

**INTEGRACIJA INTERAKTIVNIH MAPA UPOTREBOM JAVASCRIPT BIBLIOTEKA  
INTEGRATION OF INTERACTIVE MAPS USING JAVASCRIPT LIBRARIES**Nemanja Jovančević, *Fakultet tehničkih nauka, Novi Sad***Oblast – INŽENJERSTVO INFORMACIONIH  
SISTEMA**

**Kratak sadržaj** – U ovom radu dat je pregled aktuelnih interaktivnih mapa koje su besplatno dostupne na internetu. Izvršena je analiza mogućnosti integracije interaktivnih mapa. Dat je opis integracije interaktivnih mapa u Web aplikaciju upotrebom JavaScript biblioteka. Web aplikacija je razvijena u HTML programskom jeziku.

**Ključne reči:** prostorni podaci, prostorne baze podataka, rasterski slojevi, vektorski slojevi, OpenLayers, JavaScript, HTML, CSS, GeoServer

**Abstract** – This paper gives an overview of current interactive maps that are freely available on the Internet. An analysis of the possibility of integrating interactive maps was performed. A description of the integration of interactive maps into the Web application using JavaScript libraries is given. Web application is developed in HTML programming language.

**Keywords:** spatial data, spatial database, raster layers, vector layers, OpenLayers, JavaScript, HTML, CSS, GeoServer

**1. UVOD**

Napredak informacionih tehnologija u oblasti grafičke prezentacije podataka kao jednu od posledica imao je i pojavu popularnosti Web mapa. U ranoj fazi razvoja pomenutih tehnologija kreiranje interaktivnih Web mapa bilo je rezervisano samo za velike kompanije i eksperte koji su imali na raspolaganju značajna finansijska sredstva.

Danas, pojavom besplatnih servisa, kreiranje Web mapa je postalo lako dostupno širem krugu zainteresovanih strana. Upotrebom odgovarajućih alata, moguće je kreirati Web mapu, uz odgovarajuća znanja iz oblasti geografije, kartografije i programiranja. Da bi se bolje razumele Web mape potrebno je objasniti osnovne pojmove koji se javljaju kod klasičnih štampanih mapa.

Prednosti Web Mapa su mnoge: besplatne su, može im se pristupiti sa bilo kog mesta, može im se pristupiti u bilo koje vreme, mogu se implementirati i integrisati u kreirane Web aplikacije.

Neke od poznatih Web mapa su: Google, Yandex, OpenStreetMap i Bing. Za sve navedene mape razvijene su biblioteke koje omogućavaju integraciju mapa u Web aplikacije.

**NAPOMENA:**

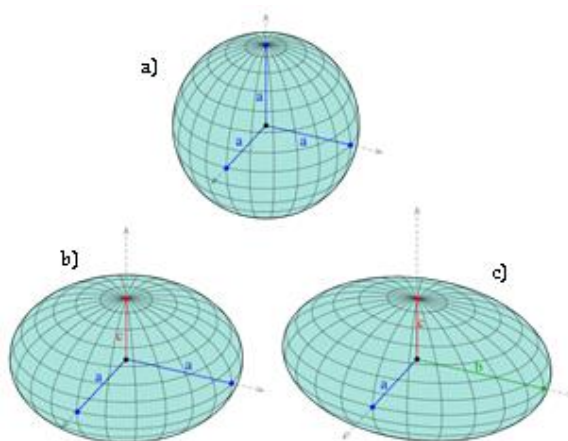
**Ovaj rad proistekao je iz master rada čiji mentor je bio dr Đorđe Pržulj, vanr. prof.**

Kod velikog broja aplikacija postoji potreba da se određeni prostorni podaci sačuvaju i prikažu na mapi. Potreba za čuvanjem prostornih podataka i prikaz istih je sve veća. Podaci se čuvaju u posebnim specijalizovanim prostornim bazama podataka. Podacima se upravlja upotrebom specifičnog SQL jezika.

**2. UPRAVLJANJE PROSTORNIM PODACIMA**

U ovom poglavlju objašnjeni su osnovni pojmovi koji se javljaju kod klasičnih štampanih mapa. Pored pomenutih pojmova biće objašnjene Merkatorova i Web Merkatorova projekcija. Biće i još objašnjenih prostorni podaci, zapisi prostornih podataka i prostorne baze podataka.

Radi boljeg razumevanja Web mapa potrebno je objasniti osnovne pojmove koji se pojavljuju kod klasičnih štampanih mapa. Mapa predstavlja matematički način prikazivanja Zemljine površine na ravan. Nauka koja se bavi izradom mapa naziva se kartografija [1]. U procesu izrade mape potrebno je definisati oblik Zemlje. Oblik Zemlje može da se definiše kao: sfera (lopta), sferoid ili elipsoid (slika 1).



Slika 1. Vrste geometrijskih tela: a) sfera, b) sferoid, c) elipsoid [2]

Proces izrade mape zahteva pored definisanog oblika, definisanje načina prenošenja površine Zemlje na ravan i definisanje karaktera deformacije

Prema načinu prenošenja površine Zemlje na ravan, mape se dele na [3]: cilindrične, kupaste i azimutne. Prema karakteru deformacija, mape se dele na: konformne, ekvivalentne i ekvidistantne.

**2.1. Merkatorova projekcija**

Merkatorova projekcija predstavlja cilindričnu projekciju Zemlje koju je predstavio Gerhard Merkator 1569. godine. Merkatorova projekcija postala je važna za razvoj

ostalnih projekcija. Jedna je od najpoznatijih projekcija i ima dugu istoriju upotrebe u izradi mapa na globalnom nivou[4]. Merkatorovom projekcijom očuvani su veličina i oblik malih objekata na mapi, veličina i oblik velikih objekata se iskrivljuje. Merkatorova projekcija ne prikazuje tačnu veličinu kontinenata koji su udaljeniji od ekvatora.

## 2.2. Web Merkatorova projekcija

Web Merkatorova projekcija razvijena je za potrebe projekcije interaktivnih Web mapa. Bazira se na Merkatorovoj projekciji. Nastala je početkom 21. veka, paralelno sa počecima razvoja interaktivnih Web mapa. I ako ima dosta sličnosti, postoji i dosta razlika između Merkatorove i Web Merkatorove projekcije. Web Merkatorova projekcija predstavlja poseban slučaj Merkatorove projekcije na sferu sa koordinatama elipsoida. Web Merkator je poznat i kao Sferni Merkator [4].

## 2.3. Prostorni podaci

Prostorni podaci predstavljaju podatke koji imaju direktnu ili indirektnu vezu sa specifičnom lokacijom ili geografskim područjem. Set prostornih podataka predstavlja kolekciju prostornih podataka koji su međusobno povezani. Prostornim podacima dobija se odgovor na pitanje „Gde?“. Oni predstavljaju položaj određene lokacije u realnom svetu.

Postoje dva načina prikupljanja prostornih podataka [5]:

1. primarno prikupljanje prostornih podataka i
2. sekundarno prikupljanje prostornih podataka.

## 2.3. Modeli prostornih podataka

Da bi se vizuelno predstavile prirodne pojave, potrebno je definisati najbolji model za predstavljanje geografskog prostora. Modeli prostornih podataka predstavljaju pravila koja se koriste za opis i predstavljanje aspekata stvarnog sveta u računaru. Postoje dva primarna modela prostornih podataka [6]:

1. rasterski model podataka i
2. vektorski model podataka.

Rasterski model podataka podatke predstavlja kroz mrežu ćelija u obliku kvadrata. Svaka ćelija ima sopstvene koordinate kojima se definiše njen položaj u mreži ćelija. Ćelije se još nazivaju i pikseli (Picture Element). Za svaki pojedinačni piksel definiše se boja. Za piksel je moguće vezati određeni atribut (nadmorska visina, temperatura i sl.). Rasterski model karakteriše i rezolucija (broj piksela). Rasterski model se najčešće prikazuje kao slika. Kvalitet slike zavisi od rezolucije, tako da veća rezolucija daje kvalitetniju sliku. Kvalitet slike ne zavisi samo od rezolucije već i od dubine boje. Dubina boje predstavlja vrednost koja je dodeljena svakom pojedinačnom pikselu, ukoliko je dubina boje veća bolji je kvalitet slike i obratno. Vektorski model podataka je suprotan od rasterskom modelu podataka. Vektorskim modelom podaci se predstavljaju pomoću geometrijskih oblika: tačke, linije i poligona. Vektorski model podataka može biti strukturiran na više načina. Neki od načina su: špageti model podataka i topološki model podataka.

U špageti modelu svaki poligon, tačka ili linija moraju biti jedinstveno definisani sopstvenim skupom X i Y koordinata, čak i ako poligoni ili linije dele iste granične

informacije. Model dovodi do pojave redundantosti podataka i time se smanjuje efikasnost ovog modela. Kod špageti modela ne postoje topološki odnosi između objekata [6].

Topološki model je suprotan od špageti modela. Podaci su povezani i obogaćeni informacijama koje se odnose na prostorne odnose. Bitna karakteristika topološkog modela podataka jeste postojanje topoloških odnosa između tačke, linije i poligona. Neki od topoloških odnosa su: preklapanje i dodirivanje. Topološki model važan je kod postavljanja prostornih upita [6].

## 2.3.2. Prostorna baza podataka

Prostorne baze podataka predstavljaju baze podataka koja omogućava upravljanje prostornim podacima. U prostornoj bazi podataka mogu se koristiti primitivni tipovi podataka kao što su integer, boolean i varchar. Pored primitivnih mogu se koristiti i prostorni tipovi podataka kao što su tačka, linija, poligon. Podaci koji se smeštaju u prostornu bazu podataka dele se na: atributivne podatke i prostorne podatke. Atributivnim podacima se opisuje geografski prostor koji se smešta u bazu podataka. Prostornim podacima se definiše položaj objekta u geografskom prostoru.

Indeksiranje prostornih baza podataka je specifično. Za indeksiranje koriste se specifična stabla pristupa. Neka od poznatih stabla pristupa su: R-tree (R-stablo) i Quad tree (Kvadratno stablo).

## 3. Korišćene tehnologije i alati

U ovom poglavlju opisane su tehnologije i alati upotrebljeni u procesu implementacije i razvoja HTML stranica, ali i tehnologije i alati upotrebljeni u procesu integracije interaktivnih mapa u HTML stranice.

Za potrebe istraživanja kreirana je Web aplikacija u koju je integrisana interaktivna mapa. Za proces integracije interaktivnih mapa korištene su sledeće biblioteke: Google Maps API, Yandex Maps API i OpenLayers za integraciju OpenStreetMap mapu.. Prostorni podaci koji su prikazani u aplikaciji čuvaju se u prostornoj bazi podataka PostGIS. Aplikaciji se pristupa preko Web pretraživača putem internet adrese aplikacije. Za potrebe editovanja i prikazivanja prostornih podataka smeštenih u PostGIS bazi podataka korišten je GeoServer. PostGIS, GeoServer i Apache instalirani su na lokalnom računaru. Aplikacija je kreirana pomoću HTML-a, CSS-a i JavaScript-a.

## 4. Implementacija vektorskih slojeva u GeoServeru

U ovom poglavlju opisani su prostorna baza podataka i proces kreiranja vektorskih slojeva GeoServer-u. Proces kreiranja obuhvata: kreiranje store-a, kreiranje slojeva, kreiranje grupe slojeva i kreiranje stilova za slojeve.

Za potrebe istraživanja kreirana je prostorna baza podataka pod nazivom „srbija“ koja se nalazi na PostgreSQL serveru lokalnog računara. U bazi su kreirane četiri tabele: `granicni_prelaz_drumski` (`id_objekta`, `naziv`, `drzava`, `the_geog`), `hidroelektrane` (`id_objekta`, `naziv`, `broj_generatora`, `ukupna_snaga`, `the_geog`), `pruge` (`id_objekta`, `the_geog`), `zeleznicke_stanice` (`id_objekta`, `vrsta_stanice`, `mesto`, `the_geog`).

GeoServer je web server koji prihvata zahtev klijenta i kao odgovor na zahtev klijenta vraća podatke. Klijent

zatim dobijene podatke upotrebom OpenLayers biblioteke prezentuje korisniku. Da bi podaci bili prikazani primenom WMS standarda, potrebno je omogućiti GeoServer-u da komunicira sa PostGIS bazom podataka. U GeoServer-u potrebno je kreirati *store* (izvor podataka).

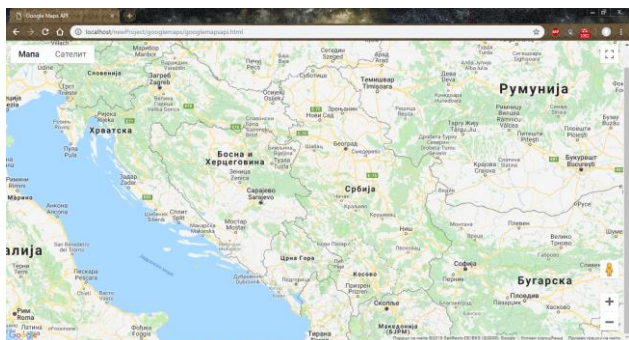
### 5. Integracija Google, Yandex i OpenStreetMap mapa upotrebom JavaScript API-a

U ovom poglavlju opisana je integracija OpenStreetMap mape upotrebom OpenLayers biblioteke, integracija Google mape upotrebom Google Maps API-a, integracija Yandex mape upotrebom Yandex Maps API-a i analiza izvršenih integracija.

Za potrebe istraživanja izvršena je analiza integracije Google mape upotrebom Google Maps API-a. Kompanija Google razvila je JavaScript API koji omogućava programeru da integriše Google mapu u sopstvenu aplikaciju.

Web stranica pisana je u HTML jeziku. Mapa je smeštena u `<div>` element. U procesu implementacije Web aplikacije potrebno je: definisati `<div>` element u kom će mapa biti smeštena, omogućiti pretraživaču učitavanje JavaScript API-a i kreirati funkciju kojom će se kreirati mapa i učitavati kreirana mapa u `<div>` element.

Google mapa bez pokazivača lokacije i putanje od tačke A do tačke B prikazana je na slici 2.



Slika 2. Google mapa bez pokazivača lokacije i putanje

Na listingu 1 može da se vidi JavaScript kod za kreiranje jednostavne mape prikazane na slici 2.

```

<script>
  var map;
  function initMap() {
    map = new
    google.maps.Map(document.getElementById('map'), {
      center: {lat: 44.0570766, lng:19.7119942 },
      zoom: 7
    });
  }
</script>
<script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyAMobXUIo4xA
4hXFWedfYc4MyLIguLUeM&callback=initMap"
async defer"></script>

```

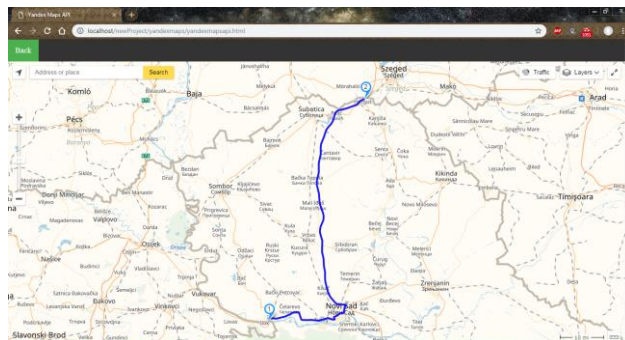
Listing 1. JavaScript kod za kreiranje jednostavne mape

Na mapu je moguće dodati pokazivač lokacije i putanju od tačke A do tačke B.

Yandex kao i Google pruža podršku integraciji mape sa Web aplikacijom pomoću JavaScript API-a. Integracija Yandex mape upotrebom JavaScript API-a je slična integraciji Google mape. Za integraciju Yandex mape potrebno je: definisati HTML element (`<div>` element) u kom će biti učitana mapa, omogućiti pretraživaču da učita

API i kreirati funkciju pomoću koje će se kreirati mapa i učitavati u prethodno definisan HTML element.

Na Yandex mapu kao i na Google mapu moguće je dodati pokazivač lokacije i putanju. Na slici 3 prikazana je Yandex mapa sa putanjom od tačke A do tačke B.



Slika 3. Yandex mapa sa putanjom od tačke A do tačke B

Na listingu 2 prikazan je JavaScript kod za kreiranje putanje od tačke A do tačke B i prikazivanje putanje na mapu.

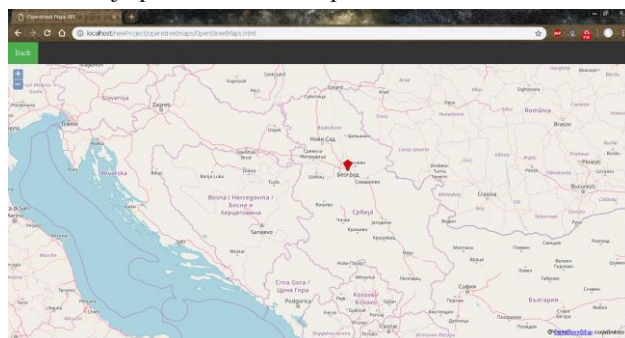
```

<script type="text/javascript">
  ymaps.ready(init);
  var myMap
  function init(){
    myMap = new ymaps.Map("map", {
      center: [44.0570766, 19.7119942],
      zoom: 7
    });
    ymaps.route([
      { type: 'wayPoint',
        point: [45.239902, 19.398539] },
      { type: 'wayPoint',
        point: [46.173530, 19.976235] }
    ], {
      mapStateAutoApply: true
    }).then(function (route) {
      route.getPaths().options.set({
        balloonContentLayout:
        ymaps.templateLayoutFactory.createClass('{{
        properties.humanJamsTime }}'),
        strokeColor: '0000ffff',
        opacity: 0.9
      });
      myMap.geoObjects.add(route);
    });
  }
</script>

```

Listing 2. JavaScript kod za kreiranje putanje i dodavanje putanje na mapu

Pored Yandex i Google mape, izvršena je integracija i OpenStreetMap mape upotrebom OpenLayers biblioteke. OpenLayers biblioteka omogućava korisniku da doda pokazivač lokacije. Na slici 4 prikazana je OpenStreetMap mapa sa pokazivačem lokacije. Dok je na listingu 3 prikazan JavaScript kod za kreiranje pokazivača i dodavanje pokazivača na mapu.



Slika 4. OpenStreetMap mapa sa pokazivačem lokacije

```

<script>
function init() {
    var marklonlat = new OpenLayers.LonLat(
        20.448610,44.785811);
    var centar = new OpenLayers.LonLat(
        19.7119942, 44.0570766);
    var osm = new OpenLayers.Layer.OSM();
    map = new OpenLayers.Map('map',
        {
            projection: 'EPSG:3857',
            layers: [
                osm
            ],
            center: centar.transform('EPSG:4326',
                'EPSG:3857'),
            zoom: 7
        });
    var markers = new OpenLayers.Layer.Markers(
        "Beograd!" );
    map.addLayer(markers);
    markers.addMarker(
        new OpenLayers.Marker(
            marklonlat.transform('EPSG:4326', 'EPSG:3857')));
};
</script>

```

Listing 3. JavaScript kod za kreiranje putanje i dodavanje na mapu

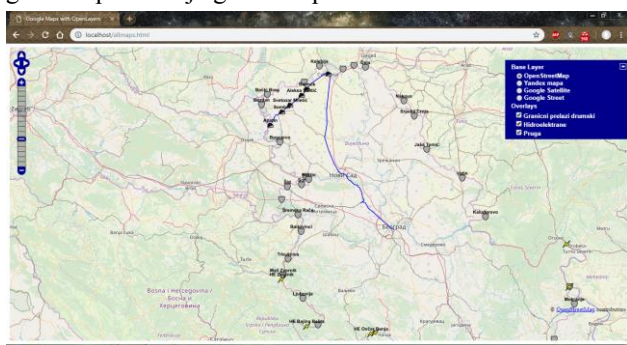
## 6. INTEGRACIJA RASTERSKIH I KORISNIČKI DEFINISANIH SLOJEVA UPOTREBOM OpenLayers-a

U ovom poglavlju opisana je integracija rasterskih i korisnički definisanih slojeva upotrebom OpenLayers biblioteke.

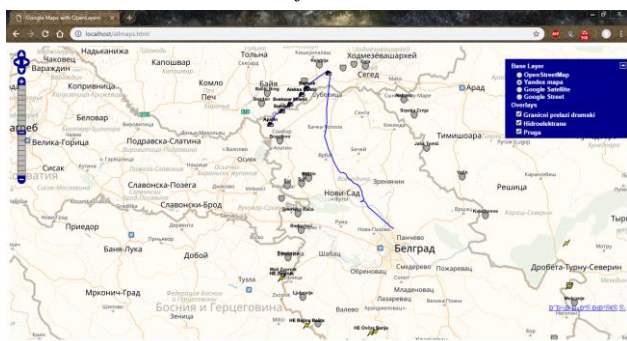
Za integraciju rasterskih i vektorskih slojeva upotrebom OpenLayers biblioteke, potrebno je omogućiti pretraživaču da učita OpenLayers biblioteku.

U GeoServer-u kreirani su vektorski slojevi kojima se grafički prezentuju prostorni podaci smešteni u prostornoj bazi podataka. Vektorski slojevi se generišu upotrebom WMS servisa GeoServer-a.

Na slikama 5 i 6 prikazane su OpenStreetMap mapa preklapljenom slojem koji grafički prezentuje granične prelaze i Yandex mapa takođe preklapljenom slojem koji grafički prezentuje granične prelaze



Slika 5. OpenStreetMap mapa sa učitanim vektorskim slojevima



Slika . Yandex mapa sa učitanim vektorskim slojevima

Na slici 5. može da se vidi da postoje odstupanja između Yandex mape i vektorskih slojeva. Odstupanje postoji zbog različite projekcije mape i vektorskih slojeva. Za projekciju Yandex mape koristi se Merkatorova projekcija (3395 projekcija), dok se vektorski slojevi projektuju u Web Merkatorovoj projekciji (3857 projekcija). Zato je potrebno prilikom kreiranja mape definisati za mapu u kojoj projekciji će biti prikazana. Pošto OpenLayers biblioteka ne podržava 3395 projekciju, nije moguće definisati da se mapa projektuje u navedenoj projekciji. Projekcija koja se koristi za korisnički definisane slojeve je 4326 projekcija i razlikuje se od projekcije u kojoj se nalazi Yandex mapa, a Google i OpenStreetMap se nalaze u 3857 projekciji.

## 7. ZAKLJUČAK

Primenom GIS sistema podaci se mogu grafički prezentovati, umesto čestog pretraživanja podataka i postavljanja upita koji mogu da prave problem korisnicima koji nemaju potrebno znanje iz oblasti koja se bavi prostornim bazama podataka. Primena GIS sistema štedi i vreme i novac na grafičkom prikazu prostornih podataka.

Upotreba OpenLayers biblioteke, GeoServer-a i JavaScript biblioteka razvijenih od strane Google i Yandex kompanija je sve veća zbog mogućnosti integracije mapa u Web aplikacije. Integracija interaktivnih Web mapa u Web aplikacije je moguća uz korišćenje određenih biblioteka koje sadrže klase potrebne za učitavanje mape i prikaz prostornih podataka smeštenih u prostornoj bazi podataka.

## 8. LITERATURA

- [1] Rečnik savremene srpske geografske terminologije, <https://tadic.education/geografski-recnik/>, posećeno u oktobru 2018.
- [2] Wikipedia, <https://en.wikipedia.org/wiki/Ellipsoid>, posećeno u oktobru 2018.
- [3] Hrvatska enciklopedija, <http://www.enciklopedija.hr/natuknica.aspx?id=30683>, posećeno u oktobru 2018.
- [4] Battersby S. E., Finn M. P., Usery E. L., Yamamoto K. H., Implications of Web Mercator and Its Use in Online Mapping, Cartographica The International Journal for Geographic Information and Geovisualization, broj 49 (2014) 85-101
- [5] Kurt E., Prikupljanje i obrada prostornih podataka u GIS-u za potrebe uspostave i održavanja jedinstvene evidencije i registra prostornih jedinica, Geodetski glasnik, broj 42 (2012) 44-52
- [6] Essentials of Geographic Information Systems, [https://saylordotorg.github.io/text\\_essentials-of-geographic-information-systems/](https://saylordotorg.github.io/text_essentials-of-geographic-information-systems/), posećeno u oktobru 2018.

## Kratka biografija:

**Nemanja Jovančević** rođen u Novom Sadu 1991. godine. Master rad na Fakultetu tehničkih nauka iz oblasti Informaciono-komunikacione tehnologije odbranio je 2018. godine kontakt: [nemanjajov@uns.ac.rs](mailto:nemanjajov@uns.ac.rs)