

3. KV3 - Izrada prototipne vizualizacije podataka

3.1. Osnovne funkcionalnosti i ponašanja

Potrebno je napisati kod za izradu grafova za podatke o studentima po vrsti učilišta koja pohađaju, njihova rasporedba po županijama te po svakom učilištu zasebno.

Napraviti ćemo različite tipove grafova za svaku od tih kategorija koja najviše odgovara za prikaz određene vrste podataka.

3.2. Napredne funkcionalnosti i ponašanja:

Prelaskom po grafu korisniku se prikazuje detaljniji opis za te podatke.

Uvesti navbar koji će korisniku omogućiti prijelaz na različite prikaze podatke. Nakon čega korisnik ima mogućnost nekakvih dodatnih sortiranje ili filtriranja po podacima iz grafa.

Dodatno će biti potrebno omogućiti korisniku filtraciju podataka po svakom grafu ili sortiranje pritiskom iz drop-down tablice ili gumba.

3.3. Implementacija osnovnih funkcionalnosti

```
x.domain(data.map(function (d) { return d.academic_name; }));
y.domain([0, d3.max(data, function (d) { return d.number_of_students; })]);

g.append("g")
  .attr("class", "axis axis-x")
  .attr("transform", "translate(0," + height + ")")
  .call(d3.axisBottom(x).tickSizeOuter(0))
  .selectAll(".tick text")
  .attr("transform", "rotate(-45)")
  .style("text-anchor", "end")
  .text(d => d.substring(0, 25));

g.append("g")
  .attr("class", "axis axis-y")
  .call(d3.axisLeft(y).ticks(10))
  .append("text")
  .attr("transform", "rotate(-90)")
  .attr("y", 6)
  .attr("dy", "0.71em")
  .attr("text-anchor", "end")
  .text("Broj studenata");

g.selectAll(".bar")
  .data(data)
  .enter().append("rect")
  .attr("class", "bar")
  .attr("x", function (d) { return x(d.academic_name); })
  .attr("y", function (d) { return y(d.number_of_students); })
  .attr("width", x.bandwidth())
  .attr("height", function (d) { return height - y(d.number_of_students); })
  .attr("fill", function (d) { return color(d.academic_name); })
```

SLIKA 1: Prikaz stupčastog grafa studenata po učilištu koji su upisali

```
function createPieChart(data) {

  const svg = d3.select(".chart")
    .attr("width", width)
    .attr("height", height)
    .append("g")
    .attr("transform", `translate(${width / 2},${height / 2})`);

  const pie = d3.pie()
    .value(d => d.numberOfStudents)
    .sort(null);

  const arc = d3.arc()
    .innerRadius(0)
    .outerRadius(radius);

  const arcs = svg.selectAll("arc")
    .data(pie(data))
    .enter()
    .append("g")
    .attr("class", "arc")
    .on("mouseover", function (event, d) {
      if (d.data) {
        svg.append("text")
          .attr("class", "tooltip")
          .attr("transform", "rotate(-90)")
          .attr("x", 0)
          .attr("y", -radius * 1.2)
          .attr("text-anchor", "middle")
          .text(`${d.data.region}: ${d.data.numberOfStudents} studenata`);
      }
    })
  }
}
```

SLIKA 2: Prikaz tortnog grafa raspodjele studenata po županiji

```

const svg = d3.select(".chart")
  .attr("width", width + margin.left + margin.right)
  .attr("height", height + margin.top + margin.bottom)
  .append("g")
  .attr("transform", `translate(${margin.left},${margin.top})`);

const x = d3.scaleBand()
  .domain(counts.map(d => d.field))
  .range([0, width])
  .padding(0.1);

const y = d3.scaleLinear()
  .domain([0, d3.max(counts, d => d.count)])
  .nice()
  .range([height, 0]);

svg.append("g")
  .attr("class", "x axis")
  .attr("transform", `translate(0,${height})`)
  .call(d3.axisBottom(x))
  .selectAll("text")
  .attr("transform", "rotate(-45)")
  .attr("dx", "-.8em")
  .attr("dy", ".15em")
  .style("text-anchor", "end")
  .text(d => d.substring(0, 50));

svg.append("g")
  .attr("class", "y axis")
  .call(d3.axisLeft(y).ticks(10));

```

SLIKA 3. Prikaz stupčastog grafa studenata po vrsti učilišta

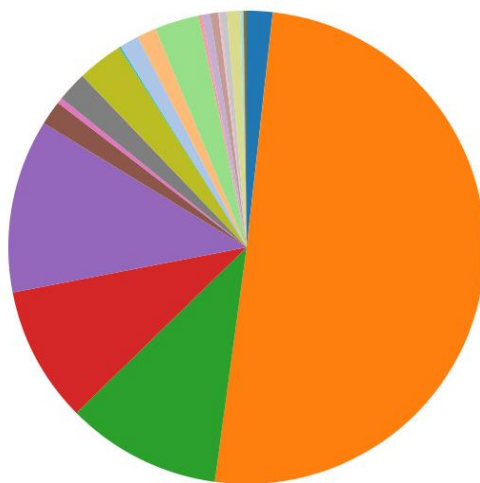
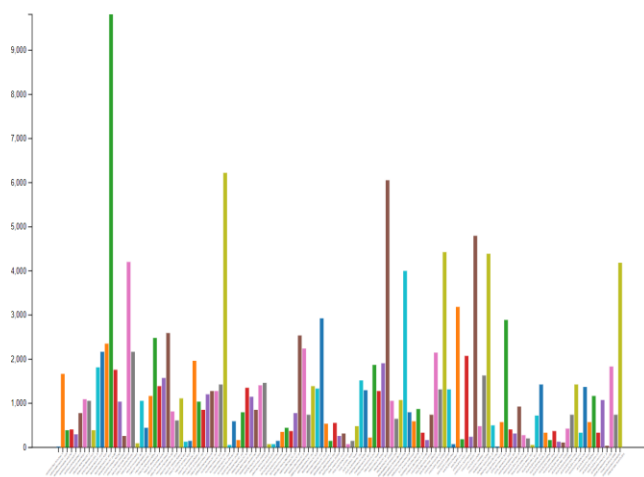
```

d3.json("cro.json", function(error, cro) {
  var data = topojson.feature(cro, cro.objects.layer1);
  var states = g.selectAll("path.county")
    .data(data.features)
    .enter()
    .append("path")
    .attr("class", "county")
    .attr("id", function(d) { return d.id; })
    .attr("d", path)
    .style("fill", function (d) {
      var value = Math.round(skala(d.properties.population));
      return colors[value]; })
    .style("stroke", "gray")
    .style("stroke-width", 1)
    .style("stroke-opacity", 1)
    .on("mouseover", function(d) {
      console.log(d.properties.population);
    })
    .on("click", onClick);
});

```

SLIKA 4: Prikaz koda za kartu Hrvatske

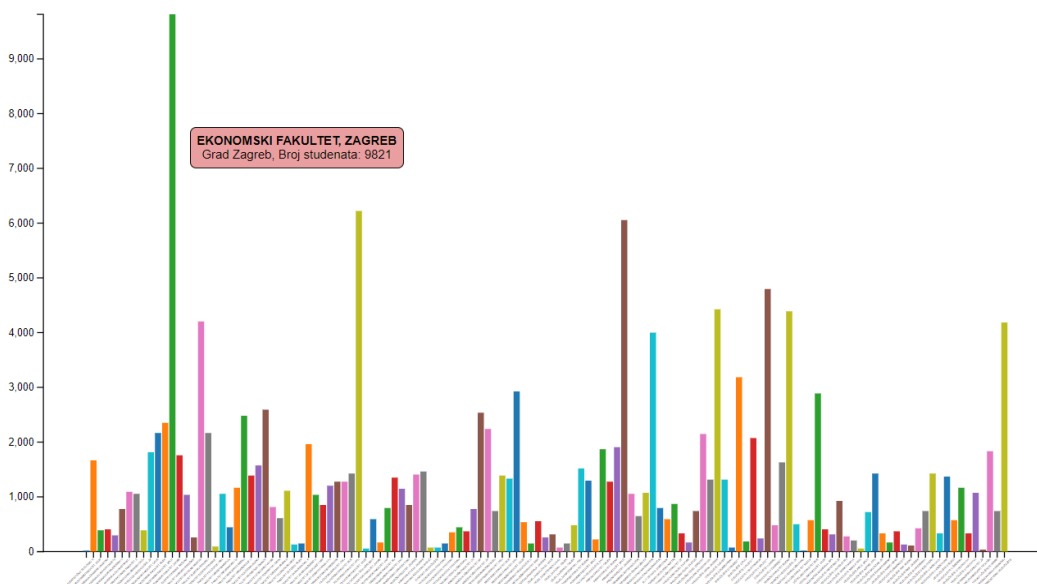
Slike 1,2,3,4. prikazuju osnovni kod korišten za prikaz osnovnih elemenata koji će se koristiti za prikaz podataka .



SLIKA 5: Prikaz rješenja osnovnih grafova

3.4. Implementacija osnovnog ponašanja

```
.on("mouseover", function (event, d) {  
  const tooltip = d3.select(".tooltip");  
  tooltip.style("display", "block");  
  tooltip.html(`<strong>${d.academic_name}</strong><br>${d.region}, Broj studenata: ${d.number_of_students}`)  
  .style("left", (event.pageX-185) + "px")  
  .style("top", (event.pageY) + "px");  
})  
.on("mouseout", function () {  
  const tooltip = d3.select(".tooltip");  
  tooltip.style("display", "none");  
});
```



SLIKA 6: Prikaz grafa nakon dodavanja funkcionalnosti hover za prikaz dodatnih podataka prelaskom po grafu

Slika 6. prikazuje prošireni osnovni graf koji prelaskom miša po grafu(stupcima grafa) pokazuje detalje tih podataka koji su jasnije vidljivi te prikazuju i županiju u kojoj se učilište nalazi.

```

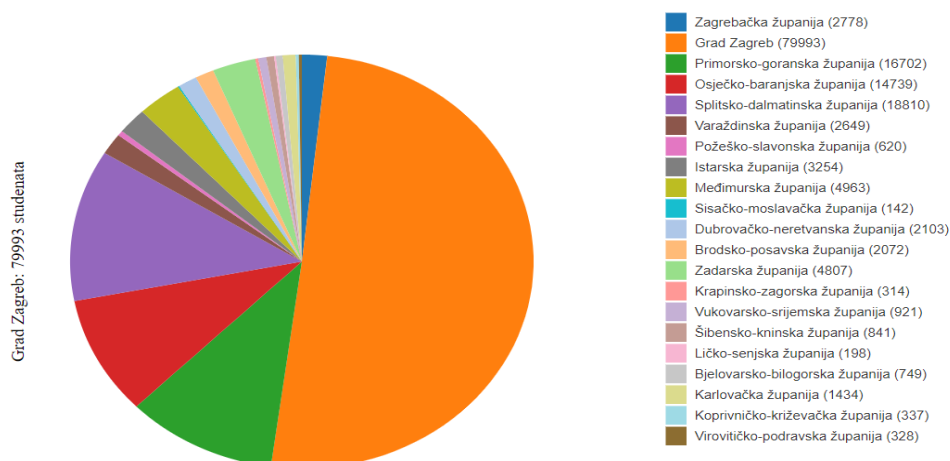
.on("mouseover", function (event, d) {
  if (d.data) {
    svg.append("text")
      .attr("class", "tooltip")
      .attr("transform", "rotate(-90)")
      .attr("x", 0) |
      .attr("y", -radius * 1.2)
      .attr("text-anchor", "middle")
      .text(`${d.data.region}: ${d.data.numberOfStudents} studenata`);
  }
})
.on("mouseout", function () {
  svg.select(".tooltip").remove();
});

const legend = d3.select("body").append("svg")
  .attr("class", "legend")
  .attr("width", 300)
  .attr("height", 450)
  .selectAll("g")
  .data(data)
  .enter().append("g")
  .attr("transform", function (_, i) {
    return `translate(10,${i * 20 + 10})`;
  });

legend.append("rect")
  .attr("x", 0)
  .attr("width", 18)
  .attr("height", 18)
  .style("fill", d => getColor(d.region));

legend.append("text")
  .attr("x", 25)
  .attr("y", 9)
  .attr("dy", ".35em")
  .style("text-anchor", "start")
  .text(function (d) {
    return d.region + " (" + d.numberOfStudents + ")";
  });
}

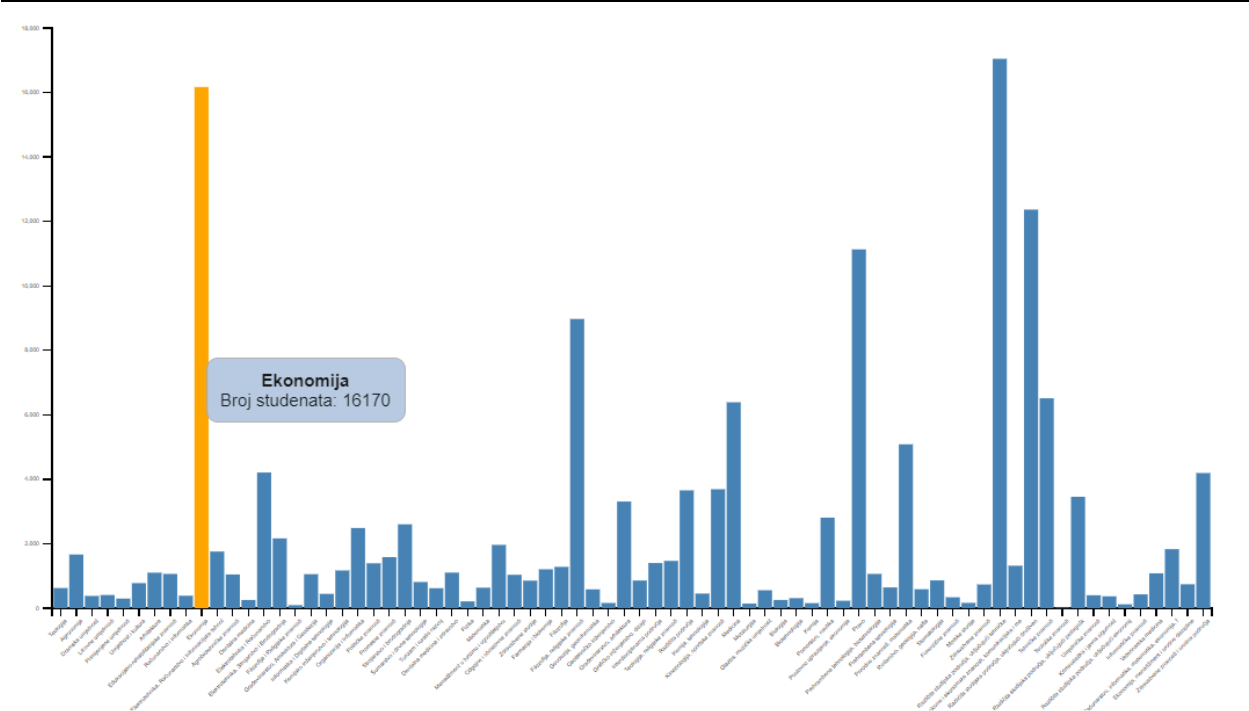
```



SLIKA 7: Prikaz tortnog grafa nakon ubacivanja legende i hover opcije prilikom prelaska po grafu

Slika 7. prikazuje tortni graf sa legendom za lakši pregled podataka sa grafa te prilikom prelaska po grafu s lijeve strane ispis županije i broja studenata za taj dio.

```
.on("mouseover", function (event, d) {
  const tooltip = d3.select(".tooltip");
  tooltip.transition()
    .duration(200)
    .style("opacity", .9);
  tooltip.html(`<strong>${d.field}</strong><br>Broj studenata: ${d.count}`)
    .style("left", (event.pageX) + "px")
    .style("top", (event.pageY - 28) + "px");
})
.on("mouseout", function (d) {
  const tooltip = d3.select(".tooltip");
  tooltip.transition()
    .duration(500)
    .style("opacity", 0);
});
```



SLIKA 8: Prikaz grafa za studente po vrsti učilišta

Slika 8. prikazuje graf sa studentima raspoređenim po vrsti učilišta koji su upisali te prelaskom po grafu(stupcima grafa) prikazuju se naziv i broj studenata za taj stupac koji prikazuju što omogućuje lakše čitanje sa grafa te i sami prikaz.

```

function onZoom() {
  projection.translate(d3.event.translate).scale(d3.event.scale);
  g.selectAll("path").attr("d", path);
}

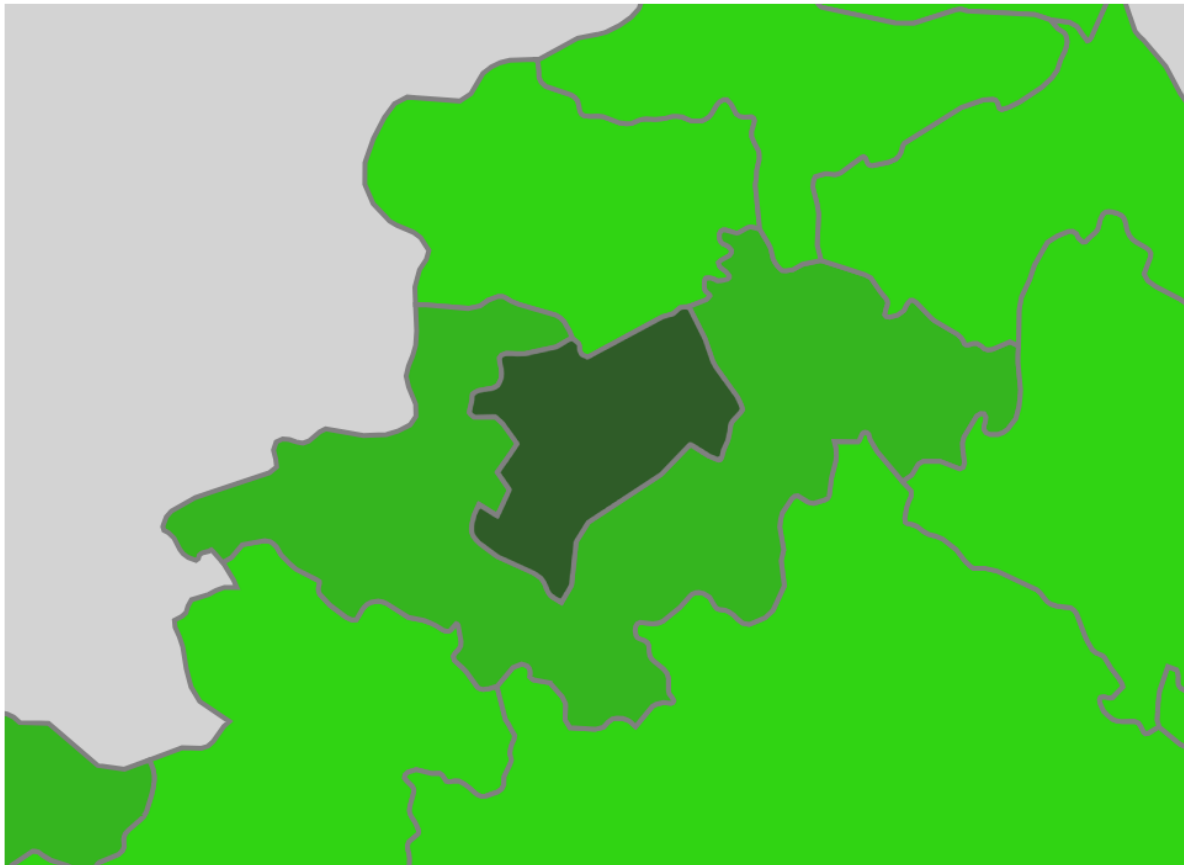
function onClick(d) {
  var a, b, c;

  var centroid = path.centroid(d);
  a = centroid[0];
  b = centroid[1];
  c = 4;
  centered = d;

  g.selectAll("path")
    | .classed("active", centered && function(d) { return d === centered; });

  d3.select("#info").html(`Županija: ${d.properties.name} <br/> Broj stanovnika: ${d.properties.population} <br/> Površina: ${d
  g.transition()
    | .duration(1000)
    | .attr("transform", `translate(${width / 2}, ${height / 2}) scale(${c}) translate(${a}, ${b})`)
    | .style("stroke-width", 1.5 / c + "px");

```



Županija: Grad Zagreb
 Broj stanovnika: 790017
 Površina: 641 km²

SLIKA 9: Prikaz karte hrvatske sa dodatnim funkcionalnostima

Slika 9. prikazuje proširenje osnovnog prikaza karte hrvatske da prilikom pritiska na određenu županije uveća prikaz na nju i ispiše podatke o površini, imenu županije i broju stanovnika.