SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU

Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek

Projektni zadatak iz predmeta

VIZUALIZACIJA PODATAKA

Prikaz učestalosti potresa u Hrvatskoj po godinama

Student: Bernard Barišić, DRC

Mentor: Barbara Bilonić

U Osijeku, rujan 2023.

SADRŽAJ

[**1. KV1 - Definiranje projektnog zadatka 3**](#_4i6s1smptslt)

[1.1. Projektni zadatak 3](#_wgoiizki7jnh)

[1.2. Podatci 3](#_jjnsdui0w5uv)

[1.3. Obrada podataka 3](#_jfu6ybs7kwrw)

[1.4. Relevantne vrste prikaza za korištene podatke 3](#_1g4rlc1z1va6)

[**2. KV2 - Dizajn vizualizacije podataka. 4**](#_smxrkz6d6wy8)

[2.1. Pitanja na koja vizualizacija daje odgovor 4](#_cqmiylhgneug)

[2.2. Skica vizualizacije podataka 4](#_i61spu8qyxmd)

[2.3. Postojeća rješenja i primjeri 4](#_qmiyijy4jsp8)

[2.4. Prilagodba podataka 4](#_cch1vltulvr)

[2.5. Boje i podatci 4](#_w22k3frzbtb2)

[**3. KV3 - Izrada prototipne vizualizacije podataka 5**](#_k5q372z1w040)

[3.1. Osnovne funkcionalnosti i ponašanja 5](#_yaexlmqqdt0u)

[3.2. Napredne funkcionalnosti i ponašanja: 5](#_y300tbmoybm5)

[3.3. Implementacija osnovnih funkcionalnosti 5](#_g3q3h1rhqzyz)

[3.4. Implementacija osnovnog ponašanja 5](#_itz19pnh67g0)

[**4. KV4 - Izrada konačne vizualizacije podataka 6**](#_sj4kshcbtchw)

[4.1. Implementacija osnovnih funkcionalnosti 6](#_dc19rzmsn9k5)

[4.2. Implementacija osnovnog ponašanja 6](#_exvg8zl4eo3w)

[4.3. Implementacija naprednih funkcionalnosti 6](#_ackz82cvngj2)

[4.4. Implementacija naprednog ponašanja 6](#_le6sbg8zsugz)

[**5. KV5 - Dovršetak projektnog zadatka i pisanje dokumentacije 7**](#_woao7m26bf75)

[5.1. Eventualne preinake i dorade rješenja - u dogovoru s nastavnikom 7](#_116xkgvrmrqh)

[5.2. Izrada dokumenta - projektne dokumentacije 7](#_6zplla712qe7)

[**Literatura 8**](#_ahmk8vkny1f9)

[**Prilog I 9**](#_jefyptb5hazf)

# KV1 - Definiranje projektnog zadatka

## Projektni zadatak

Naziv zadatka: Prikaz učestalosti potresa u Hrvatskoj po godinama

Opis problema: Nesortirane podatke o potresima u Hrvatskoj sortirati i prikazati vizualno.

Opis zadatka: Vizualni prikaz učestalosti potresa u Hrvatskoj u periodu od 2000. do 2023 godine.

Cilj projekta: Cilj ovog projekta je prikupiti podatke o potresima u Hrvatskoj(2000-2023 godine), sortirati podatke i programski prikazati vizualizaciju samih podataka.

Poveznica na git repozitorij projekta: https://github.com/Bernard137/VP\_Projekt.git

## Podatci

* + 1. Kao izvor podataka koristio sam stranice koje imaju dostupne informacije o potresima.
    2. Izvori koji su korišteni za ovaj projekt nalaze se na stranici <https://datahub.io/core/geo-countries#curl>, https://www.usgs.gov/programs/earthquake-hazards i dostupni su u CSV formatu i mnogim drugim formatima.

## Obrada podataka

* + 1. Podaci se obrađuju i povezuju pomoću geojson i json formata koji pripadaju programskom jeziku Javascript dostupni u biblioteci d3.js.

## Relevantne vrste prikaza za korištene podatke

* + 1. Koristit će se buttoni, barplot i karta Hrvatske.

# KV2 - Dizajn vizualizacije podataka.

## Pitanja na koja vizualizacija daje odgovor

Broj potresa u određenoj godini i jačina samih potresa.

## Skica vizualizacije podataka

## A screenshot of a computer Description automatically generatedPostojeća rješenja i primjeri

* + 1. Primjeri riješenih projekata: https://seismo.berkeley.edu/seismo.real.time.map.html https://ds.iris.edu/ieb/index.html?format=text&nodata=404&starttime=1970-01-01&endtime=2025-01-01&minmag=0&maxmag=10&mindepth=0&maxdepth=900&orderby=time-desc&src=usgs&limit=1000&maxlat=42.01&minlat=34.28&maxlon=29.32&minlon=11.26&zm=7&mt=ter
    2. Korištene stranice: <https://d3-graph-gallery.com/barplot.html>, https://www.tutorialsteacher.com/d3js/create-bar-chart-using-d3js, https://d3-graph-gallery.com/graph/interactivity\_button.html, https://codepen.io/babibn/pen/jOMWLLM
    3. Gore navedeni linkovi će poslužiti za izradu barplota,mape Hrvatske i buttona.

## Prilagodba podataka

* + 1. Podaci koji će se koristiti za obradu podataka će biti u excel,geojson,json formatu.
    2.  A close up of numbers

       Description automatically generated
    3. A screenshot of a computer

       Description automatically generated

## Boje i podatci

* + - 1. Boja za button: white, boja za mapu: #3d3f41, boja pozadine: #cee3f0, boja barplota: #F7FABC(Weak), #DDF272(Light), #F39C0D(Moderate), #DC4718(Strong)

# KV3 - Izrada prototipne vizualizacije podataka

## Osnovne funkcionalnosti i ponašanja

* + 1. Odabir “play” buttona ili button sa nazivom godine u sebi.
    2. Prikazati podatke na temelju korisnikova odabira.
    3. Korisnik odabire godinu ili play button, na temelju odabranog, prikazuje se sadržaj.

## Napredne funkcionalnosti i ponašanja:

* + 1. Prilikom pritiska na button“play” u bilo kojem trenutku se može zaustaviti prikaz klikom na button “pause”.
    2. Klikom na barplot možemo odabrati koje podatke želimo vidjeti na mapi.
    3. Kao što je već prethodno navedeno, korisnik može koristiti barplot tako što će prelaskom miša preko stupca dobiti prikaz podataka.

## Implementacija osnovnih funkcionalnosti

* + 1. Prikaz na način kako je objašnjeno u 3.2.3 i radi skraćivanja koda naveden je samo jedan primjer u ovom slučaju “Weak” slučajA screenshot of a computer program

       Description automatically generated
    2. Prelaskom miša preko stupca “Weak” dobijamo informacije o broju potresa i jacini magnitude i to korištenjem gore navedenog koda. A graph with text and numbers

       Description automatically generated

## Implementacija osnovnog ponašanja

* + 1. Struktura od HTML-a se sastoji od naslova, elemenata ‘<div>’ za gumbe, stupčasti graf, ispis trenutne godine I geografski prikaz. Svakom gumbu je dodjeljena pripadajuća klasa u skladu s godinom na koju se odnosi. Sadrži i atribut ‘onclick’ koji prilikom klika poziva funckiju za iscrtavanje stupčastog grafa s odgovarajućim parametrom.

ttA screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

-CSS dio koji je proizvoljno napravljen za potrebe programa.

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

-sa prve slike vidimo da je definirana širina I visina svg elemenata te varijabla ‘projection’ koja će biti korištena prilikom iscrtavanja geografskog prikaza. Također JSON podaci o potresima nalaze se u konstanti ‘dataCSV’. Ti podaci su potrebni kako bi JSON mogao iscrtati stupčasti graf

-Za svaku godinu se uzima svaka grupa I za rezultat se dobiva brojčana vrijednost koliko potresa te grupe ima svaka godina. Tako će konstanta ‘data’ sadržavati vrijednosti kao što je navedeno na slici 2 (year,group i value) pri čemu će za svaku jedinstvenu kombinaciju godine I grupe postojati ‘value’ koji govori koliko potresa te grupe određena godina sadrži.Shodno tome prosljeđujemo GeoJSON dokument funkciji za korištenje. Sa funkcijom ‘filter’ odabiru se samo podaci koji predstavljaju geografski prikaz Hrvatske.

# KV4 - Izrada konačne vizualizacije podataka

## Implementacija osnovnih funkcionalnosti

* + 1. ‘Color’ dodjeljuje boje krugovima koji predstavljaju potres. Konstanta ‘size’ vrijednostima od 2.5 do 6.4(magnitude) daje vrijednost u pikselima od 4 do 30.Funkcija ‘zoom’ prilikom svakog zumiranja i pomicanja poziva funkciju ‘updateCircles()’ i ona ažurira elemene x,y,z na način da skalira veličinu krugova na temelju iznosa magnitude podijeljenim s trenutnom vrijednosti k koja predstavlja trenutno mjerilo(početna vrijednost od k svg elementa iznosi 1).’Mouseover’,’mousemove’ i ‘mouseleave’ definiraju funkcije za prikazivanje željenog opisa prilikom prelaska miša. A white background with text

       Description automatically generated
    2. Prikaz gore navedenog koda.A map of the country

       Description automatically generated

## Implementacija osnovnog ponašanja

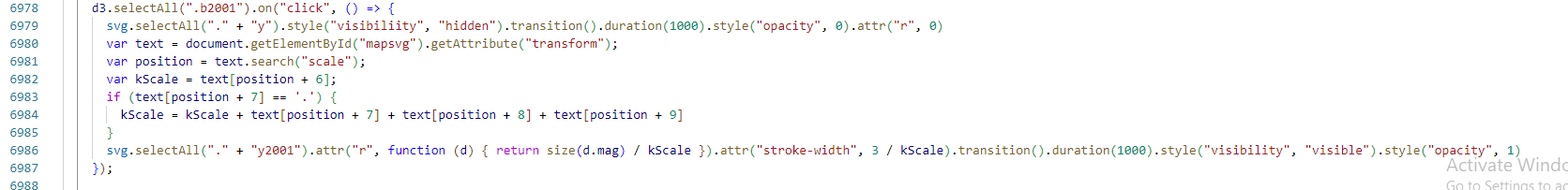
* + 1. Podaci spremljeni kao JSON se dodjeljuju svg elementu koji se koriste za dodjeljivanje vrijednosti atributima za svaki krug. Primjerice vrijednosti 'cx' je dodijeljeno 'projection([d.long, d.lat])[0]', vrijednosti 'cy' je dodijeljeno 'projection([d.long, d.lat])[1]', vrijednosti 'r' je dodijeljeno 'size(d.mag)' i slično.Zadnja linija koda sa slike prikazuje krugove za 2000.godinu koji su prikazani prilikom pokretanja stranice.A screenshot of a computer

       Description automatically generated
    2. Prikaz početne stranice sa kodom umetnutim sa prethodne slike.A screenshot of a map

       Description automatically generated

## Implementacija naprednih funkcionalnosti

* + 1. Prikaz korištenja funkcionalnosti ‘PLAY’ i ‘PAUSE’ buttona.
    2. A screenshot of a computer

       Description automatically generated
    3. Navedeni kod odabire button te na klik izdvaja i sprema vrijednost ‘k’ koja predstavlja trenutno mjerilo svg elementa. Nakon toga se odaberu odgovarajući krugovi prema godini te im se dodjeljuju potrebni atributi, a ti sami atributi će za radijus imati vrijednosti magnitude podijeljen sa trenutnom vrijednosti mjerila ‘k’ kako bi bio pravilno skaliran u odnosu na geografski prikaz. Također kod za ostale buttone(buttoni sa godinama) je analogan ovom dole navedenom. 
    4. U ovom dijelu dodajemo stupčasti graf koji ima definiranu širinu i visinu.Osi x je dodijeljena vrijednost x koja sadrži vrijednosti JSON dokumenta, u ovom slučaju(Weak,Light,Moderate,Strong).Funkcija ‘barplot(data)’ poziva se svakim klikom gumba za godinu I njen zadatak je da osvježi elemente stupčastog grafa.Također kod u nastavku prikazuje opis prilikom prelaska miša preko pojedinog stupca. Također je definirana logika odabira pojedinog stupca čime se filtrira prikaz krugova na geografskom prikazu.

A screenshot of a computer program

Description automatically generatedA screenshot of a computer

Description automatically generated

## Implementacija naprednog ponašanja

* + 1. Potvrda ispravnosti gore navedenog koda.A screenshot of a computer screen

       Description automatically generated

# KV5 - Dovršetak projektnog zadatka i pisanje dokumentacije

## Izrada dokumenta - projektne dokumentacije

* + 1. Projekt se sastoji od više datoteka(html,excel).
    2. d3.js .
    3. Pokrenuti html datoteku pod imenom ‘Earthquake CRO.html’

# Literatura

Priručnik za LV

<https://datahub.io/core/geo-countries#curl>

<https://d3-graph-gallery.com/barplot.html>

<https://www.tutorialsteacher.com/d3js/create-bar-chart-using-d3js>

<https://d3-graph-gallery.com/graph/interactivity_button.html>

https://codepen.io/babibn/pen/jOMWLLM

<https://earthquake.usgs.gov/earthquakes/search/>

<https://hr.wikipedia.org/>

https://ds.iris.edu/ieb/index.html?format=text&nodata=404&starttime=1970-01-01&endtime=2025-01-01&minmag=0&maxmag=10&mindepth=0&maxdepth=900&orderby=time-desc&src=usgs&limit=1000&maxlat=42.01&minlat=34.28&maxlon=29.32&minlon=11.26&zm=7&mt=ter

# Prilog I

Poveznica na git repozitorij projekta: https://github.com/Bernard137/VP\_Projekt.git

Programski kod:

<!DOCTYPE html>

<meta charset="utf-8">

<script src="https://d3js.org/d3.v6.js"></script>

<body>

  <div>

    <button class="btn b2000" onclick="barplot(dataBar[0])">2000</button>

    <button class="btn b2001" onclick="barplot(dataBar[1])">2001</button>

    <button class="btn b2002" onclick="barplot(dataBar[2])">2002</button>

    <button class="btn b2003" onclick="barplot(dataBar[3])">2003</button>

    <button class="btn b2004" onclick="barplot(dataBar[4])">2004</button>

    <button class="btn b2005" onclick="barplot(dataBar[5])">2005</button>

    <button class="btn b2006" onclick="barplot(dataBar[6])">2006</button>

    <button class="btn b2007" onclick="barplot(dataBar[7])">2007</button>

    <button class="btn b2008" onclick="barplot(dataBar[8])">2008</button>

    <button class="btn b2009" onclick="barplot(dataBar[9])">2009</button>

    <button class="btn b2010" onclick="barplot(dataBar[10])">2010</button>

    <button class="btn b2011" onclick="barplot(dataBar[11])">2011</button>

    <button class="btn b2012" onclick="barplot(dataBar[12])">2012</button>

    <button class="btn b2013" onclick="barplot(dataBar[13])">2013</button>

    <button class="btn b2014" onclick="barplot(dataBar[14])">2014</button>

    <button class="btn b2015" onclick="barplot(dataBar[15])">2015</button>

    <button class="btn b2016" onclick="barplot(dataBar[16])">2016</button>

    <button class="btn b2017" onclick="barplot(dataBar[17])">2017</button>

    <button class="btn b2018" onclick="barplot(dataBar[18])">2018</button>

    <button class="btn b2019" onclick="barplot(dataBar[19])">2019</button>

    <button class="btn b2020" onclick="barplot(dataBar[20])">2020</button>

    <button class="btn b2021" onclick="barplot(dataBar[21])">2021</button>

    <button class="btn b2022" onclick="barplot(dataBar[22])">2022</button>

    <button class="btn b2023" onclick="barplot(dataBar[23])">2023</button>

  </div>

  <h1 id="naslov">Potresi u Hrvatskoj za godinu: </h1>

  <div id="vizdata"></div>

  <div>

    <button class="btn play">PLAY</button>

  </div>

  <h1 id="current">2000</h1>

  <div id="map">

    <svg id="mapsvg" width="1000" height="750" transform="scale(1)"></svg>

  </div>

</body>

<style>

  #map {

    position: absolute;

    margin-left: 750px;

    margin-top: -430px;

    border: 2px solid #1f1e1e;

    overflow: hidden;

    background-color: #3d3f41;

  }

  #naslov {

    position:absolute;

    margin-left: 1000px;

  }

  #current {

    position: absolute;

    margin-left: 1430px;

    margin-top: -482px;

  }

  body {

    background-color: #cee3f0;

    font-family:'Nunito';

    font-weight: bold;

  }

  .btn {

    background-color: white;

    border: 2px solid #e7e7e7;

    padding: 16px 16px;

    text-align: center;

    text-decoration: none;

    font-weight: bold;

    display: inline-block;

    font-size: 15px;

    margin: 4px 2px;

    transition-duration: 0.4s;

    cursor: pointer;

  }

  .play {

    position: absolute;

    margin-left: 240px;

    margin-top: 5px;

    width: 100px;

  }

  .btn:hover {

    background-color: #e7e7e7;

  }

  circle:hover {

    fill-opacity: 1;

  }

</style>

<script>

  const svg = d3.select("svg"),

    width = +svg.attr("width"),

    height = +svg.attr("height");

  var projection = d3.geoMercator()

    .center([16.5, 44.55])

    .scale(6500)

    .translate([width / 2, height / 2])

  // Filtrirani podaci iz CSV excel formata skinuti sa 'https://www.usgs.gov/programs/earthquake-hazards' stranice

  const dataCSV = [ {

      "year": 2000,

      "lat": 45.427,

      "long": 14.676,

      "mag": 2.5,

      "group": "Weak",

      "place": "10 km WNW of Delnice, Croatia"

    },

]; // radi skračivanja koda ovdje je naveden samo jedan primjer podataka koji će biti korišteni

function calcSums(entries) {

    var result = {};

    entries.forEach(function (val) {

      var key = val.year + ":" + val.group;

      result[key] = result[key] || { year: val.year, group: val.group, value: 0 };

      result[key].value++;

    });

    return Object.keys(result).map(function (key) {

      return (result[key]);

    });

  }

  const data = JSON.stringify(calcSums(dataCSV), null, 2);

  console.log(data);

  const dataBar = [];

  for (i = 0; i < 24; i++) {

    var value = 2000 + i;

    dataBar[i] = JSON.parse(data).filter(({ year }) => year === value);

  }

  // izvor za geojson data: 'https://datahub.io/core/geo-countries#curl'

  d3.json("https://datahub.io/core/geo-countries/r/0.geojson").then(function (data) {

    data.features = data.features.filter(d => d.properties.ADMIN == "Croatia")

    const color = d3.scaleOrdinal()

      .domain(["Weak", "Light", "Moderate", "Strong"])

      .range(["#F7FABC", "#DDF272", "#F39C0D", "#DC4718"])

    const size = d3.scaleLinear()

      .domain([2.5, 6.4])

      .range([4, 30])

    const zoom = d3.zoom()

      .on('zoom', updateCircles)

    function updateCircles(event, d) {

      svg.attr('transform', event.transform);

      svg.selectAll("." + "y" + document.getElementById("current").textContent).attr("r", d => size(d.mag) / (event.transform.k).toFixed(2)).attr("stroke-width", 3 / (event.transform.k).toFixed(2))

    }

    var map = d3.selectAll("#map")

      .call(zoom)

    svg

      .append("g")

      .selectAll("path")

      .data(data.features)

      .join("path")

      .attr("fill", "#b8b8b8")

      .attr("d", d3.geoPath()

        .projection(projection)

      )

      .style("stroke", "black")

      .style("opacity", .3)

    var mouseover = function (event, d) {

      tooltip.style("visibility", "visible")

    }

    var mousemove = function (event, d) {

      tooltip

        .html("Mjesto: " + d.place + "<br>" + "Jacina magnitude: " + d.mag)

        .style("top", (event.pageY - 100) + "px")

        .style("left", (event.pageX + 50) + "px");

    }

    var mouseleave = function (event, d) {

      tooltip.style("visibility", "hidden")

    }

    svg

      .selectAll("myCircles")

      .data(dataCSV)

      .enter()

      .append("circle")

      .attr("class", d => "y y" + d.year + " " + d.group)

      .attr("cx", d => projection([d.long, d.lat])[0])

      .attr("cy", d => projection([d.long, d.lat])[1])

      .attr("r", d => size(d.mag))

      .style("fill", d => color(d.group))

      .attr("stroke", d => color(d.group))

      .attr("stroke-width", 3)

      .attr("fill-opacity", .5)

      .on("mouseover", mouseover)

      .on("mousemove", mousemove)

      .on("mouseleave", mouseleave)

    svg.selectAll("." + "y").style("opacity", 0).style("visibility", "hidden")

    svg.selectAll("." + "y2000").transition().duration(1000).style("visibility", "visible").style("opacity", 1).attr("r", function (d) { return size(d.mag) })

    var y = 2000;

    var button = document.getElementsByClassName("btn");

    function step() {

      play(y);

      barplot(dataBar[y - 2000]);

      if (y == 2023) {

        y = 2000;

      } else {

        y++;

      }

    }

    function play(y) {

      svg.selectAll("." + "y").style("visibiliity", "hidden").transition().duration(1000).style("opacity", 0).attr("r", 0)

      var current = d3.select("#current").html(y);

      var text = document.getElementById("mapsvg").getAttribute("transform");

      var position = text.search("scale");

      var kScale = text[position + 6];

      if (text[position + 7] == '.') {

        kScale = kScale + text[position + 7] + text[position + 8] + text[position + 9]

      }

      console.log(y);

      svg.selectAll("." + "y" + y).attr("r", function (d) { return size(d.mag) / kScale }).attr("stroke-width", 3 / kScale).transition().duration(1000).style("visibility", "visible").style("opacity", 1)

    }

    d3.selectAll(".play").on("click", () => {

      if (d3.selectAll(".play").text() == "PLAY") {

        d3.selectAll(".play").text("PAUSE")

        y = parseInt(document.getElementById("current").textContent);

        if (y == 2023) {

          y = 2000;

        } else {

          y++;

        }

        play(y);

        barplot(dataBar[y - 2000]);

        if (y == 2023) {

          y = 2000;

        } else {

          y++;

        }

        for(i = 0; i <= 23; i++){

          button[i].disabled = true;

        }

        timer = setInterval(step, 2500);

      } else {

        d3.selectAll(".play").text("PLAY")

        for(i = 0; i <= 23; i++){

          button[i].disabled = false;

        }

        clearInterval(timer);

      }

    });

    d3.selectAll(".b2000").on("click", () => {

      svg.selectAll("." + "y").style("visibiliity", "hidden").transition().duration(1000).style("opacity", 0).attr("r", 0)

      var text = document.getElementById("mapsvg").getAttribute("transform");

      var position = text.search("scale");

      var kScale = text[position + 6];

      if (text[position + 7] == '.') {

        kScale = kScale + text[position + 7] + text[position + 8] + text[position + 9]

      }

      svg.selectAll("." + "y2000").transition().duration(1000).style("visibility", "visible").style("opacity", 1).attr("r", function (d) { return size(d.mag) / kScale }).attr("stroke-width", 3 / kScale)

  // radi skračivanja koda također i ovdje je naveden samo jedan primjer za postavljanje buttona, proces je identičan za ostale buttone.

 const margin = { top: 100, right: 30, bottom: 70, left: 60 },

    w = 560 - margin.left - margin.right,

    h = 500 - margin.top - margin.bottom;

  const svgBar = d3.select("#vizdata")

    .append("svg")

    .attr("width", w + margin.left + margin.right)

    .attr("height", h + margin.top + margin.bottom)

    .append("g")

    .attr("transform", `translate(${margin.left},${margin.top})`)

  const x = d3.scaleBand()

    .range([0, w])

    .domain(dataBar[20].map(d => d.group))

    .padding(0.2);

  svgBar.append("g")

    .attr("transform", `translate(0,${h})`)

    .call(d3.axisBottom(x))

  const y = d3.scaleLinear()

    .range([h, 0]);

  const yAxis = svgBar.append("g")

    .attr("class", "myYaxis")

  var tooltip = d3.select("#vizdata")

    .append("div")

    .style("position", "absolute")

    .style("visibility", "hidden")

    .style("background-color", "white")

    .style("border", "solid")

    .style("border-width", "1px")

    .style("border-radius", "5px")

    .style("padding", "10px")

    .style("z-index", "1")

  function barplot(data) {

    y.domain([0, d3.max(data, d => d.value)]);

    yAxis.transition().duration(1000).call(d3.axisLeft(y));

    var u = svgBar.selectAll("rect")

      .data(data)

    var text = d3.select("#current")

      .data(data)

      .html(d => d.year)

    if (data == dataBar[14]) {

      var text = d3.select("#current")

        .html("2014")

    }

    if (data == dataBar[9]) {

      var text = d3.select("#current")

        .html("2009")

    }

    const color = d3.scaleOrdinal()

      .domain(["Weak", "Light", "Moderate", "Strong"])

      .range(["#F7FABC", "#DDF272", "#F39C0D", "#DC4718"])

    u

      .join("rect")

      .style("fill", d => color(d.group))

      .transition()

      .duration(1000)

      .attr("x", d => x(d.group))

      .attr("y", d => y(d.value))

      .attr("width", x.bandwidth())

      .attr("height", d => h - y(d.value))

      .attr("id", d => d.group)

    setTimeout(function () {

      var year = document.getElementById("current").textContent;

      var Weak = svgBar.selectAll("#Weak")

        .attr("class", "deselected")

        .on("mouseover", function () {

          Weak.attr("opacity", .7);

          if (d3.selectAll(".selected").\_groups[0].length) {

            svg.selectAll("." + "y").filter(function () { return this.classList.contains('Weak') }).style("visibility", "visible");

          } else {

            svg.selectAll("." + "y").filter(function () { return !this.classList.contains('Weak') }).style("visibility", "hidden");

          }

          return tooltip.html("Broj potresa: " + data[0].value + "<br>Jacina magnitude: 2.5-3.9").style("visibility", "visible");

        })

        .on("mousemove", function () { return tooltip.style("top", (event.pageY - 100) + "px").style("left", (event.pageX + 50) + "px"); })

        .on("mouseout", function () {

          if (!(d3.selectAll("#Weak.selected").\_groups[0].length) && d3.selectAll(".selected").\_groups[0].length) {

            svg.selectAll("." + "y").filter(function () { return this.classList.contains('Weak') }).style("visibility", "hidden");

          }

          if (d3.selectAll(".selected").\_groups[0].length) {

            svg.selectAll("." + "y").filter(function () { return this.classList.contains('.selected') }).style("visibility", "visible");

          } else {

            svg.selectAll(".y" + year).style("visibility", "visible");

          }

          Weak.attr("opacity", 1);

          return tooltip.style("visibility", "hidden");

        })

        .on("click", function () {

          if (d3.selectAll("#Weak.deselected").\_groups[0].length) {

            Weak.attr("class", "selected");

            var selected = d3.selectAll("#Weak.selected");

            selected.attr("stroke", "black").attr("stroke-width", 3);

          } else {

            Weak.attr("class", "deselected");

            var selected = d3.selectAll("#Weak.deselected");

            selected.attr("stroke", "black").attr("stroke-width", 0);

          }

        });

      Weak.attr("stroke", "black").attr("stroke-width", 0);

//također i ovdje radi skraćivanja koda, analogno je i za ostale vrijednosti(Weak,Light,Moderate,Strong).

  }, 500);

  }

  barplot(dataBar[0])

</script>