# KV4 - Izrada prototipne vizualizacije podataka

## Osnovne funkcionalnosti i ponašanja

* + 1. Prikaz USA karte is json podataka
    2. Klikom na državu ispisuju se podaci za određenu godinu

## Napredne funkcionalnosti i ponašanja:

* + 1. Klikom na državu zumira se da bude u centru SVG-a
    2. Odabirom određene godine, mjenjaju se podaci relevantni za tu godinu

## Implementacija osnovnih funkcionalnosti

A map of the united states

Description automatically generated

//first view

createMap(unemployment\_data.unemployment, map\_data, yearExtent[0]);

d3.select('#detail')

.append('svg')

.attr('id', 'details')

.attr('width', 480)

.attr('height', 210)

.attr('transform', function (d) {

return 'translate(' + 805 + ',' + -606 + ')';

});

});

function createMap(unemployment, mapData, year) {

d3.select('#map svg').remove();

//getting data from selected year

unemployment = unemployment.filter((d) => d.Year == year);

var width = 800,

var height = 600;

var svg = d3

.select('#map')

.append('svg')

.attr('width', width)

.attr('height', height);

var projection = d3.geoAlbersUsa().translate([width / 2, height / 2]);

var path = d3.geoPath().projection(projection);

var originalScale = projection.scale();

var originalTranslate = projection.translate();

var states = topojson.feature(mapData, mapData.objects.states).features;

// color

var colorScale = d3

.scaleLinear()

.domain(

d3.extent(

unemployment,

(d) => +d['Percent (%) of Labor Force Unemployed in State/Area']

)

)

.range([d3.rgb('#ffc9bb'), d3.rgb('#c61a09')]);

// Combine data from map with unemploymenty

states.forEach(function (state) {

var unemploymentData = unemployment.find(

(d) =>

d['State/Area'].toLowerCase() ===

state.properties.name.toLowerCase()

);

state.properties.unemployment = unemploymentData

? +unemploymentData[

'Percent (%) of Labor Force Unemployed in State/Area'

]

: 0;

state.properties.total = unemploymentData

? +unemploymentData['Total Unemployment in State/Area']

: 0;

});

svg

.append('g')

.selectAll('path')

.data(states)

.join('path')

.attr('d', path)

.attr('stroke', 'white')

.attr('fill', (d) => colorScale(d.properties.unemployment))

.on('mouseover', function (event, d) {

d3.select(this)

.style('cursor', 'pointer')

.style('stroke', 'black')

.attr('stroke-width', '0.5px');

})

.on('mouseout', function (event, d) {

d3.select(this)

.style('stroke', 'white')

.attr('stroke-width', '0.5px');

})

.on('click', function (event, d) {

//get data for clicked state

var unemploymentData = unemployment.find(

(e) =>

e['State/Area'].toLowerCase() === d.properties.name.toLowerCase()

);

}

## Implementacija osnovnog ponašanja

//zoomout funcionality

if (zoomedState === d) {

zoomedState = null;

svg

.transition()

.duration(750)

.call(zoom.transform, d3.zoomIdentity);

} else {

var bounds = path.bounds(d),

dx = bounds[1][0] - bounds[0][0],

dy = bounds[1][1] - bounds[0][1],

x = (bounds[0][0] + bounds[1][0]) / 2,

y = (bounds[0][1] + bounds[1][1]) / 2,

scale = Math.max(

1,

Math.min(8, 0.9 / Math.max(dx / width, dy / height))

),

translate = [width / 2 - scale \* x, height / 2 - scale \* y];

// new state

svg

.transition()

.duration(750)

.call(

zoom.transform,

d3.zoomIdentity

.translate(translate[0], translate[1])

.scale(scale)

);

zoomedState = d;

}

});

// zoom

var zoom = d3.zoom().scaleExtent([1, 8]).on('zoom', zoomed);

svg.call(zoom);

function zoomed({ transform }) {

svg.selectAll('path').attr('transform', transform);

}

A map of the united states

Description automatically generated with medium confidence

A black background with white text

Description automatically generated with low confidence