# KV2 - Dizajn vizualizacije podataka.

## Pitanja na koja vizualizacija daje odgovor

[Navesti i opisati na koja pitanja će vizualizacija pružati odgovor.

*Precizirati pitanja na koja se odgovara vizualizacijom podataka. Potrebno je osigurati da su pitanja jasno formulirana i da se mogu odgovoriti na temelju dostupnih podataka.*]

* + 1. Pitanja na koja će vizualizacija dati odgovore:
* Koliko je igara Steam izdao u posljednjih 12 godina ili u određenoj godini po mjesecima
* Kolika je prosječna cijena Steam igara u posljednjih 12 godina ili u određenoj godini po mjesecima
* Kolika je podržanost igara ovisno o operacijskome sustavu u posljednjih 12 godina ili u određenoj godini po mjesecima
* Kakvo je zadovoljstvo ljudi za izdane Steam igre u posljednjih 12 godina ili u određenoj godini po mjesecima

## Skica vizualizacije podataka

[Prikazati skice različitih načina prikaza podataka, uz objašnjenje njihove svrhe]

* + 1. Skice grafova koji će se koristiti:

Slika na kojoj se prikazuje tekst, snimka zaslona, crta, dijagram

Opis je automatski generiran

Odgovara na pitanje:

Koliko je igara Steam izdao u posljednjih 12 godina ili u određenoj godini po mjesecima?

Slika na kojoj se prikazuje tekst, snimka zaslona, crta, broj

Opis je automatski generiran

Odgovara na pitanje:

Kakvo je zadovoljstvo ljudi za izdane Steam igre u posljednjih 12 godina ili u određenoj godini po mjesecima?

Slika na kojoj se prikazuje snimka zaslona, tekst, dijagram, radnja

Opis je automatski generiran

Odgovara na pitanje:

Kolika je prosječna cijena Steam igara u posljednjih 12 godina ili u određenoj godini po mjesecima?

Slika na kojoj se prikazuje snimka zaslona, krug, dijagram, grafika

Opis je automatski generiran

Odgovara na pitanje:

Kolika je podržanost igara ovisno o operacijskome sustavu u posljednjih 12 godina ili u određenoj godini po mjesecima?

## Postojeća rješenja i primjeri

[Navesti primjere sličnih projekata ili kodova koji će biti korisni za izradu ovog projekta s pripadajućim poveznicama i pojašnjenjem koji elementi/dijelovi se planiraju upotrijebiti]

* + 1. Stranice koje će se koristiti za izradu vizualizacije su:
* Git repozitorij sa laboratorijskih vježbi: <https://github.com/realGmto/DataVisualization>
* Stranica sa zbirkom različitih vizualizacija: <https://www.geckoboard.com>
* D3 dokumentacije: <https://devdocs.io/d3~7/> , <https://d3js.org>
  + 1. Kodovi koji će se koristi za izradu vizualizacija mogu se pronaći u git repozitoriju sa laboratorijskih vježbi. Koristiti će se ovi kodovi:

|  |
| --- |
| var margin = {top: 20, bottom: 70, left:120, right: 20};  var width = 1300 - margin.left - margin.right;  var height = 800 - margin.top - margin.bottom;  var barPadding = 5;  var barWidth = width / total.length - barPadding;  var color = d3.scale.ordinal()  .range(["red", "#ffa700", "#fff400", "#a3ff00", "green", "#00BA2D", "#009658", "#007185", "#0053A8", "blue"]);  var xScale =d3.scale.ordinal()  .domain(year)  .rangeRoundBands([0,width]);  var yScale = d3.scale.linear()  .domain([0,d3.max(total)])  .range([height,0]);  var svg = d3.select("body")  .append("svg")  .attr("height",height + margin.bottom + margin.top)  .attr("width",width + margin.left + margin.right +barWidth)  .append("g")  .attr("transform", "translate(" + margin.left + "," + margin.top +")");    var xAxis = d3.svg.axis()  .scale(xScale)  .orient("bottom")  .tickFormat(function(d, i) { return i + 2001; });  var yAxis = d3.svg.axis()  .scale(yScale)  .orient("left")  .ticks(10);  svg.append("g")  .attr("class", "x axis")  .attr("transform", "translate(0," + height + ")")  .call(xAxis)  .selectAll("text")  .style("text-anchor", "middle");  svg.append("text")  .attr("x", (width / 2))  .attr("y", (height + (margin.bottom / 2)))  .attr("dy", "1em")  .style("text-anchor", "middle")  .text("Godina");  svg.append("g")  .attr("class", "y axis")  .call(yAxis)  svg.append("text")  .attr("transform", "rotate(-90)")  .attr("x",0 - (height / 2))  .attr("y", 0 - margin.left)  .attr("dy", "1em")  .style("text-anchor", "middle")  .text("Ukupna sredstva (EUR)");  var barchart = svg.selectAll("rect")  .data(total)  .enter()  .append("rect")  .attr("x", function(d, i) { return xScale(year[i]); })  .attr("y", function(d) { return yScale(d); })  .attr("width", barWidth)  .attr("height", function(d) { return height - yScale(d); })  .attr("fill", function(d,i) {return color(i)}); |

Kod 1: Prikaz koda za izradu stupčastoga grafa

|  |
| --- |
| var margin = {top: 20, bottom: 70, left:120, right: 20};  var width = 1300 - margin.left - margin.right;  var height = 800 - margin.top - margin.bottom;  var xScale =d3.scaleTime()  .domain([startDate, endDate])  .range([0,width]);  var yScale = d3.scaleLinear()  .domain([0,d3.max(dataset,function(d) {return d.value})])  .range([height,0]);  var svg = d3.select("body")  .append("svg")  .attr("height",height + margin.bottom + margin.top)  .attr("width",width + margin.left + margin.right)  .append("g")  .attr("transform", "translate(" + margin.left + "," + margin.top +")");    svg.append("g")  .attr("class", "x-axis")  .attr("transform", "translate(0," + height + ")")  .call(d3.axisBottom(xScale));  svg.append("g")  .attr("class", "y-axis")  .call(d3.axisLeft(yScale));  svg.append("text")  .attr("x", (width / 2))  .attr("y", (height + (margin.bottom / 2)))  .attr("dy", "1em")  .style("text-anchor", "middle")  .text("Vrijeme");  svg.append("text")  .attr("transform", "rotate(-90)")  .attr("x",0 - (height / 2))  .attr("y", 0 - margin.left)  .attr("dy", "1em")  .style("text-anchor", "middle")  .text("Vrijednost");    var line = d3.line()  .x(function(d) { return xScale(d.date); })  .y(function(d) { return yScale(d.value); });    svg.append("path")  .datum(dataset)  .attr("class", "line")  .attr("d", line)  .style("stroke","blue")  .style("fill", "none"); |

Kod 2: Prikaz koda za izradu linijskoga grafa

|  |
| --- |
| var height = 500;  var width = 500;  var outerRadius = 200;  var innerRadius = 0;  var color = d3.scaleOrdinal(d3.schemeCategory10);  var data = [  {name: 'jabuka', value: 17},  {name: 'kruska', value: 23},  {name: 'banana', value: 83},  {name: 'jagoda', value: 47},  {name: 'ribizla', value: 15}  ];  var arc = d3.arc()  .innerRadius(innerRadius)  .outerRadius(outerRadius);  var pie = d3.pie()  .value(function(d) { return d.value; });  var svg = d3.select("body").append("svg")  .attr("width", width)  .attr("height", height);    var pieArcs = svg.selectAll("g.pie")  .data(pie(data))  .enter()  .append("g")  .attr("class", "pie")  .attr("transform", "translate(" + (width / 2) + ", " + (height / 2) + ")");  /\*pieArcs.append("path")  .attr("fill", function(d, i) { return color(i); })  .attr("d", arc);    pieArcs.append("text")  .attr("transform", function(d) { return "translate(" + arc.centroid(d) + ")"; })  .attr("text-anchor", "middle")  .text(function(d) { return d.value + " - " + d.data.name; });\*/  pieArcs.append("path")  .attr("fill", function(d, i) { return color(i); })  .attr("id", function(d,i) { return "Arc\_"+i; })  .attr("d", arc);    pieArcs.append("text")  .attr("dy",-30)  .attr("x",function(d){return d.value\*2.5;})  .append("textPath")  .attr("xlink:href",function(d,i){return "#Arc\_"+i;})  .text(function(d){return d.data.name;}); |

Kod 3: Prikaz koda za izradu kružnoga grafa

* + 1. Koristiti će se (kod 1.) za izradu stupčastoga grafa, (kod 2.) za izradu linijskoga grafa i (kod 3.) za izradu kružnoga grafa. Isto tako će se iz repozitorija koristiti primjeri tranzicija (kod 4.).

|  |
| --- |
| var svg = d3.select("body")  .append("svg")  .attr("height",500)  .attr("width",500);    var rect = svg.append("rect")  .attr("x",100)  .attr("y",100)  .attr("height",100)  .attr("width",100)  .style("fill","white")  .style("stroke","black");  rect.transition()  .delay(3000)  .duration(5000)  .ease(d3.easeLinear)  .style("fill","black")  .transition()  .attr("width",500/2)  .style("opacity",0); |

Kod 4: Prikaz koda za dodavanje tranzicija

## Prilagodba podataka

[Opisati potrebne prilagodbe podataka te ih prilagodili odabranom načinu prikaza]

* + 1. Postupak pripreme podataka je bio ovakav: Prvo su podaci preuzeti sa kaggle stranice u .csv formatu. Sljedeće je bilo rastaviti „date\_release“ stupac u tri nova stupca „['year', 'month', 'day']“. Nakon toga su se iz skupa podataka izbacili stupci „['steam\_deck','discount','price\_original','date\_release','day']“. Kada se sve to odradilo onda su se mapirala imena mjeseci umjesto broja. Zadnje što se sa podacima odradilo je izbacivanje svih onih redova čiji je stupac „'year'“ bio manji od 2011.
    2. Podaci su došli u .csv formatu ali zbog jednostavnosti rada u Javascript-u prebačeni su u .json format.
    3. Nakon uređivanja i prilagodbi dobivaju se ovakvi podaci:

Slika na kojoj se prikazuje tekst, snimka zaslona, broj, Font

Opis je automatski generiran

Slika 1: Prikaz dijela podataka nakon pripreme

* + 1. Iz (Slika 1.) se može vidjeti da su podaci uspješno prilagođeni a i dokaz tomu je (Slika 2.) graf koji je izrađen pomoću tih podataka.

Slika na kojoj se prikazuje tekst, snimka zaslona, dijagram, Font

Opis je automatski generiran

Slika 2: Prikaz izrađenoga grafa korištenjem ranije navedenim podacima

## Boje i podatci

* + 1. Popis boja sa njihovim obrazloženjima:
* Crevno / Zelena – Koriste se u kružnom grafu za prikaz podržavanosti Steam igara na određenome operacijskome sustavu. Koristi se ovaj kontrast jer je odmah razlučivo koji su pozitivni a koji negativni podaci.
* Različite nijanse tamno plave – Koriste se kao pozadina stranice te pružaju moderni estetski prikaz.
* Svijetlo plava – Koristi se za crtanje stupaca i linija kod stupčastih i linijskih grafova. Ova boja se koristi zato što se ističe od pozadine ali i dalje ulazi u estetiku cjelokupnoga dojma
* Bijela – Koristi se za tekst i razne informacije zbog svoje vidljivosti na tamnoj pozadini