalan programski jezik sa svojim vlastitim specifikacijama (ECMAScript) i nema povezanosti s Javom. JavaScript se može

pokrenuti ne samo na web pregledniku već na bilo kojem uređaju koji ima instaliran program (JavaScript engine) [10].

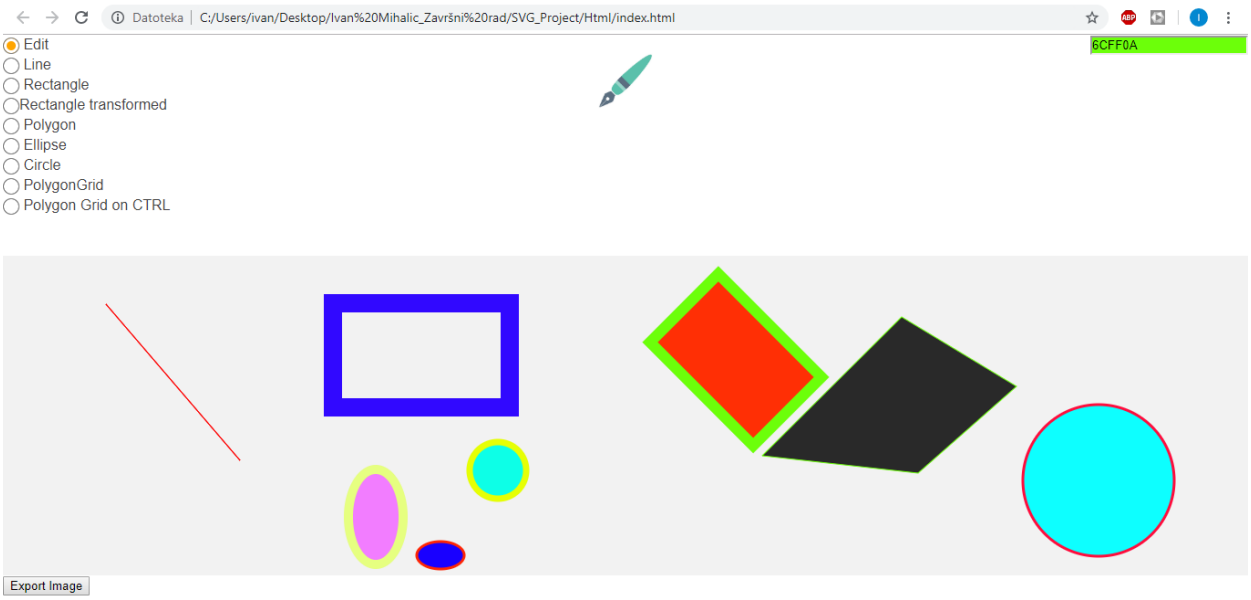
4.1.5. XML format

XML je jedan od najpopularnijih tehnologija za razmjenu podatka između baza podataka i računala korisnika. XML podaci se mogu umetnuti unutar HTML oznaka i prikazati ih na web stranici. XML koristi početni i završni *tag* uz pomoć kojeg se može odrediti točna vrsta podatka koji se pregledava. XML omogućuje stvaranje bilo koje oznake za opisivanje i strukturiranje podataka. XML ne ovisi o platformi što znači da svaki program stvoren za korištenje XML-a može čitati i obrađivati podatke, neovisno o hardveru ili operacijskom sustavu [11].

4.1.6. Visual Studio Code

Visual Studio Code je alat za razvojno okruženje koji je Microsoft razvio za Windows, Linux i Mac OS. Uključuje podršku za ispravljanje pogrešaka, isticanje sintakse, inteligentno dovršavanje koda, isječke, korigiranje koda i ugrađeni Git te GitHub. Visual Studio Code je vrlo prilagodljiv, dopušta korisnicima da promijene temu. postavke i instaliraju proširenja koja dodaju dodatnu funkcionalnost. Visual Studio Code je moderan alat za programiranje koji pruža programerima da ga koriste s različitim programskim jezicima i pruža dodatne značajke. Visual Studio Code je besplatan otvoreni izvor i objavljen je pod dozvolom MIT licence [12].

4.2. Dizajn aplikacije



Slika 4.1. *Web aplikacija za izradu vektorske grafike*

Prema slici 4.1. jasno se prikazuje web aplikacija za izradu vektorske grafike i nacrtane različite grafičke elemente. Korisniku su s lijeve strane ponuđeni osnovni grafički elementi za crtanje te *edit* kao mogućnost uređivanja grafičkog elemena i ispod toga je radna površina koja je namjenjena za crtanje grafičkih elemenata. Gore desno se nalazi izbornik koji nudi širok spektar različitih boja za odabrati. Korisnik ima mogućnost spremiti nacrtanu sliku i preuzeti je na svoje računalo u SVG formatu.

4.3. Opis aplikacije

Web aplikacija je zamišljena kao aplikacija u kojoj će korisnik moći odabrati i izrađivati osnovne grafičke elemente te ih potom uređivati po njegovoj želji. U ovom poglavlju će se objasniti pomoću slika napisani kôd i pomoću teksta kako je web aplikacija realizirana. Web aplikacija je napisana u HTML-u jer se SVG može ugraditi u HTML stranice.

```
<!DOCTYPE html>
<html>

    <head>
        <meta charset='utf-8'>
        <meta http-equiv="X-UA-Compatible" content="chrome=1">
        <meta name="description" content="Svg.draw.js : Demo-Page">
```

Slika. 4.2. *Prikazuje meta elemente i spajajanje CSS-a*

Prvo su u radu upotrebljeni metapodaci s oznakom <meta> koji daju metapodatke o HTML dokumentu. Meta elementi je mali dio informacijskog kôda koji se nalazi unutar <head>... </head> elementa. Metapodaci neće biti prikazani na stranici, ali će se moći obraditi strojno. Meta elementi se obično koriste za određivanje opisa stranice, ključnih riječi, autora dokumenta, zadnje izmjene i drugih metapodataka te na taj način web dizajneri mogu sami preuzeti kontrolu nad okvirom za prikaz web stranice koja je korisniku vidljiva. (Slika 4.2.)

Korištena je meta oznaka koja određuje kodiranje znakova za HTML dokument („<meta charset='utf-8'>“). Druga meta oznaka je korisna jer se većinom odnosi na preglednik koji se koristi i koja prati zadnje promjene koje je Microsoft napravio svojim preglednicima. („meta http-equiv=“X-UA-Comatible“ content=“chrome-1“>). Određeni su meta podaci i spojen HTML dokument sa CSS-om koji će poslužiti za sami izgled web aplikacije.

```
<body>
    <div id="boxes">
        <div id="properties">
            <form id="frmTypeSelect">
                <label class="rad">
                    <input type="radio" class="radioBtnClass" id="rbtnEdit" name="type" value="edit">
                    <i></i> Edit<br>
                </label>
                <label class="rad">
                    <input type="radio" class="radioBtnClass" id="rbtnLine" name="type" value="line">
                    <i></i> Line<br>
                </label>
                <label class="rad">
                    <input type="radio" class="radioBtnClass" id="rbtnRectangle" name="type" value="rect">
                    <i></i> Rectangle<br>
                </label>
                <label class="rad">
                    <input type="radio" class="radioBtnClass" id="rbtnRectangleTrans" name="type" value="rect_transformed">
                    <i></i>Rectangle transformed<br>
                </label>
                <label class="rad">
                    <input type="radio" class="radioBtnClass" id="rbtnPolygon" name="type" value="polygon">
                    <i></i> Polygon<br>
                </label>
                <label class="rad">
```

Slika 4.3. *Prikazuje upotrebu div i class elemenata*

Korisniku je omogućeno da odabere osnovni grafički element s kojim će izrađivati svoju vlastitu vektorsku grafiku. Od ponuđenih grafičkih elemenata korisnik može odabrati liniju, pravokutnik,

transformirani pravokutnik, poligon, elipsu, kružnicu, poligon koji se kreće po *gridu*, poligon koji se kreće po *gridu*, a istovremeno držeći tipku CTRL. Ovakav način odabira različitih grafičkih elemenata je napravljen u tijelu HTML-a (unutar <body>...</body> elementa) koristeći različite klase i identifikatore. Kada dodajemo svojstvo *id* HTML elementu dodjeljujemo mu naziv identifikatora. (Slika 4.3.)

```
body, html{
    margin:0;
    width:100%;
    }
    #boxes>div {
        width: 50%;
        height: 50%;
    }
    #boxes {
        display:flex;
        align-items: flex-start;
        justify-content: right;
        text-align: -webkit-auto;
    }

    #palette {
        justify-content: right;
        text-align: right;
    }

    #logo {
        justify-content: center;
        text-align: center;
    }
```

Slika 4.4. *Prikaz CSS*

Prema slici 4.4. vidi se pristupanje u CSS-u oznaci *id* s početnim znakom *#*.

```
        </div>
        <div id="logo">
            <img src="../Multimedia/penLogo.jpg" height="100px" width="100px">
        </div>
        <div id="palette">
            <input id="colorPicker" class="jscolor" value="000000">
            <input type="hidden" id="hdfColor" value="000000">
        </div>
    </div>
```

Slika 4.5. *Prikazuje dodavanje loga i mogućnost odabira boje*

Prema slici 4.5. može se primjetiti da je web aplikaciji dodan logo kojem se pristupa preko elementa <img/> i kojem je dodjeljena vrijednost njegovog naziva. *Src* nam predstavlja ime svojstva i vrijednost mu je („../Multimedia/penLogo.jpg"). Slika je spremljena u istoj datoteci u kojoj je spremljena web aplikacija kako bih se slika mogla prikazati u web pregledniku.

Prije samog odabira grafičkog elementa korisnik može odabrati boju s kojom će crtati te mu je ponuđeno širok spektar boja i nijansa. Ako korisnik ne odabere boju onda mu je određena crna boja za crtanje. Nakon što korisnik odabere boju i grafički element s kojim će crtati na raspolaganju mu je radna površina namjenjena za crtanje i izrađivanje vektorke grafike. Nakon što korisnik nacrta grafički element implementirana mu je mogućnost da odabere drugi grafički element za crtanje ili da nastavi crtati s istim grafičkim elementom.

```
100         </div>
101         <div id="modal" class="modal">
102             <div class="modal-content">
103                 <div class="modal-labeled-input">
104                     <label for="editStrokeColor">Stroke color: </label>
105                     <input id="editStrokeColor" class="jscolor" value="FFFFFF">
106                 </div>
107                 <div class="modal-labeled-input">
108                     <label for="editFillColor">Fill color: </label>
109                     <input id="editFillColor" class="jscolor" value="FFFFFF">
110                 </div>
111                 <div class="modal-labeled-input">
112                     <label for="editStrokeWidth">Stroke width: </label>
113                     <input id="editStrokeWidth" value="0">
114                 </div>
115                 <hr>
116                 <div class="modal-buttons">
117                     <div>
118                         <button id="editDrawingButton">Update</button>
119                     </div>
120                     <div>
121                         <button id="deleteDrawingButton">Delete</button>
122                     </div>
123                     <div>
124                         <button id="cancelModalButton">Cancel</button>
125                     </div>
126                 </div>
127             </div>
128         </div>
```

Slika 4.6. *Edit opcija*

Prema slici 4.6. vidi se prikaz kôda koji služi korisniku kada želi uređivati grafički element. Nakon što korisnik odabere *edit* opciju i lijevim klikom miša pritisne na grafički element koji želi uređivati otvara se prozor u kojem je ponuđena mogućnost uređivanja elementa. Korisnik može promijeniti boju linije, ispuniti element bojom ili promjeniti debljinu linije. Nakon što je korisnik izabrao mogućnosti uređivanja ponuđena mu je mogućnost da primjeni promijenu na grafičkom elementu pod opcijom *update*. Korisnik isto tako može obrisati grafički element pod opcijom *delete* ili odustati od uređivanja pod opcijom *cancel*.

```
129
130         <script src="../Javascript/svg_min.js"></script>
131         <script src="../Javascript/svg.draw.min.js"></script>
132         <script src="../Javascript/dom_to_image_min.js"></script>
133         <script src="https://ajax.aspnetcdn.com/ajax/jQuery/jquery-3.4.0.min.js"></script>
134         <script src="../Javascript/jscolor.js"></script>
135         <script>
```

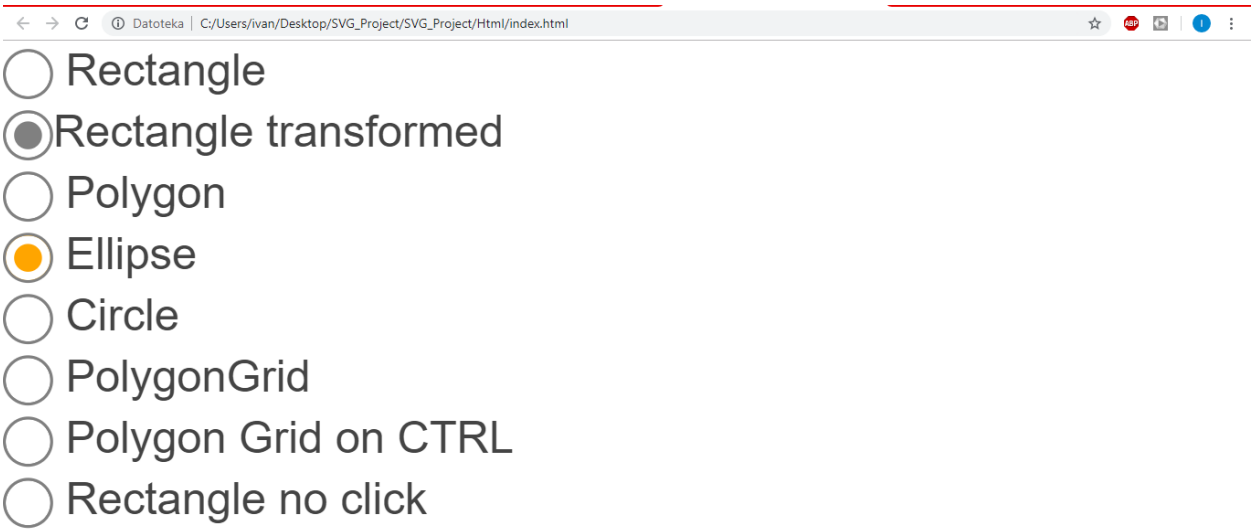
Slika 4.7. *Prikazuje implementiranje JavaScripta u HTML dokument*

Kako bih korisnik mogao uspješno nacrtati grafički objekt ovisno o njegovom odabiru u radu je bilo potrebno implementirati određeni JavaScript kôd. U ovoj web aplikaciji kôd je implementiran između html elementa <script>..<script> i koji se nalazi između <body>..</body> elementa. Implementirana je JavaScript ekstenzija koja nam služi za crtanje odnosno izradu osnovnih grafičkih elemenata. (Slika 4.7.)

```
215
216     switch(shape) {
217         case 'line':
218             currentDrawing = drawing.polygon().draw().attr('stroke-width',1).attr('fill','none').stroke({color: '#'+$('#hdfColor').val()});
219             addLineListeners();
220             break;
221         case 'rect':
222             currentDrawing = drawing.rect().draw().attr('stroke-width',1).attr('fill','none').stroke({ color: '#'+$('#hdfColor').val(), width: 5 });
223             break;
224         case 'rect_transformed':
225             currentDrawing = drawing.rect().draw().attr('stroke-width',1).attr('fill','none').stroke({color: '#'+$('#hdfColor').val()}).scale(4).rot
226             break;
227         case 'ellipse':
228             currentDrawing = drawing.ellipse().draw().attr('stroke-width',1).attr('fill','none').stroke({color: '#'+$('#hdfColor').val()});
229             break;
230         case 'circle':
231             currentDrawing = drawing.circle().draw().attr('stroke-width',1).attr('fill','none').stroke({color: '#'+$('#hdfColor').val()});
232             break;
233         case 'polygon':
234             currentDrawing = drawing.polygon().draw().attr('stroke-width',1).attr('fill','none').stroke({color: '#'+$('#hdfColor').val()});
235             addPolygonListeners();
236             break;
237         case 'polygonGrid':
238             currentDrawing = drawing.polygon().draw({snapToGrid:20}).attr('stroke-width',1).attr('fill','none').stroke({color: '#'+$('#hdfColor').va
239             addPolygonListeners();
240             break;
```

Slika 4.8. *switch funkcija*

Kada korisnik odabere grafički element za crtanje napravljena je funkcija koja određuje za sve grafičke elemente da imaju radnu površinu za crtanje veličine (100%, 100%), boja koja je postavljena na crno ili drugačije ako korisnik odabere drugu boju te određena je debljina linije za crtanje. (Slika 4.8.)



Slika 4.9. *Prikazuje odabir različitih grafičkih elemenata*

Prema slici 4.9. vidi se prikaz kada je korisnik izabrao elipsu za crtanje, a miš se nalazi na kružiću za odabir transformiranog pravokutnika. Funkcija služi kada korisnik želi odabrati grafički element za crtanje i kada s mišem prelazi preko praznih kružića koji služe za odabrati grafički element. Korisniku se u sivoj boji popuni kružić i kako s mišem prelazi preko različitih elemenata tako se različiti kružići popune i tako sve dok ne odabere grafički element za crtanje. Nakon toga se kružić oboji u narančastoj boji, a ostali kružići ostanu prazni pošto ih korisnih nije odabrao. Tako da korisnik zna u svakom trenutku koji je grafički element za crtanje izabrao.

```
188
189         // By changing color we cancel drawing and draw new with different color
190         $('#colorPicker').change(function() {
191             $('#hdfColor').val($('#colorPicker').val());
192             isDrawingFinished = false;
193             startDraw(chosenOption);
194             isDrawingFinished = true;
195         });
196
197         // By changing option we cancel drawing and start drawing new drawing of different shape
198         $('input[type=radio][name=type]').change(function(e){
199             isDrawingFinished = false;
200             startDraw(e.target.value);
201             isDrawingFinished = true;
202         });
203
```

Slika 4.10. *Prikazuje kôd za mjenjanje boje elementa*

Prema slici 4.10. se vidi prikaz kôda za mijenjanje boje elementa te kada korisnik želi promijeniti boju potrebno je trenutno crtanje prekinuti i započeti novo crtanje sa novom bojom koju je korisnik odabrao, ista stvar je i kod odabira drugog grafičkog elementa.

```
317             function keydownPolygon(e) {
318                 if(e.keyCode == 13){
319                     currentDrawing.draw('done');
320                     currentDrawing.off('drawstart');
321                 }
322             }
323             function keydownPolygonGridOnCtrl(e) {
324                 if(e.keyCode == 13){
325                     currentDrawing.draw('done');
326                     currentDrawing.off('drawstart');
327                 }
328                 if(e.keyCode == 17){
329                     currentDrawing.draw('param', 'snapToGrid', 20);
330                 }
331             }
332             function keyupPolygonGridOnCtrl(e) {
333                 currentDrawing.draw('param', 'snapToGrid', 1);
334             }
335
336
337             function openEditModal(drawing) {
338                 editedDrawing = drawing;
339
```

Slika 4.11. *Prikazuje funkciju kojom korisnik kada pitisne enter završava crtanje*

Kada korisnik pritisne *enter* završava crtanje trenutačnog elementa u ovom slučaju poligona i korisnik kada drži tipku ctrl ima mogućnosti crtati poligon po *gridu* i crtanje opet završava pritiskom tipke *enter*. (Slika 4.11.)

```
399
400         // Downloads svgElement as an image
401         function exportDrawing(svgElement) {
402             // construct dataUrl
403             const imageXml = new XMLSerializer().serializeToString(svgElement);
404             const imageBase64 = window.btoa(imageXml);
405             const dataUrl = '' + imageBase64;
406             // ask user for a file name of the file which will be downloaded
407             const fileName = prompt('Enter the name of the file');
408             // create download link with data url on the fly that is not seen on the page and click on it to download
409             var link = document.createElement('a');
410             link.download = fileName;
411             link.href = dataUrl;
412             link.click();
413         }
414
```

Slika 4.12. *Prikazuje kod koji služi za spremanje izrađene slike*

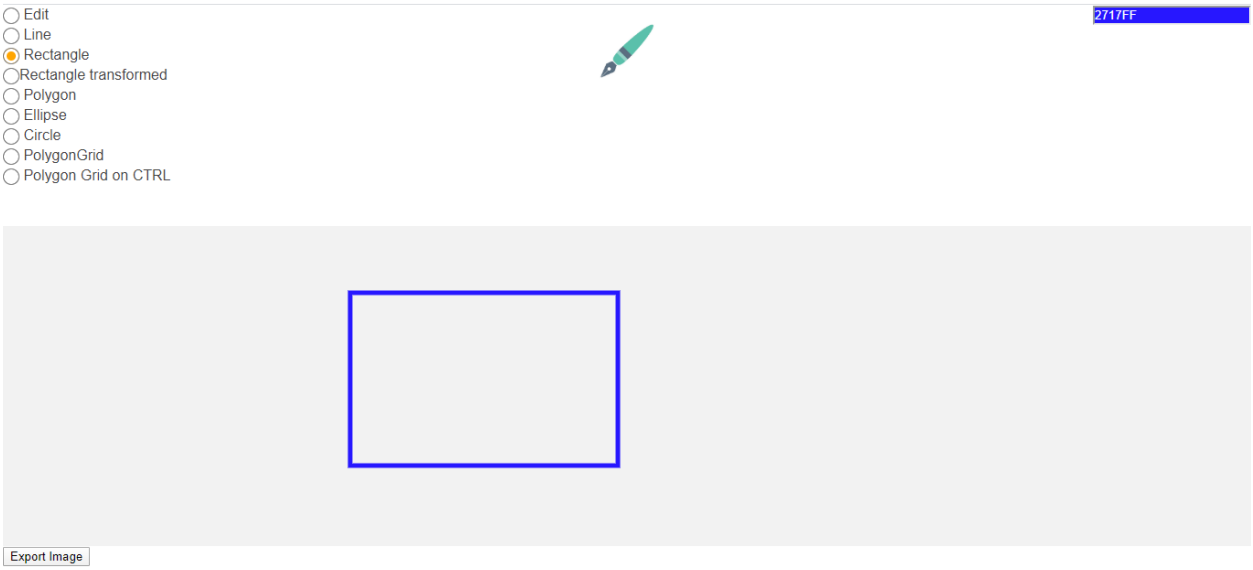
Korisnik nakon što izradi svoju sliku ima mogućnost spremiti sliku u SVG formatu i preuzeti je na svoje računalo. (Slika 4.12.)

4.4. Mogućnosti aplikacije

Korisnik može izabrati između osam različitih grafikih elemenata za crtanje te će se u ovom poglavlju proći kroz svaki izrađeni grafički element.

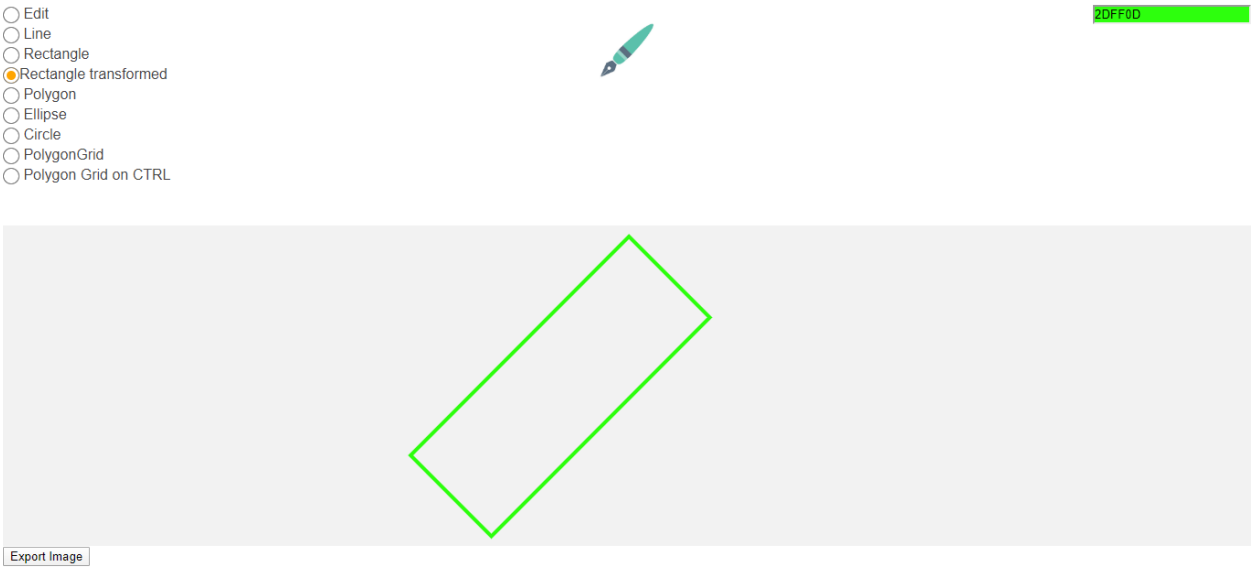
Slika 4.13. *Linija*

Prvi osnovni grafički element koji korisnik može izraditi je ravna linija. Prema slici 4.13. vidi se prikaz izrađene ravne linije u web aplikaciji za izradu vektorske grafike. Korisnik ima mogućnosti izraditi više ravnih linija na istom crtežu te korisnik može sam s mišom odrediti duljinu linije i u kojem smjeru će ravna linija biti okrenuta. Korisnik može prije izrade ravne linije promijeniti boju linije ili nakon što izradi ravnu liniju ima mogućnosti promijeniti boju linije, debljinu linije ili izbrisati nacrtani grafički element.



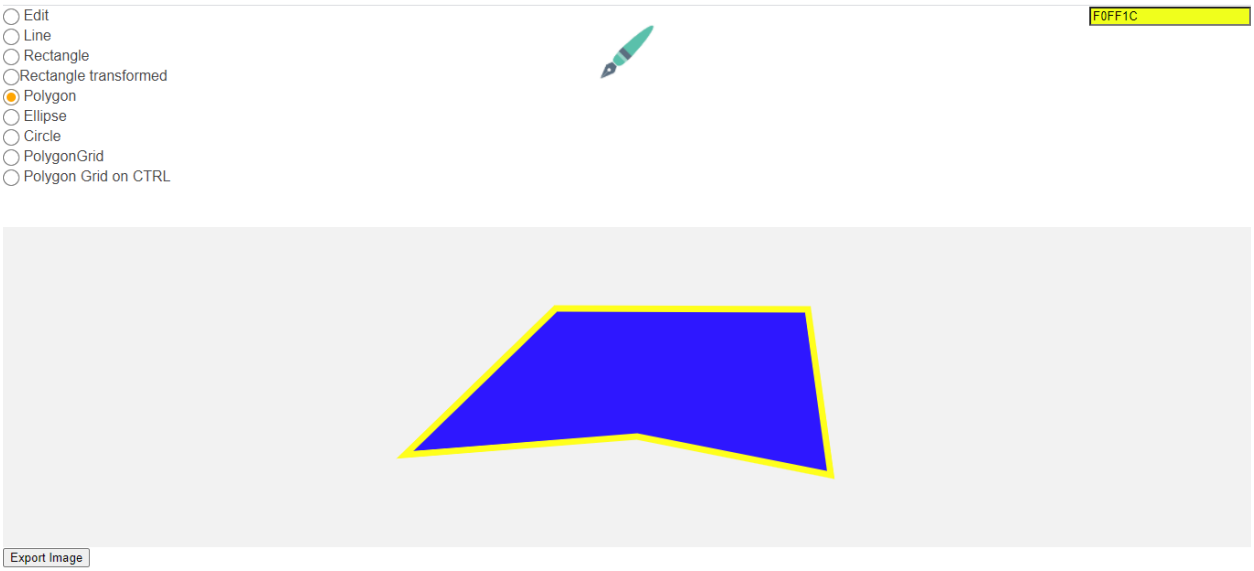
Slika 4.14. *Pravokutnik*

Drugi osnovni grafički element koji korisnik može izraditi je pravokutnik. Prema slici 4.14. vidi se prikaz izrađenog pravokutnika. Korisnik ima mogućnosti izraditi više pravokutnika na istom crtežu te korisnik može sam s mišom odrediti veličinu pravokutnika. Korisnik može prije izrade pravokutnika promijeniti boju pravokutnika ili nakon što izradi pravokutnik ima mogućnosti promijeniti boju linije pravokutnika, debljinu linije pravokutnika, ispuniti pravokutnik sa bojom ili izbrisati nacrtani pravokutnik.



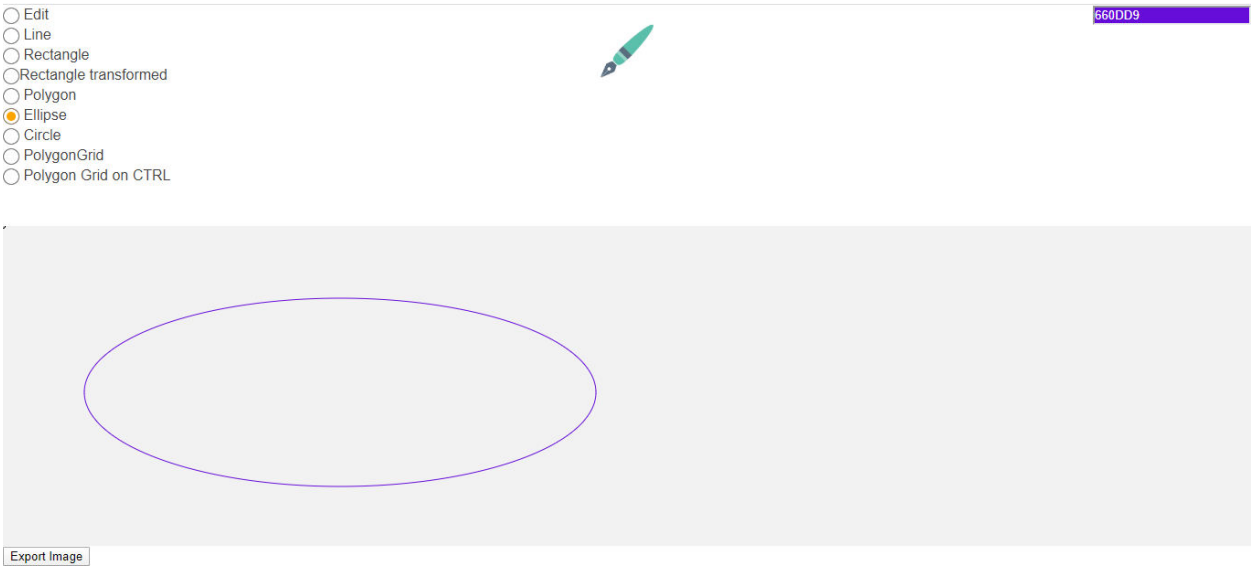
Slika 4.15. *Transformirani pravokutnik*

Treći osnovni grafički element koji korisnik može izraditi je transformirani pravokutnik. Prema slici 4.15. vidi se prikaz izrađenog transformiranog pravokutnika. Korisnik ima mogućnosti izraditi više transformiranih pravokutnika na istom crtežu te korisnik može sam s mišom odrediti veličinu transformiranog pravokutnika. Korisnik može prije izrade transformiranog pravokutnika promijeniti boju linije transformiranog pravokutnika. Nakon što izradi transformirani pravokutnik korisnik ima mogućnost promijeniti boju linije, debljinu linije elementa, ispuniti element sa bojom ili izbrisati nacrtani element.



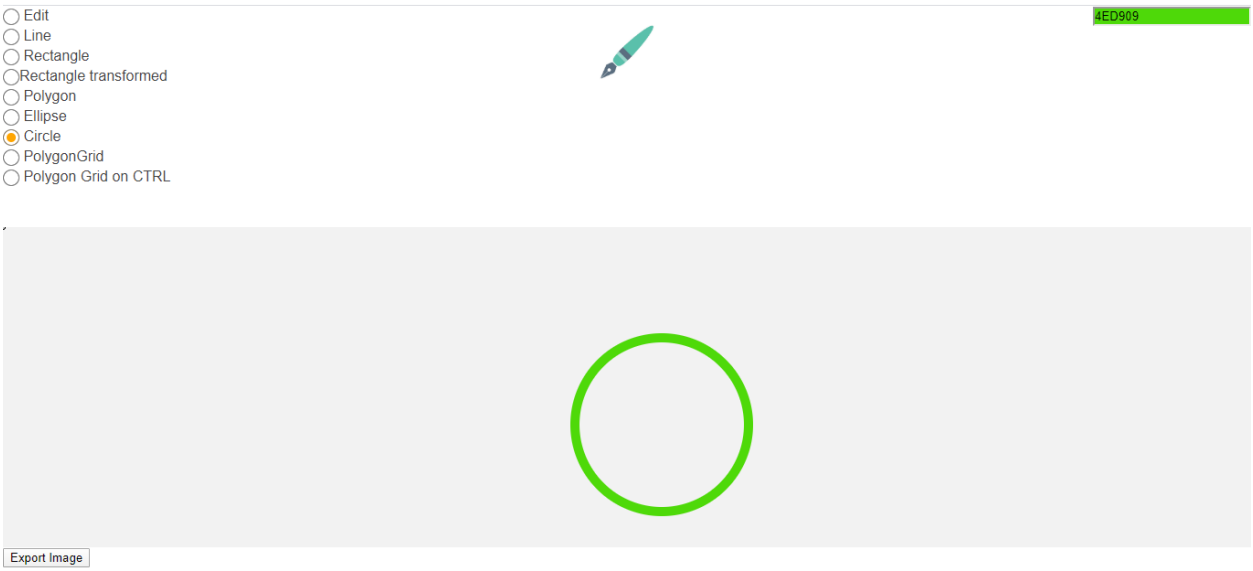
Slika 4.16. *Poligon*

Četvrti osnovni grafički element koji korisnik može izraditi je poligon. Prema slici 4.16. vidi se prikaz izrađenog poligona sa pet vrhova. Korisnik ima mogućnosti izraditi više poligona na istom crtežu te korisnik može sam s mišom odrediti broj vrhova poligona. Korisnik prvo treba odabrati poligon za crtanje te s lijevim klikom miša pritisnuti na površinu namijenjenu za crtanje u web aplikaciji. Nakon toga će jedan vrh biti označen te korisnik može u bilo kojem smjeru odrediti duljinu linije i označiti drugi vrh. Poligon označen sa dva vrha je isto što i izrađena ravna linija. Ako korisnik pritisne s lijevim klikom miša i označi jedan vrh i pritisne enter neće se izraditi ništa jer potrebno je minimalno dva vrha označiti da se izradi ravna linija. Korisnik može izraditi poligon sa koliko god želi vrhova te prije izrade poligona može promijeniti boju linije poligona. Nakon što izradi poligon korisnik ima mogućnosti promijeniti boju linije poligona, debljinu linije poligona, ispuniti poligon sa bojom ili izbrisati poligon.



Slika 4.17. *Elipsa*

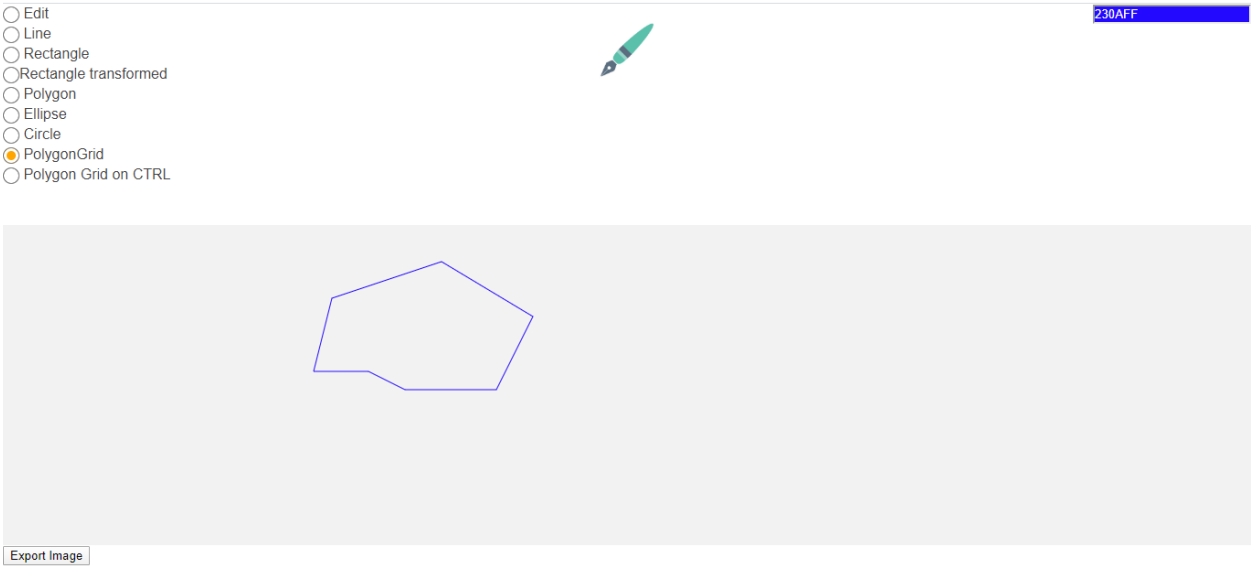
Peti osnovni grafički element koji korisnik može izraditi je elipsa. Prema slici 4.17. vidi se prikaz izrađene elipse. Korisnik ima mogućnosti izraditi više elipsa na istom crtežu te korisnik može sam s mišom odrediti veličinu elipse. Korisnik može prije izrade elipse promijeniti boju linije elipse. Nakon što izradi elipsu korisnik ima mogućnosti promijeniti boju linije elipse, debljinu linije elipse, ispuniti elipsu sa bojom ili izbrisati nacrtanu elipsu.



Slika 4.18. *Kružnica*

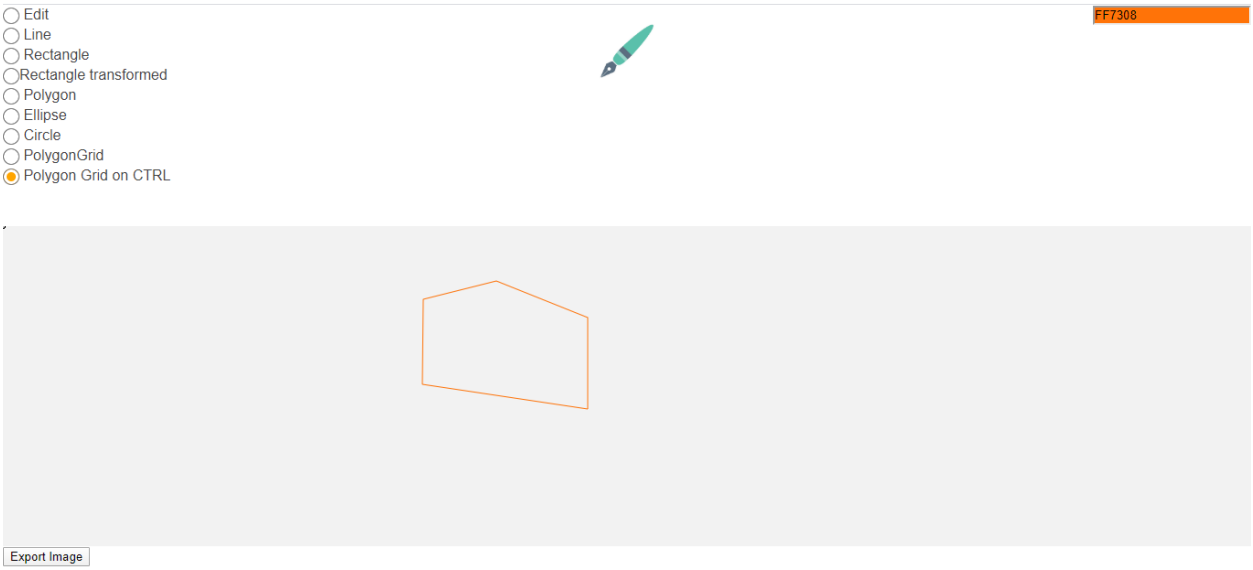
Šesti osnovni grafički element koji korisnik može izraditi je kružnica. Prema slici 4.18. vidi se prikaz izrađene kružnice. Korisnik ima mogućnosti izraditi više kružnica na istom crtežu te

korisnik može sam s mišom odrediti veličinu kružnice. Korisnik može prije izrade kružnice promijeniti boju linije kružnice. Nakon što izradi kružnicu korisnik ima mogućnosti promjeniti boju linije kružnice, debljinu linije kružnice, ispuniti kružnicu sa bojom ili izbrisati nacrtanu kružnicu.



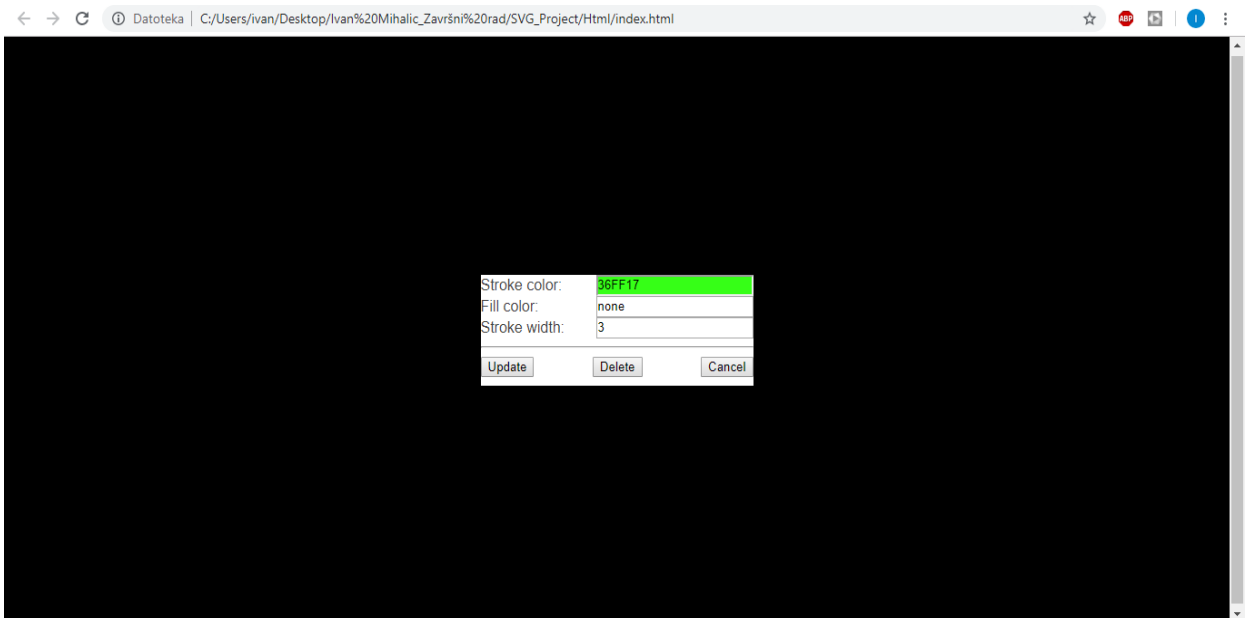
Slika 4.19. *Poligon po gridu*

Sedmi osnovni grafički element koji korisnik može izraditi je poligon po *gridu*. Prema slici 4.19. vidi se prikaz izrađenog poligona. Korisnik ima mogućnosti izraditi više poligona na istom crtežu. Glavna razlika između poligona i poligona po *gridu* je u tome što korisnik nakon što s lijevim klikom miša označi prvi vrh kao i kod poligona i nastavi se kretati mišem po površini za crtanje, miš će se kretati po *gridu*. *Grid* je pomočna pravokutna mreža točaka koju korisnik ne vidi, ali mu pomaže kod crtanja. Korisnik može prije izrade poligona promijeniti boju linije poligona. Nakon što izradi poligon korisnik ima mogućnosti promijeniti boju linije poligona, debljinu linije poligona, ispuniti poligon sa bojom ili izbrisati nacrtani poligon.



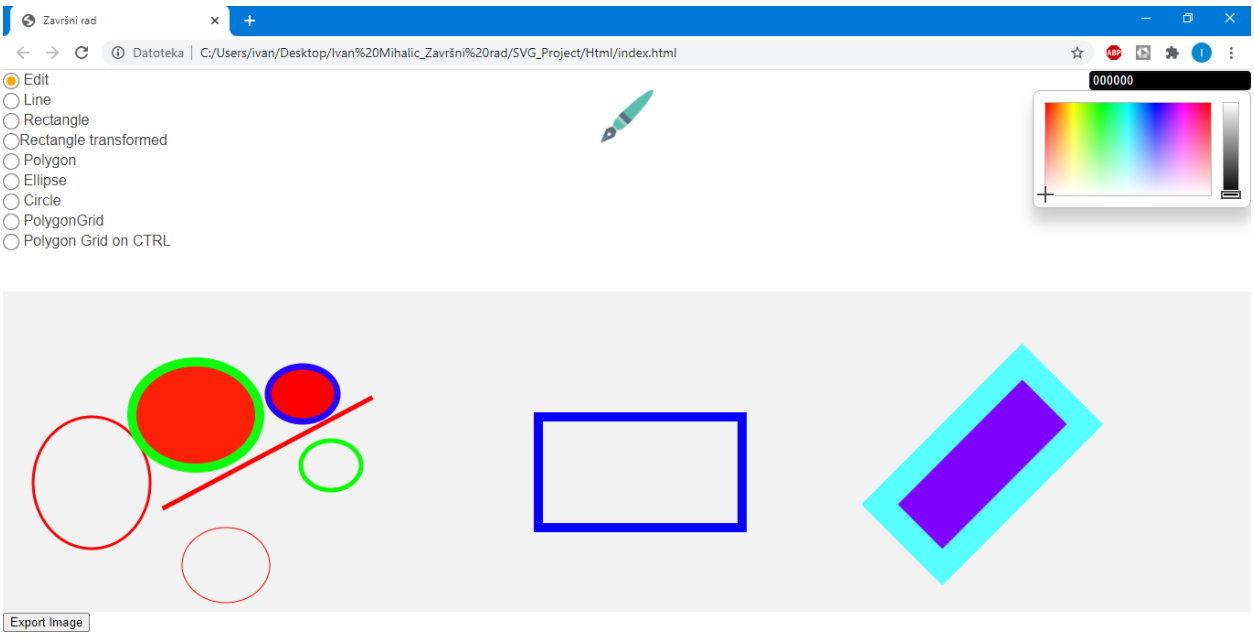
Slika 4.20. *Poligon po gridu uz pritisnutu tipku CTRL.*

Osmi osnovni grafički element koji korisnik može izraditi je poligon po *gridu* uz pritisnutku tipku *CTRL*. Prema slici 4.20. vidi se prikaz izrađenog poligona. Kao i kod prethodnog grafičkog elementa ovdje je bitna razlika što se korisniku nudi mogućnost crtanja po *gridu* samo kada drži tipku *CTRL* tako da korisnik može u svakom trenutku odabrati mogućnost koja mu najbolje odgovara. Korisnik može prije izrade poligona promijeniti boju linije poligona. Nakon što izradi poligon korisnik ima mogućnosti promijeniti boju linije poligona, debljinu linije poligona, ispuniti poligon sa bojom ili izbrisati nacrtani poligon.



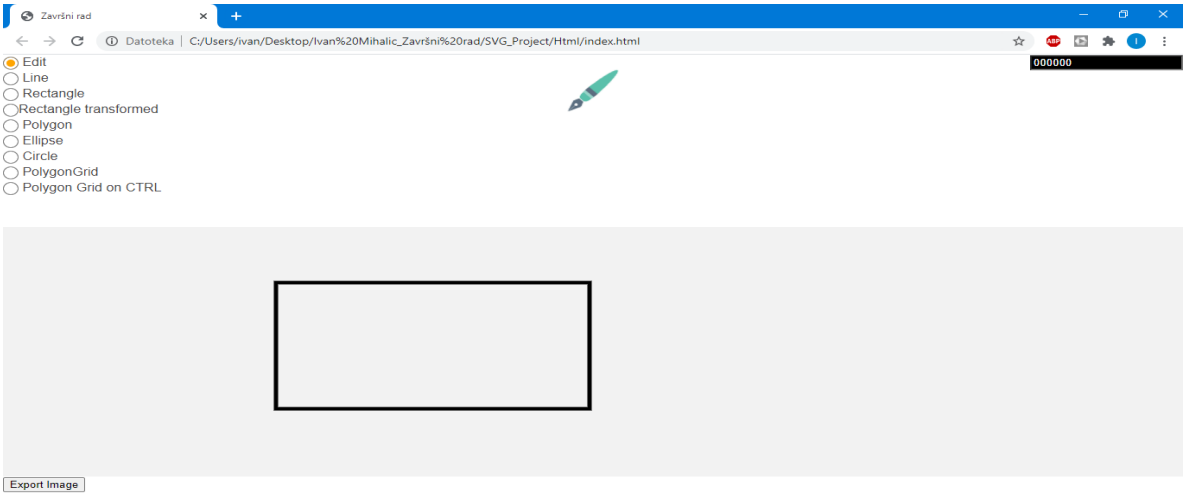
Slika 4.21. *Prikazuje prozor koji se otvori kada korisnik odabere da želi uređivati grafički element.*

Prema slici 4.21. se prikazuje prozor koji se korisniku otvori nakon što je pritisnuo *edit* opciju i s lijevim klikom miša pritisnuo na grafički element koji želi uređivati. Od ponuđenih opcija korisnik može promijeniti boju linije grafičkog elementa, ispunit grafički element bojom, promijeniti debljinu linije, primjeniti izmijenjenu promjenu na grafičkom elementu, obrisati grafički element ili odustati od naredbe uređivanja te se vratiti izradi vektorske grafike.

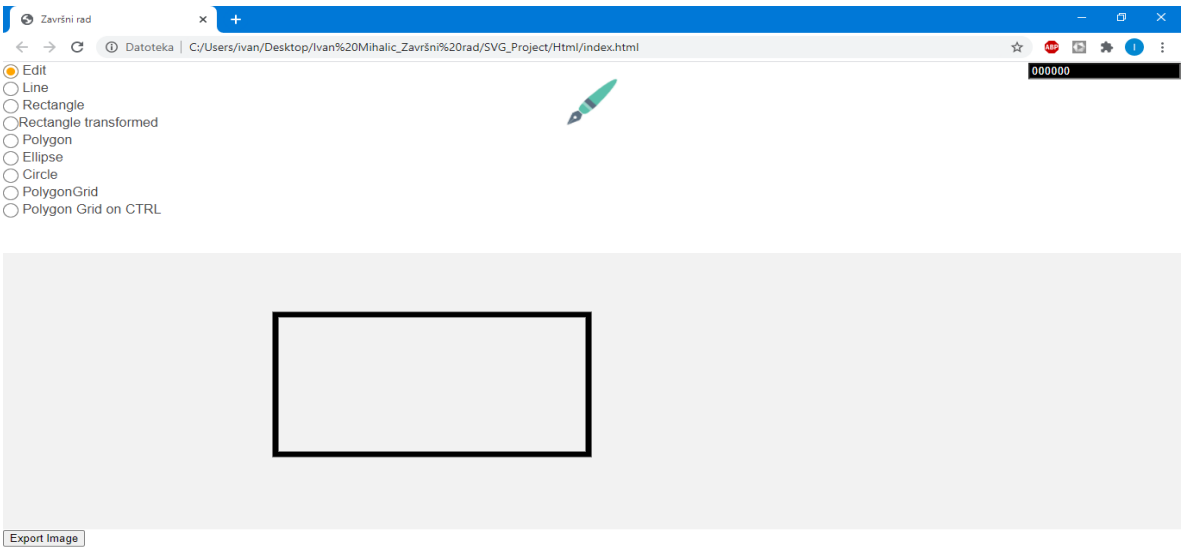


Slika 4.22. *web aplikacija s izborom različitih boja.*

Korisnik može odabrati širok spektar različitih boja i nijansi kao što je prikazano u gornjem desnom kutu. (Slika 4.22.). Korisnik također može izrađivati više različitih grafičkih elemenata na istom crtežu.

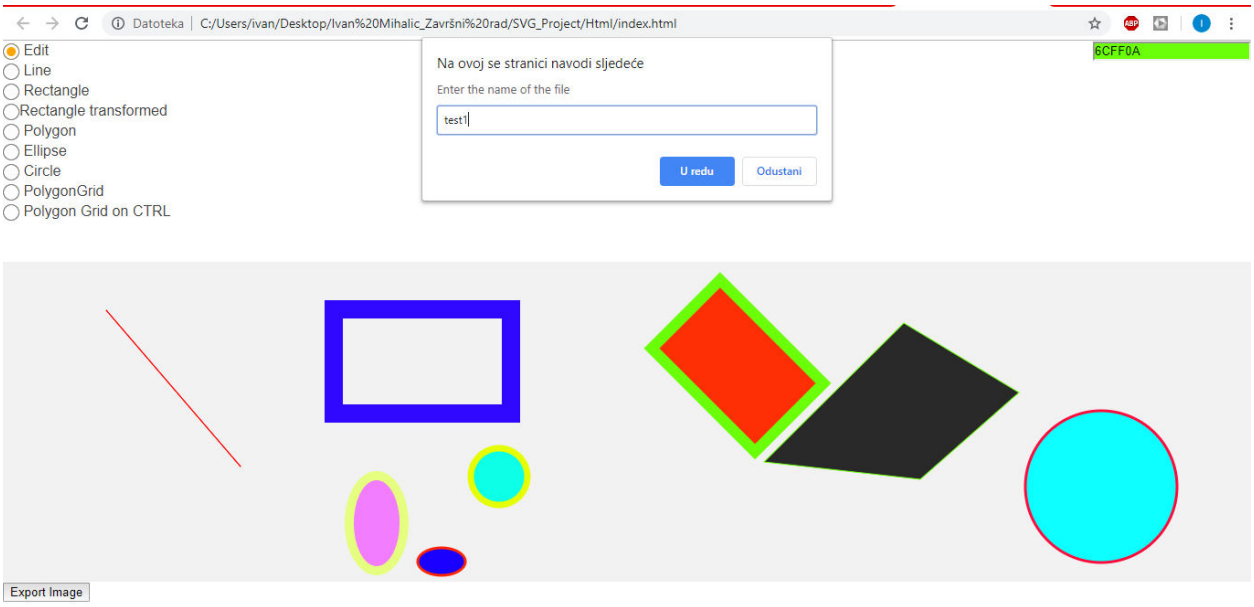


Slika 4.23. *Pravokutnik kada se miš ne nalazi na grafičkom elementu.*



Slika 4.24. *Pravokutnik s debljom linijom kada se miš nalazi na grafičkom elementu*

Prema slici 4.23. i 4.24. može se primjetiti razlika u debljini linije između dva pravokutnika. To se događa u slučaju kada korisnik želi uređivati grafički element i odabere *edit* opciju. Kada korisnik s mišom dođe do grafičkog elementa kojeg želi uređivati u tom trenutku se debljina linije malo uveća iz razloga kako bi korisnik lakše mogao primjetiti koji grafički element želi uređivati. U slučaju kada korisnik izradi jako puno različitih grafičkih elemenata koji su jako blizu jedan drugome ova opcija može biti od velike pomoći za lakše snalaženje u web aplikaciji i samoj izradi grafičkih elemenata.



Slika 4.25. *Prikazuje prozor koji se ponudi korisniku nakon što želi spremiti sliku.*

Korisnik prvo treba u donjem lijevom kutu stisnuti s lijevim klikom miša „Export Image“ i zatim će se otvoriti prozor koji se vidi prema slici 4.25. i koji pita korisnika koji naziv želi za svoju izrađenu sliku. Nakon što upiše naziv korisnik treba pritisnuti „U redu“ i automatski će se izrađena slika preueti na korisnikovo računalo sa unešenim nazivom u SVG formatu.

5. ZAKLJUČAK

U ovome radu bilo je potrebno izraditi web aplikaciju za izradu vektorske grafike uz pomoć SVG-a. Web aplikacija je napravljena i korisnik može izrađivati više različitih grafičkih elemenata te ih potom uređivati ovisno o njegovoj želji. Nakon što je korisnik gotov sa izradom vektorske graike napravljena je mogućnost da korisnik preuzme svoj rad na svoje računalo u SVG formatu te da datoteku nazove po vlastitoj želji. SVG se danas sve učestalije koristi, a razlog tomu je što svi moderni preglednici trenutno podržavaju SVG format. SVG-ovi se prikazuju u XML-u tako da je veličina datoteke stvarno mala. Time se povećava vrijeme učitavanja i performanse. SVG-ovi su potpuno skalabilni što znači da ih se može povećavati i smanjivati bez žrtvovanja kvalitete slike što je i dokazano u radu. Slike će također prikazati oštre slike na svim mrežnim uređajima, drugi formati slika zahtijevaju alternativnu ili veću veličinu slike da bi se postigla ista stvar. Kao nedostatak SVG-a može se istaknuti kako se samo radi s dvodimenzionalnim slikama, tako da fotografije ne rade u osnovi sve što bi se moglo snimiti fotoaparatom ili mobitelom.

LITERATURA

[1] Shelby Tupper, Inkscape Review | PCMag [online]

Dostupno na: https://www.pcmag.com/reviews/inkscape [Pristupljeno: 11.9.2020.]

[2] Inkscape dokumentacija, Pitanja i odgovori | Inkscape [online]

Dostupno na: https://inkscape.org/hr/o-programu/pregled/ [Pristupljeno: 11.9.2020.]

[3] Louie Andre, SVG-edit review: Pricing & Software Features 2020 [online]

Dostupno na: https://reviews.financesonline.com/p/svg-edit/ [Pristupljeno: 11.9.2020.]

[4] FinancesOnline, SVG-edit vs Inkscape 2020 Comparison [online]

Dostupno na: https://comparisons.financesonline.com/svg-edit-vs-inkscape [Pristupljeno: 11.9.2020.]

[5] Woodford, Chris, Computer graphics: Introduction and history – explain the staff [online],

Dostupno na: https://www.explainthatstuff.com/computer-graphics.html [Pristupljeno: 23.9.2020.]

[6] Mozilla Developer Network dokumentacija, SVG: Scalable Vector Graphics | MDN [online]

Dostupno na: https://developer.mozilla.org/en-US/docs/Web/SVG [Pristupljeno: 23.9.2020.]

[7] Mozilla Developer Network dokumentacija, HTML: Hypertext Markup Language | MDN [online],

Dostupno na: https://developer.mozilla.org/en-US/docs/Web/HTML[Pristupljeno: 23.9.2020.]

[8] Mozilla Developer Network dokumentacija, CSS: Cascading Style Sheets | MDN [online],

Dostupno na: https://developer.mozilla.org/en-US/docs/Web/CSS [Pristupljeno: 23.9.2020.]

[9] Mozilla Developer Network dokumentacija, JavaScript | MDN [online]

Dostupno na: https://developer.mozilla.org/en-US/docs/Web/JavaScript [Pristupljeno: 23.9.2020.]

[10] Ilya Kantor, An Introduction to JavaScript [online]

Dostupno na: https://javascript.info/intro [Pristupljeno: 23.9.2020.]

[11]Microsoft, XML za početnike – podrška za Office [online]

Dostupno na: https://support.microsoft.com/hr-hr/office/xml-za-po%C4%8Detnike-a87d234d-4c2e-4409-9cbc-45e4eb857d44 [Pristupljeno: 23.9.2020.]

[12] WordPress VIP, TechCrunch [online]

Dostupno na: https://techcrunch.com/2015/04/29/microsoft-shocks-the-world-with-visual-studio-code-a-free-code-editor-for-os-x-linux-and-windows/ [Pristupljeno: 23.9.2020.]

POPIS UPOTREBLJENIH OZNAKA

PDF – *Portable Document Format*

SVG – *Scalable Vector Graphic*

XML – *EXtensible Markup Language*

HTML – *Hyper Text Markup Language*

CSS – *Cascading Style Sheets*

W3C – *World Wide Web Consortium*

JPEG - *Joint Photographic Experts Group*

PNG – *Portable Network Graphics*

GIF - *Graphics Interchange Format*

TIFF - *Tagged Image File Format*

SAŽETAK

U ovom završnom radu bilo je potrebno napraviti web aplikaciju za izradu vektorske grafike. Viđene su tehnologije koje su se koristile i njihove mogućnosti. SVG se koristio za izradu vektorske grafike. HTML, CSS i JavaScript kao programerski jezici su se koristili za izradu web aplikacije. Visual Studio Code je služio kao softversko rješenje u kojem se je kôd pisao i kompajlirao. Kroz cijeli rad je objašnjeno kako se web aplikacija napravila i detaljno je prikazano sa tekstom i slikama kako je kôd redom napisan. U ovom radu objasnila se bitna razlika između rasterske i vektorske grafike u primjerima sa slikama. Za kraj možemo reći kako SVG ima više prednosti nego mana i kako će se njegovo korištenje i dalje sve više upotrebljavati za razvoj vektorske grafike u budućnosti.

Ključne riječi:

SVG, HTML, CSS, JavaScript, XML, Visual Studio Code, rasterska grafika, vektorska grafika.

ABSTRACT

Title: „WEB APPLICATION FOR CREATING VECTOR GRAPHICS“

In this project, it was necessary to create a web application for the creation of vector graphics. The technologies that were used and their capabilities were also observed. SVG was used to create vector graphics. HTML, CSS, and JavaScript as programming languages were used to create a web application. Visual Studio Code served as a software solution in which the code was written and compiled. Throughout the work it is explained how the web application has been made and it is detailed in text and images as it is written in a row. This paper explains the essential difference between raster and vector graphics in the examples with the images. For the end we can say that SVG has more prons than cons and how its use will increasingly be used to develop vector graphics in the future.

Keywords:

SVG, HTML, CSS, JavaScript, XML, Visual Studio Code, Raster Graphics, Vector Graphics.

ŽIVOTOPIS

Ivan Mihalić rođen je 14.ožujka 1996. godine u Sisku. Od 2003. do 2011. godine pohađa Osnovnu školu Mate Lovraka u Kutini. Godine 2011. upisuje Tehničku školu smjer tehničar za računalstvo u Kutini koju završava 2015. godine polaganjem ispita državne mature. Godine 2015. nakon završetka srednjoškolskog obrazovanja upisuje preddiplomski stručni studij Informatike na Elektrotehničkom fakultetu, današnjem Fakultetu računarstva, elektrotehnike i informacijskih tehnologija u Osijeku.

Vlastoručni potpis:

_____