مستندات جی آی اس

B-GIS

فهرست

فصل اول ۴

Geoserver ۵

نصب و راه اندازی GeoServer ۵

War file ۵

نصب پلاگین GDAL ۷

نصبGDAL Native Libraries ۸

نصب پلاگین Oracle Spatial Plugin in GeoServer ۱۰

Stores ۱۱

تعریف data source جئوسرور (از نوع دیتابیسی اوراکل و رستری ecw) ۱۱

اضافه کردن store از نوع ECW ۱۹

Layers ۱۹

Styles ۲۰

نحوه اضافه کردن استایل به لایه های وکتوری ۲۱

نمونه ای از استایل ها در جئو سرور ۲۴

Dashed line ۲۵

Transparent polygon ۲۶

فصل ۲ ..... ۲۷

معماری سامانه ۲۸

Openstreetmap ۲۸

دیتابیس ۲۹

فصل ۳ ..... ۳۱

استفاده از داده های جئومتری در مدل ها ۳۲

چند نمونه از کوئری های جئومتری ۳۳

جدول function های hibernate spatial ۳٤

معرفی نوع داده Geometry : ۳۵

نمونه داده های جئومتری ذخیره شده در دیتابیس ۳۷

WKT(Well Known Text) ۳۷
کلاس WktUtility ۳۹
نمونه ای از فایل xml مدل جئومتری ٤۰
مدیریت لایه ها ٤۰
تقسیمات کشوری ٤۲
کلاس Style ٤٤
انواع سازنده های کلاس Style ۴۵
انواع کلاس های استفاده شده مرتبط با کلاس Style ۴۵
نحوه ارتباط سامانه ها با Baharan_GIS ٤۸
Map-config.js ۴۸
Tracking ٥۷
فصل۴ ..... ۶۱
راهنمای نقشه ٦۲
جستجو ۶۲
لایه ها ۶۳
دسته بندی ۶۳
مسیریابی ۶۸
تعیین مسافت ۷۰
نوار Navigation ۷۱

فصل اول

GeoServer

Geoserver

برای شروع کار GIS ابتدا نیاز به یک سرور داریم تا بتواند نقشه های مورد نظر ما از فایل های رستری و وکتوری تا نقشه هایی که به صورت دیتابیسی ذخیره میشوند را نمایش دهد علت اصلی استفاده از این سرور برای نمایش اطلاعات به صورت آفلاین است . در صورت استفاده از روش آنلاین . برای انجام این کار از geoserver استفاده میکنیم که میتوان jar فایل مربوط به آن را در محل مورد نظر استارت کرد و از آن استفاده کرد.

نصب و راه اندازی GeoServer

جئو سرور سروری است برای لود کردن نقشه های جغرافیایی با فرمت های مختلف ، اگر نیاز داشته باشید در اپلیکیشن خود نقشه ای لود کنید و روی آن عملیاتی مانند مسیریابی ، علامت گذاری و ... انجام دهید بسته به نیازتان میتوانید از سرور هایی مانند GeoServer ، ArcGIS و ... استفاده نمایید که در اینجا به نحوه نصب و راه اندازی GeoServer می پردازیم و در ادامه چگونگی نصب پلاگین GDAL را شرح میدهیم تا امکان لود فرمت هایی مانند ecw ، ehdr و ... به GeoServer اضافه کنیم.

برای راه اندازی میتوانید از لینک زیر GeoServer را دانلود کنید

http://geoserver.org/download/

از صفحه زیر میتوانید پکیج مورد نظر خودتون رو دانلود کنید

http://geoserver.org/release/۲,۱۲,۰/

برای مثال میتونید از فایل نصبی ویندوز یا war فایل استفاده کنید تا جئو سرور را راه اندازی کنید

War file

برای مثال اگر از war فایل استفاده میکنید کافیه که فایل مورد نظر رو زیر پوشه webappsدر مسیر تامکت قرار بدید و از پوشه bin فایل startup.bat را اجرا کنید تا تامکت سرور را استارت کند.

حالا برای باز کردن داشبورد GeoServer کافیه به آدرس زیر بروید

http://localhost:۸۰۸۰/geoserver

برای ورود از username و password پیش فرض زیر استفاده کنید

Username : admin

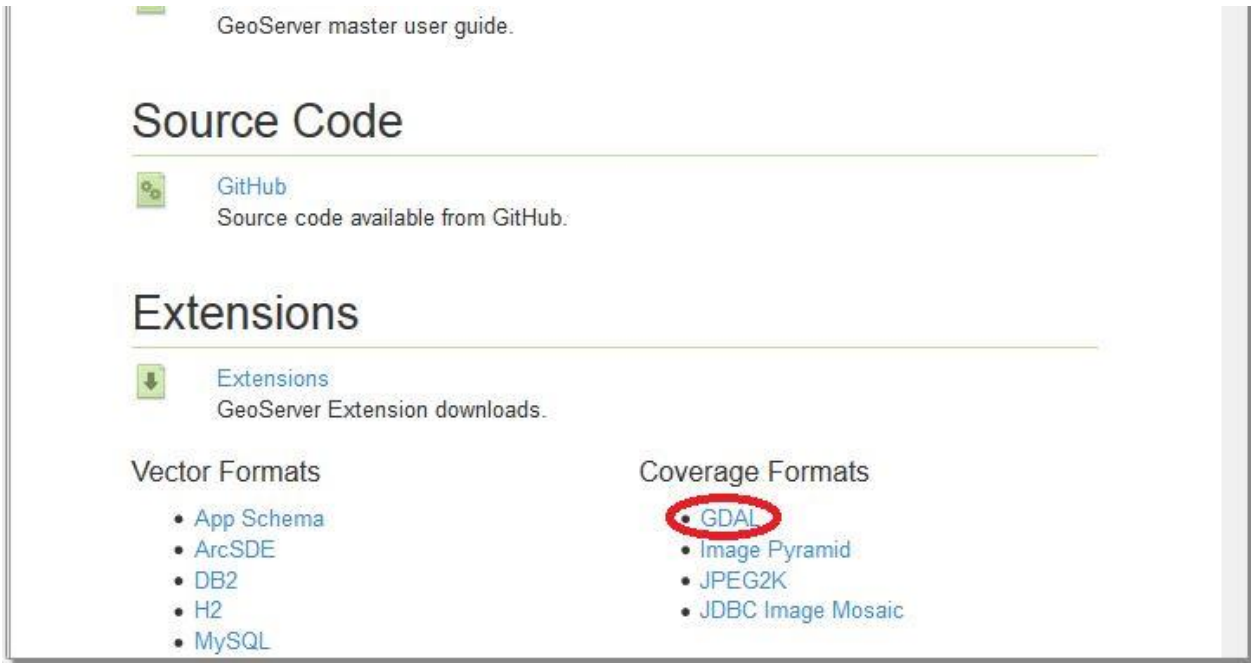
Password : geoserver

نصب پلاگین GDAL

یکی از پسوند های مرسوم رستری در GeoServer فایل ecw میباشد که برای لود کردن این فایل های نیازمند پلاگین GDAL هستیم ، برای نصب میتوانید از لینک زیر استفاده کنید

http://docs.geoserver.org/latest/en/user/data/raster/gdal.html

برای نصب ابتدا به آدرس بالا رفته و همانطور که در شکل مشاهده میکنید پلاگین را دانلود میکنید



حالا به دایرکتوری که war فایل خود را قرار داده اید بروید (پوشه webapps تامکت را باز کنید) داخل پوشه geoserver ، سپس WEB-INF و پوشه lib بروید و فایل دانلود شده را درون پوشه extract کنید و بعد از مواجهه با پیغام وجود داشتن فایل ها در این پوشه ، گزینه raplace all را انتخاب کنید

نصب GDAL Native Libraries

جئو سرور برای استفاده از این لایبراری ها باید لایبراری GDAL رو از طریق سیستم عامل نصب کنه تا بتونه استفادشون کنه برای این کار باید عملیات زیر را انجام بدیم

در ابتدا فایل gdal-data.zip را از آدرس زیر دانلود میکنید

http://demo.geo-solutions.it/share/github/imageio-ext/releases/۱,۱.X/۱,۱,۱۶/native/gdal

حالا برای نصب native library به آدرس زیر رفته و بسته به نوع سیستم عامل خود یکی را نتخاب میکنید برای مثال ما وارد پوشه ویندوز میشویم و در اینجا با توجه به نوع سیستم عاملتان که ۶۴ یا ۳۲ بیتی میباشد باید گزینه درست را انتخاب کنید. برای مثال ویندوز ۱۰ نسخه ۶۴ بیتی نیازمند این نسخه میباشد

gdal-۱,۹,۲-MSVC۲۰۱۰-x۶۴.zip

http://demo.geo-solutions.it/share/github/imageio-ext/releases/۱,۱.X/۱,۱,۱۶/native/gdal

حال فایل های دانلود شده را در دایرکتوری مناسبی قرار دهید و عملیات زیر را انجام دهید

به تب advance در system properties بروید و environment variables را باز کنید

در قسمت user_variables یک متغیر تعریف کنید به نام GDAL_DATA و آدرس محلی که فایل gdal-data.zip را در آن extract کردید را بدهید

حالا در system variables متغیری با نام GDAL_DRIVER_PATH ایجاد کرده و آدرس فولدری که فایل های دی ال ال ECW در آن قرار دارد را بدهید بدهید این فایل ها در مسیر زیر قرار دارند

.C:\Program Files\GDAL\gdalplugins

و در نهایت در system variables به دنبال متغیر path بگردید و editرا بزنید ، سپس آدرسی را که برای GDAL_DRIVER_PATH دادید را به آن اضافه کنید و ok را بزنید

حالا سرور را استارت کنید از پنل سمت چپ ، بخش دیتا ، بر روی stores کلیک کنید . اینجا جایی هست که ما دیتا سورسی برای جئو سرور ایجاد میکنیم تا بتواند بر اساس فایل ، دیتابیس و ... یک نقشه یا لایه را لود کنید

با انتخاب گزینه add new store میتوانیم مشاهده کنیم در بخش raster data sources گزینه های جدیدی من جمله ecw اضافه شده است که میتوانیم از آنها استفاده کنیم

تذکر : لازم به ذکر است اگر از ورژن ۱,۹ GDAL به بالا استفاده میکنید میبایست فایل زیر را

imageio-ext-gdal-bindings-۱,۹,۲.jar را از پوشه java info که در پوشه gdal-۱,۹,۲-MSVC۲۰۱۰-x۶۴ قرار دارد کپی کرده و در زیر پوشه web-apps\geoserver\WEB-INF\lib قرار دهید

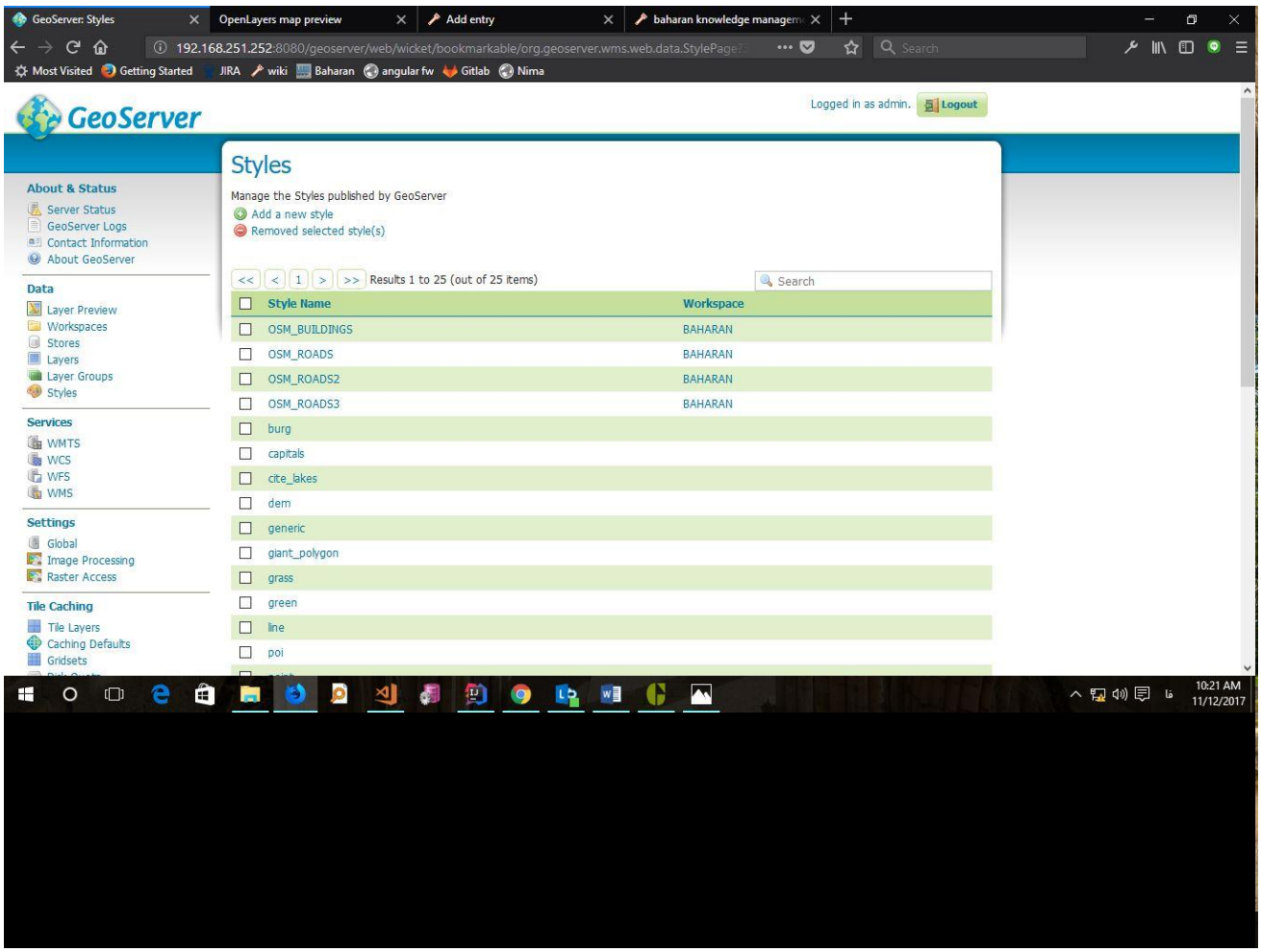
اگر این کار را انجام ندهید به هنگام استارت شدن سرور با خطا مواجه میشوید.

نصب پلاگین Oracle Spatial Plugin in GeoServer

برای لود کردن دیتای نقشه از دیتابیس اوراکل توسط GeoServer باید پلاگین oracle را نصب کنیم

برای این منظور ابتدا اکستنشن اوراکل رو از این قسمت دانلود میکنید توجه داشته باشید که ورژن این پلاگین ها باید با ورژن جئوسرور یکی باشد

همانطور که در شکل زیر مشاهده میکنیداز بخش vector formats گزینه oracle را انتخاب کنید و فایل های مورد نیاز را دانلود کنید.



و آن را در مسیر WEB-INF/lib ، از حالت extract خارج کنید

سپس به سایت اوراکل رفته و JDBC Driver مربوطه رو دانلود میکنید و در مسیر بالا قرار میدهید

با انجام این کار و استارت کردن جئوسرور با رفتن به قسمت store و انتخابadd new store در بخش وکتور ها در کنار postGis نام oracle هم مشاهده میشود

Stores

در واقع دیتا ست های ما هستند که در این بخش میتوانیم با استفده از دیتا های مختلف (عکس ، شیپ فایل ، دیتابیس و ...)دیتا سورس های مختلفی را به سرور خود اضافه کنیم

تعریف data source جئوسرور (از نوع دیتابیسی اوراکل و رستری ecw)

برای این منظور ابتدا وارد صفحه اصلی جئوسرور شده و لاگین میکنیم

localhost:۸۰۸۰/geoserver

user : admin

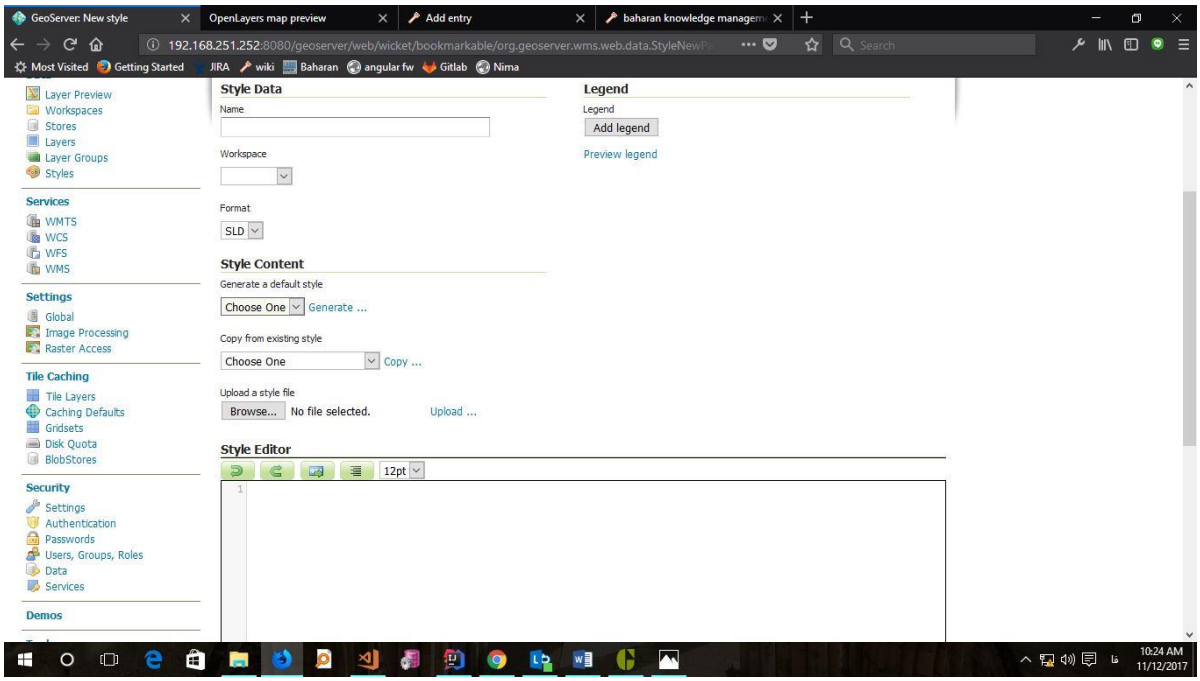
pass:geoserver

در نظر داشته باشید که برای نمایش هر نقشه به چند چیز نیاز داریم

Workspace

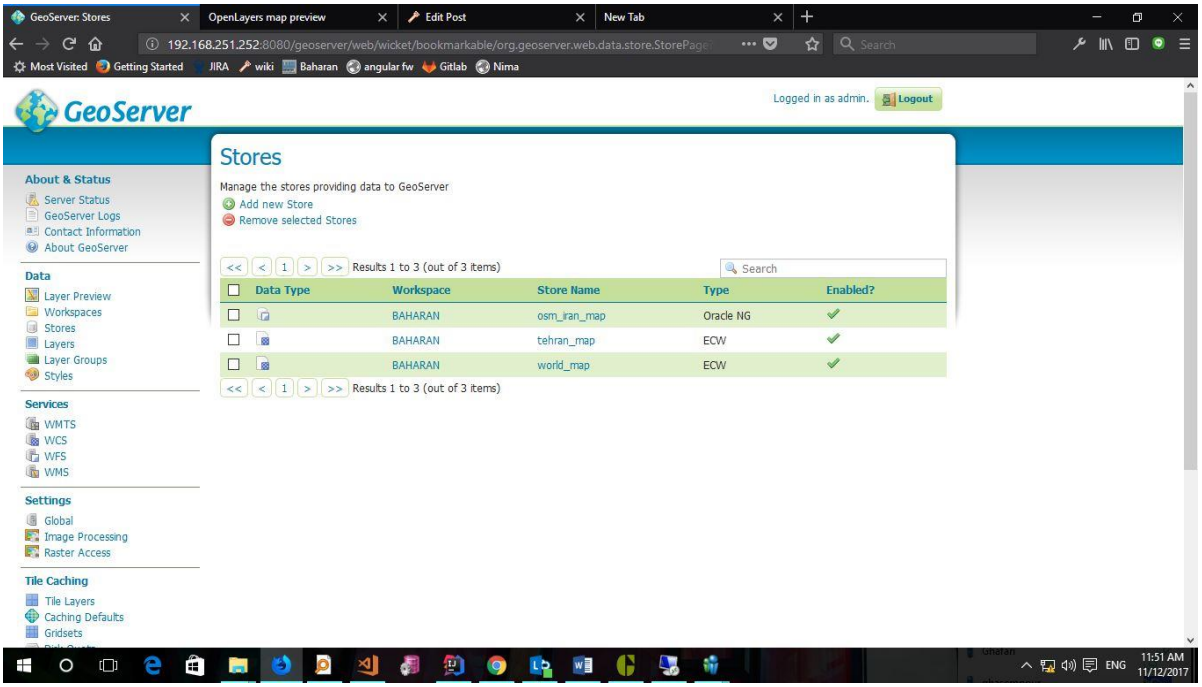
بهتره که در شروع کار برای خودتون یک workspace آماده کنید تا کارهاتون نظم بهتری داشته باشه و دیتاسورس هاتون رو به اون وصل کنید

برای این کار از پنل سمت چپ (بخش Data) روی workspaces کلیک میکنید نام و آدرس دلخواهی را انخاب کنید و submit کنید

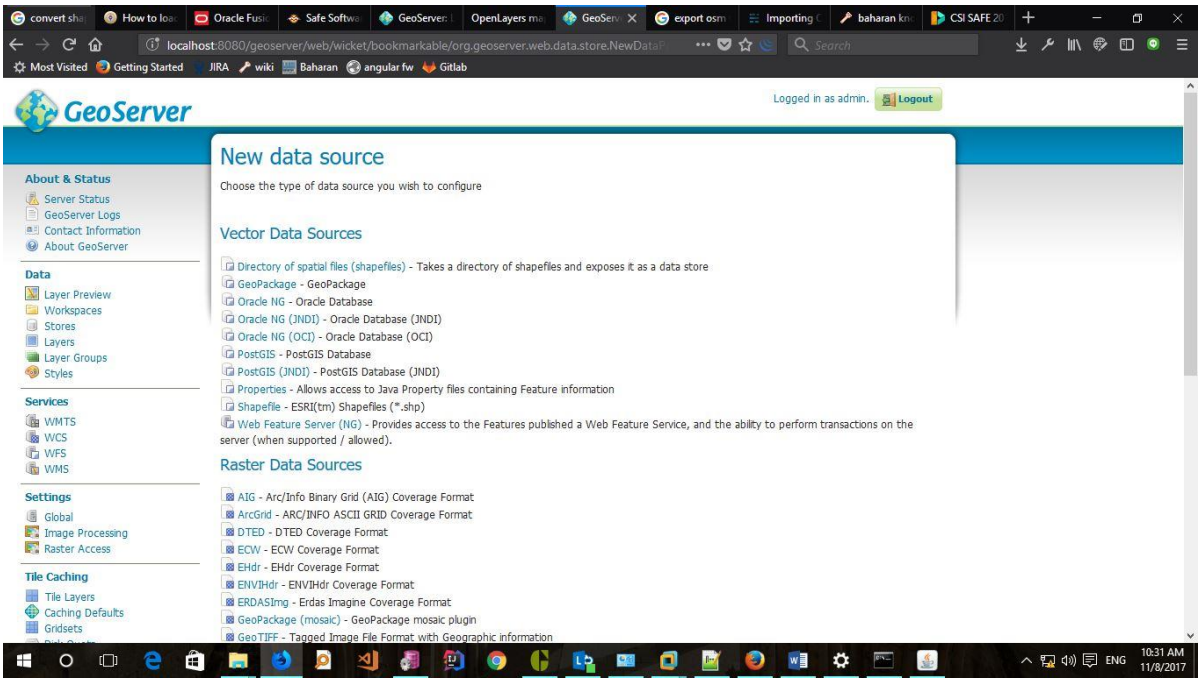


Stores : store ها همان دیتا سورس هستند که میتوانند به صورت رستر یا وکتور باشند که ما در این مثال با چند نمونه از آن ها آشنا میشویم

برای ایجاد یک store جدید از پنل سمت چپ بر روی store که با آیکون دیتابیس مشخص شده است کلیک کنید

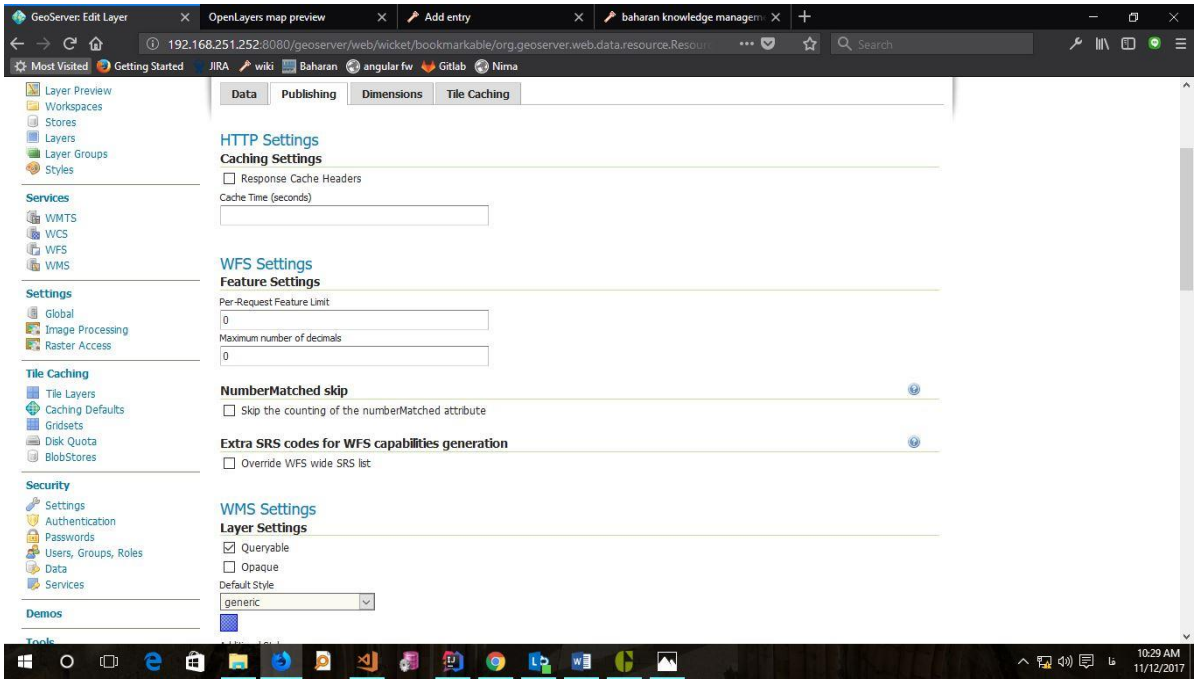


با کلیک روی این گزینه لیستی از store های موجود را مشاهده میکنید . با کلیک روی add new store در بالای صفحه به صفحه که در زیر مشاده میکنید هدایت میشوید.



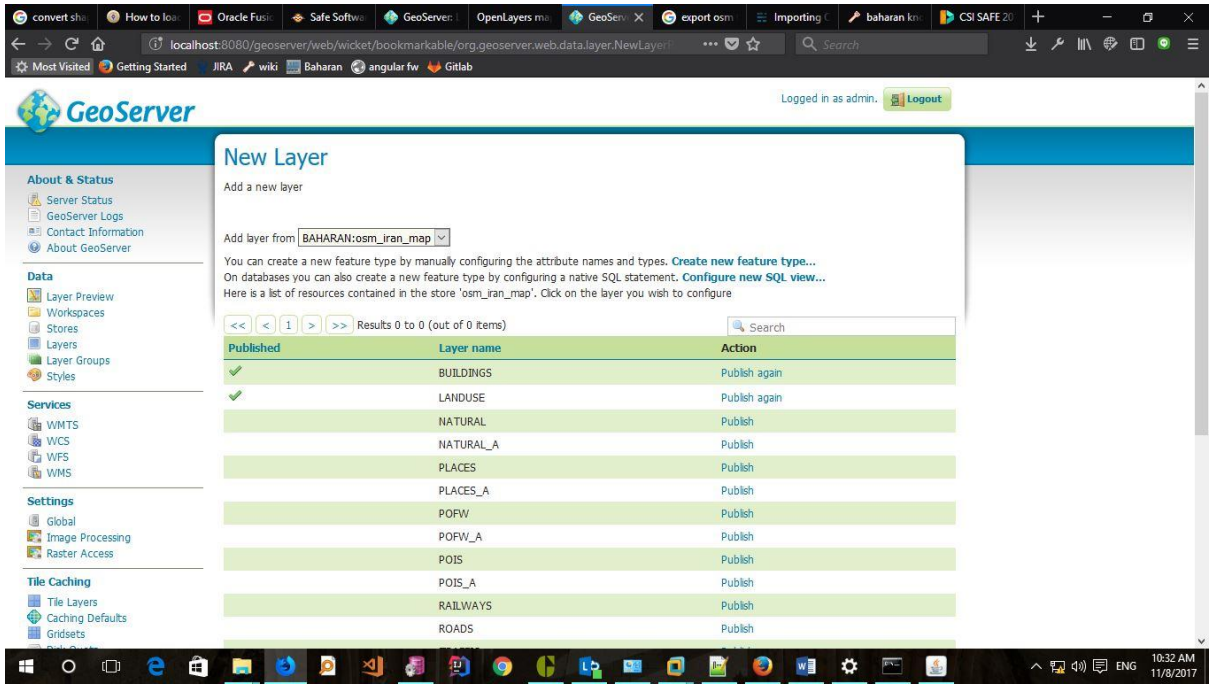
در ان قسمت قصد نصب یک دیتا سورس اوراکلی که دیتای جئومتری آن را از سایت openstreetmap دانلود و به اوراکل convert کرده بودیم را داریم.

بر روی oracle NG کلیک میکنیم

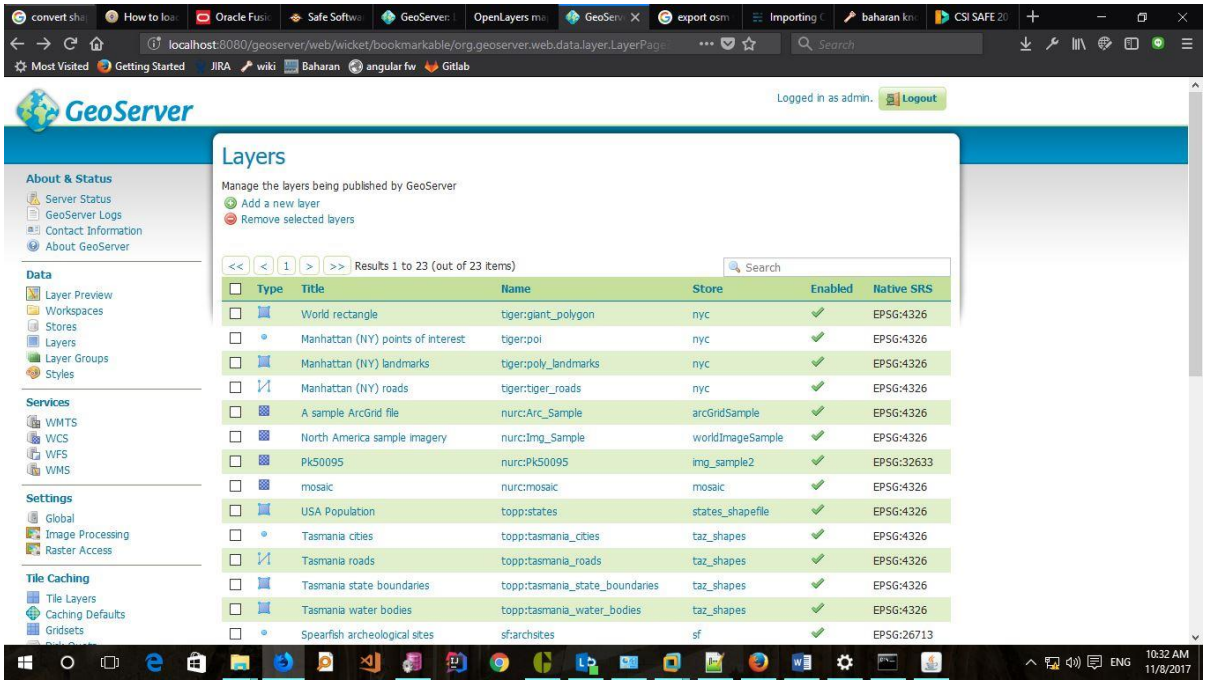


در صفحه باز شده مشخصات دیتابیس را وارد میکنیم (همان طور که در بالا مشاهده میکنید) تنها connection parameters را پر کنید (تا password) و save کنید

بعد از ذخیره به صفحه layers هدایت میشود و باید تک تک جدول هایی که در دیتابیس قرار داشته اند را publish کنید در حقیقتت آن ها را به یک لایه تبدیل کنید

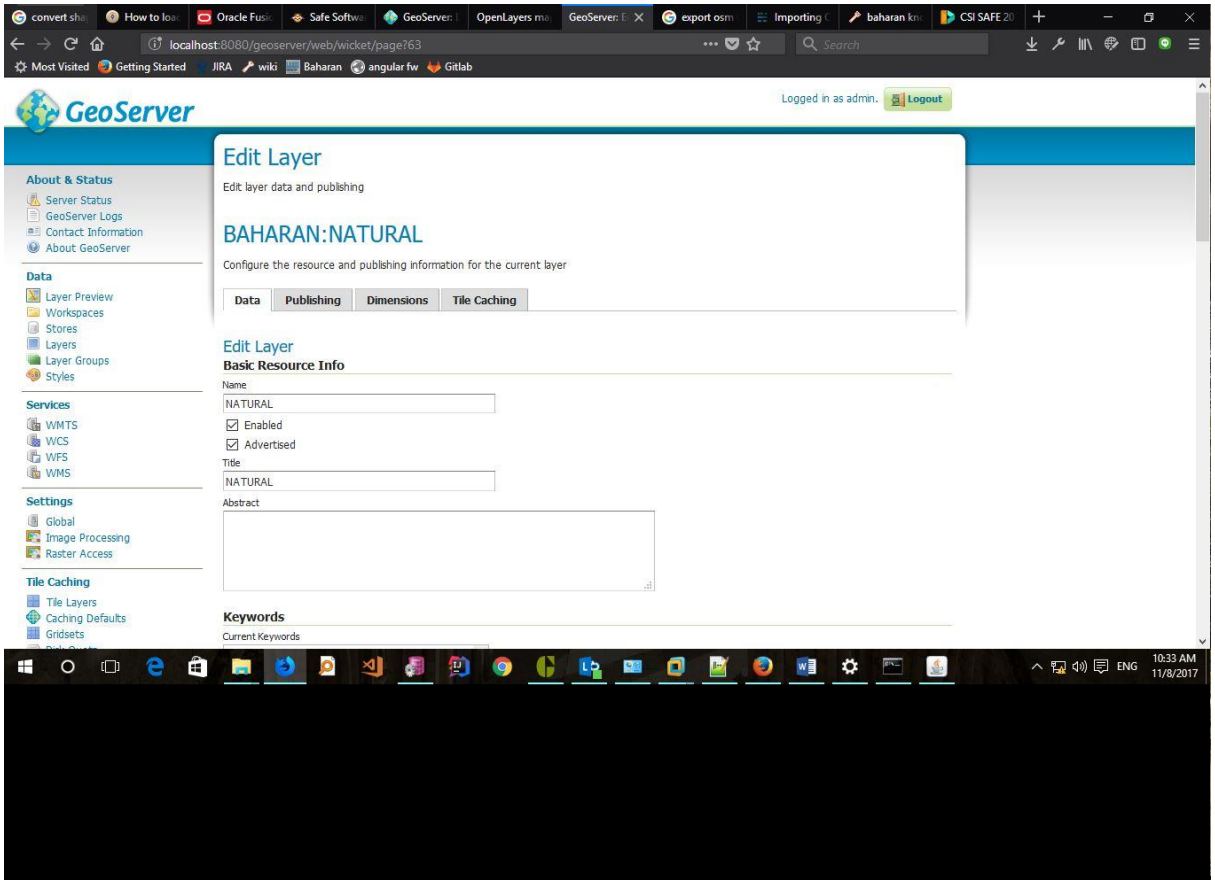


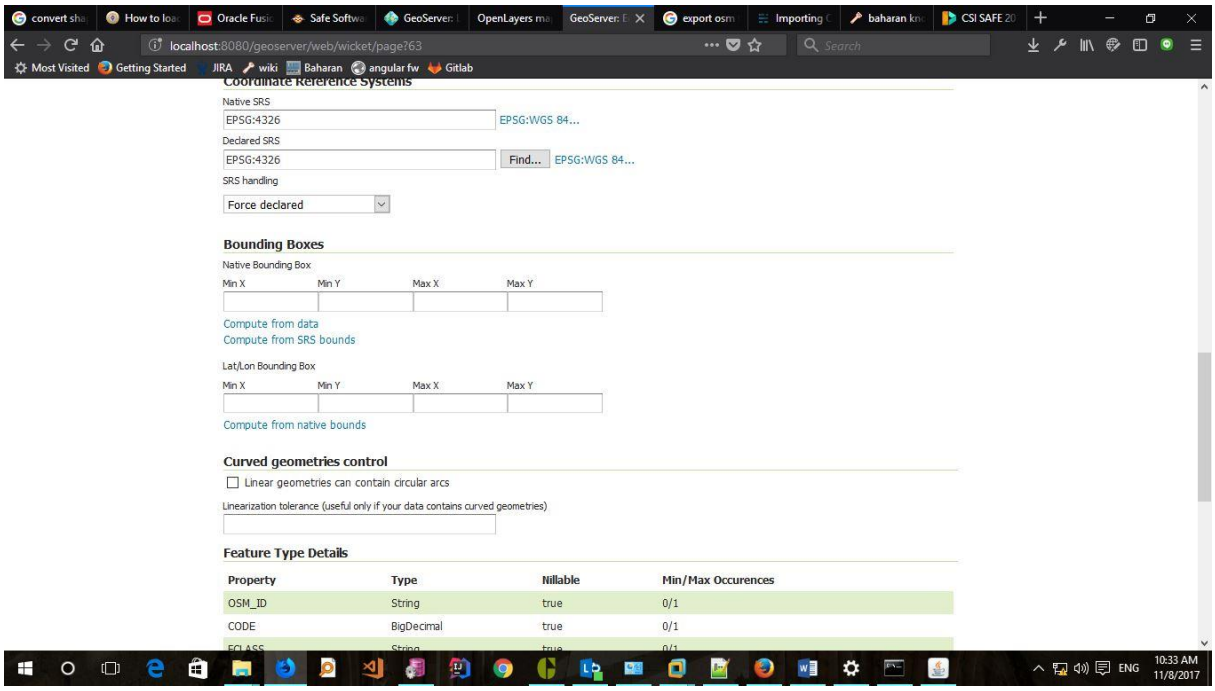
بعد از هر بار انجام عملیات publish برای دیدن این بخش و تکمیل لایه های دیتابیس ، از پنل سمت چپ روی layers کلیک کنید



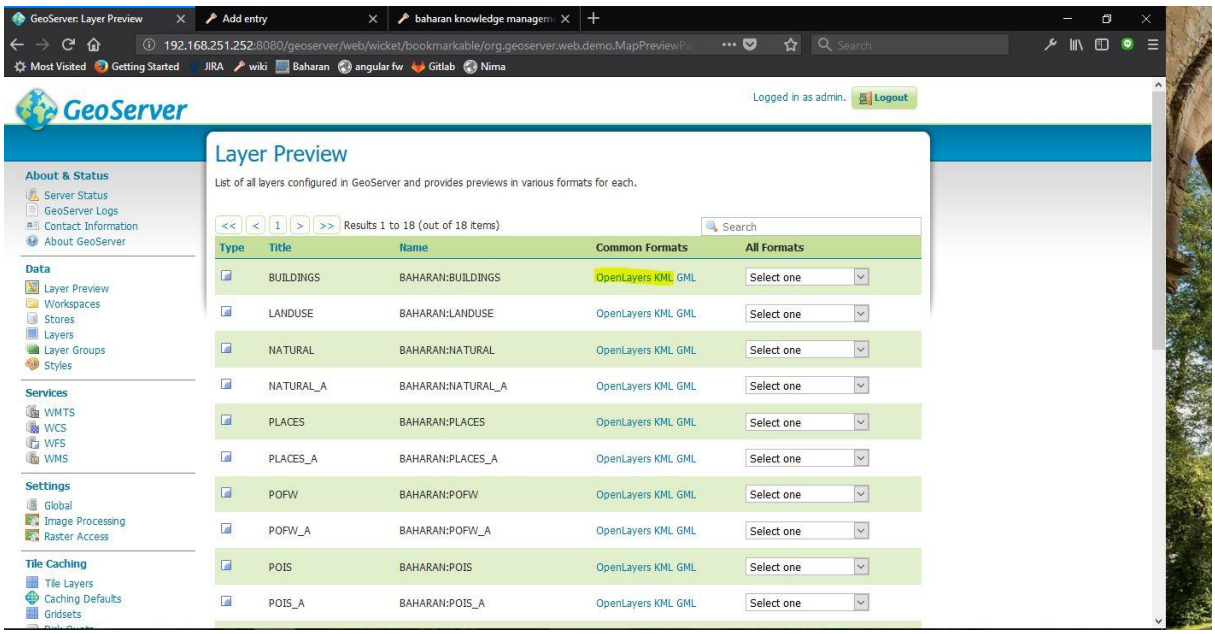
بر روی add a new layer کلیک کنید و از کمبو add layer from دیتاسورس خودتون رو لود کنید.
تا لیست جداول را مشاهده کنید

حالا برای تولید لایه ها روی publish کلیک کنید تا به صفحه زیر هدایت شوید

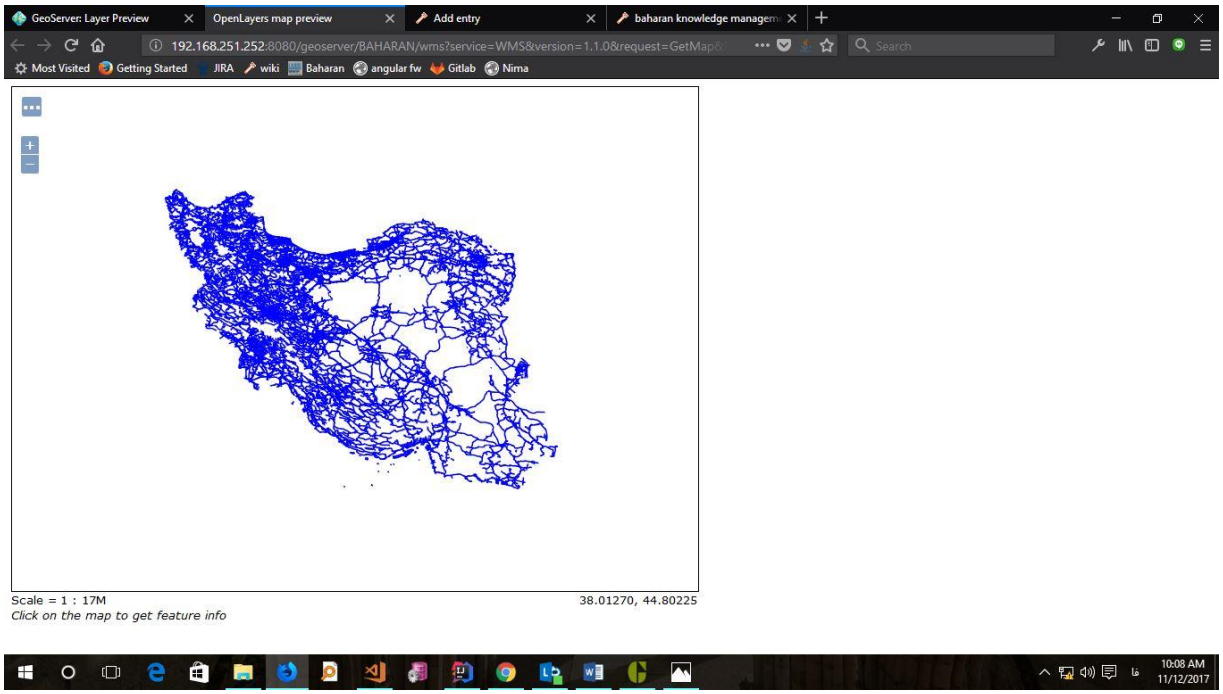




در این بخش نام و عنوان را مشخص کنید و در بخش boundings گزینه compute from data را بزنید تا از روی دیتابیس متوجه شود که کمترین بیشترین مقادیر x و y چیست تا بر اساس آن کل نقشه را در اولین لود به صورت یکجا نمایش دهد.بر روی گزینه save کلیک کنید حالا برای نمایش نقشه کافی است از پنل سمت چپ به layer preview بروید . اینجا جایی است که کلیه لایه ها برا نمایش وجود دارند.

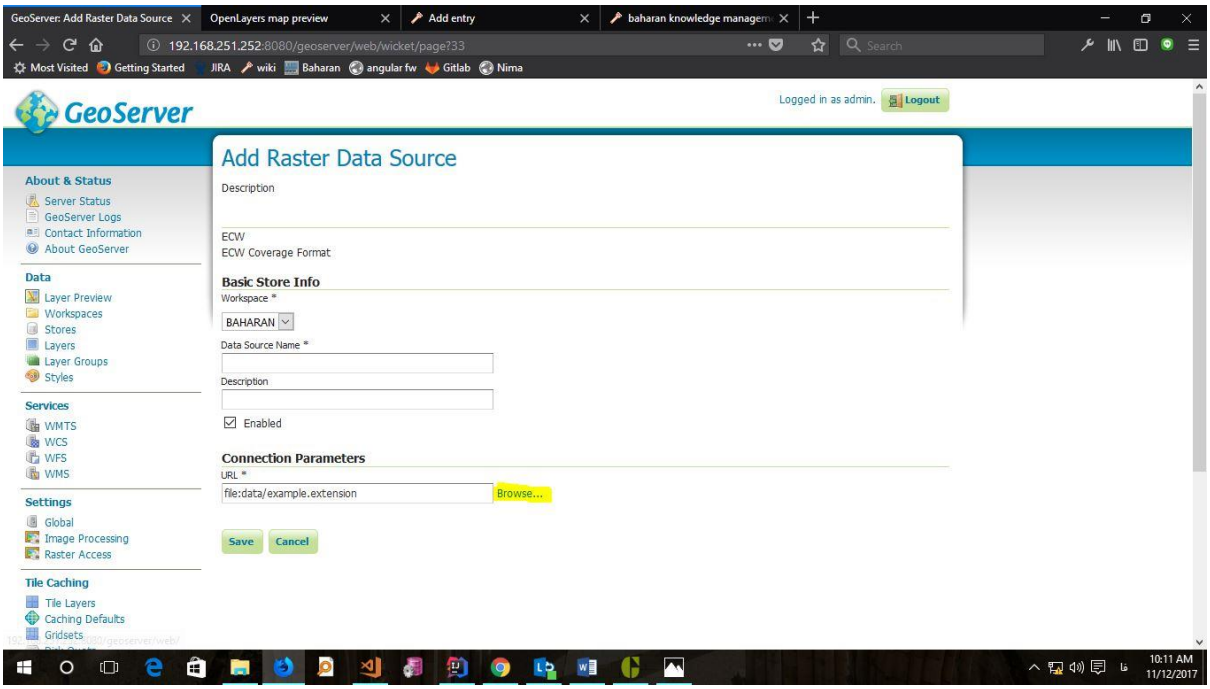


کافی است بر روی open layers کلیک کنید تا نقشه به شما نمایش داده شود.


GeoServer: Layer Preview
OpenLayers map preview
Add entry
baharan knowledge managem
192.168.251.252:8080/geoserver/BAHARAN/wms?service=WMS&version=1.1.0&request=GetMap&
Search
Most Visited
Getting Started
JIRA
wiki
Baharan
angular fw
Gitlab
Nima
Scale = 1 : 17M
38.01270, 44.80225
Click on the map to get feature info
10:08 AM
11/12/2017

اضافه کردن store از نوع ECW

برای این منظور کافیه در هنگام ایجاد store جدید که در بالا گفته شد گزینه ECW را انتخاب کنید

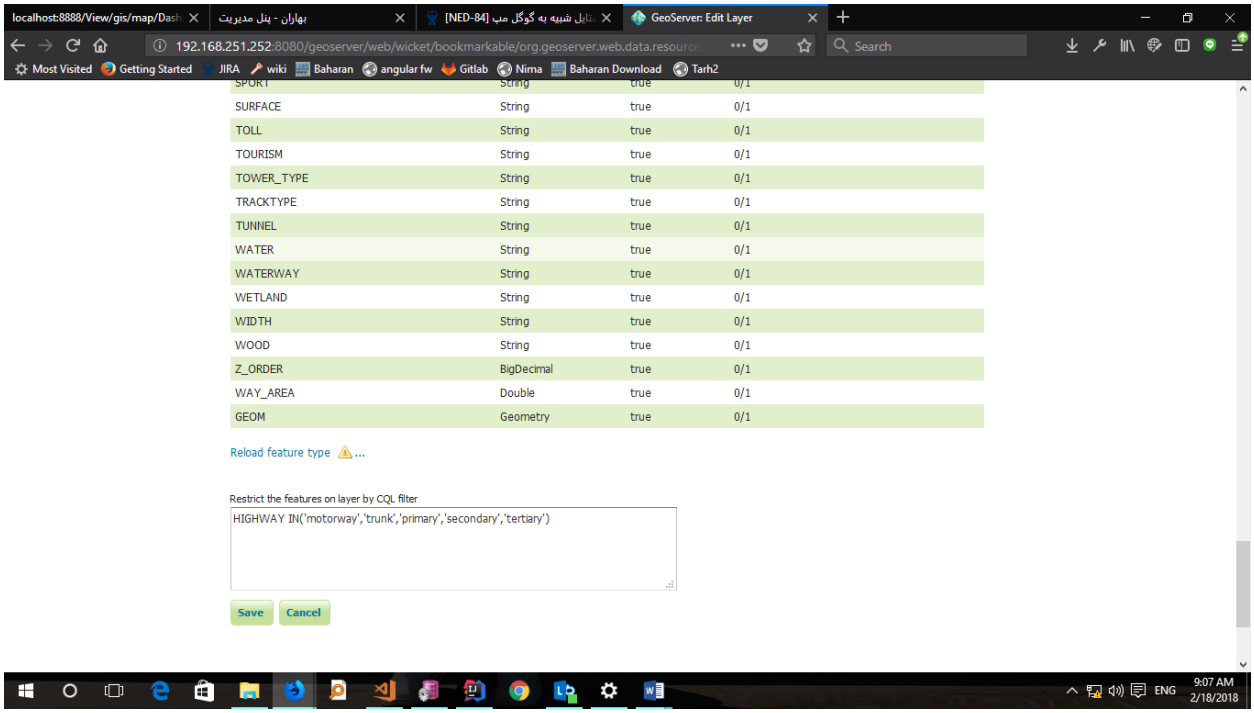


مشخصات دلخواهتان را وارد کنید و سپس آدرس فایل ecw مورد نظر را که حتما آن را باید در دیتا دایرکتوری خود اپلیکیشن قرار دهید را انتخاب کنید و ذخیره کنید

سپس به صفحه layer هدایت میشوید تا لایه را publish کنید عملیات را مانند مثال قبلی انجام دهید و در نهایت لایه شما آماده نمایش است

Layers

برای بهره برداری از دیتاسورس ها layer هایی با تنظیمات دلخواه میسازیم و میتوانیم فیلتر های خاصی برای نمایش را روی آن ها اعمال کنیم . برای مثال لایه ای که نمایانگر پمپ بنزین های سطح شهر باشد. برای این منظور میتوان در هنگام استفاده از داده های دیتابیسی از کوئری های CQL یا ECQL بهره برد .

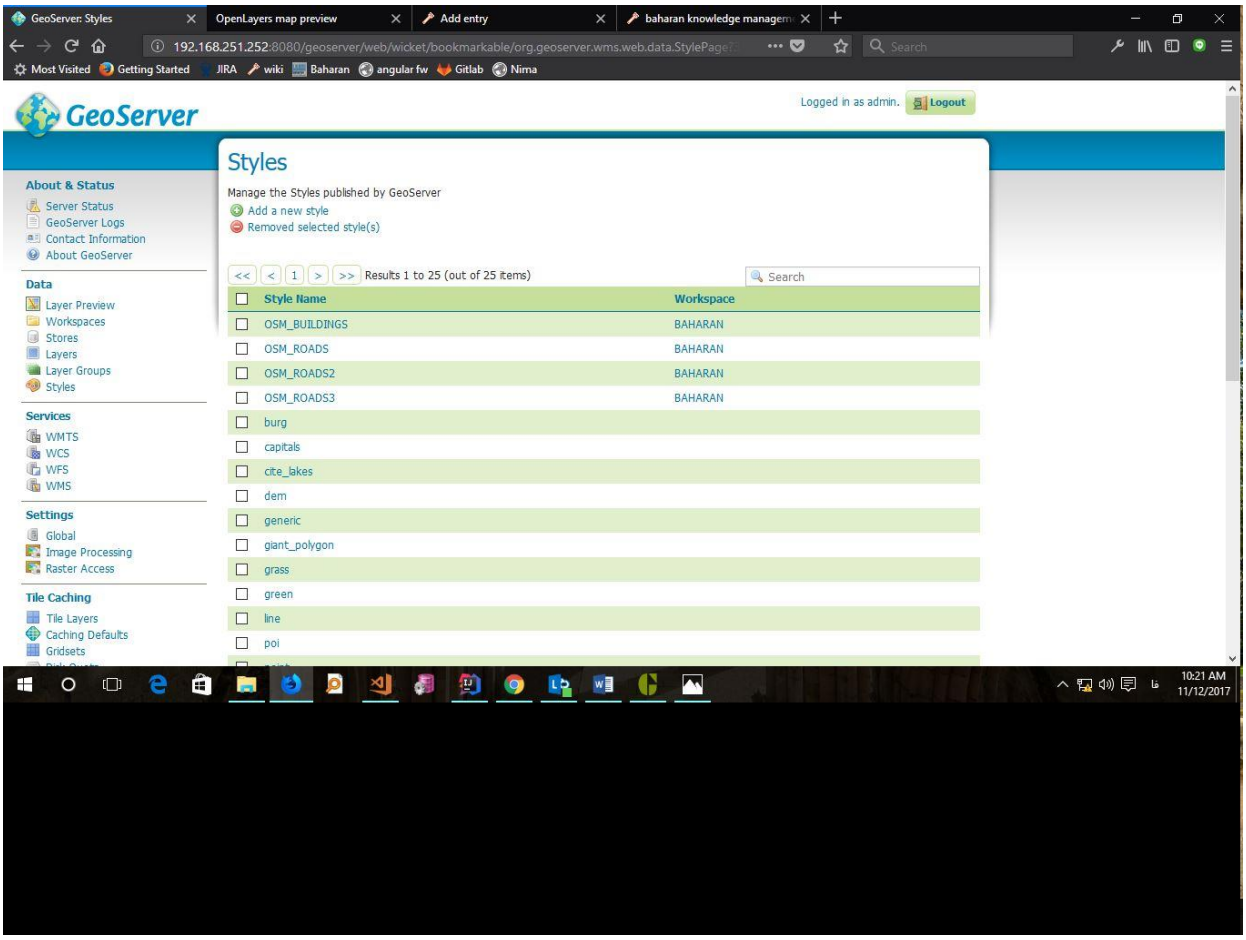


Styles

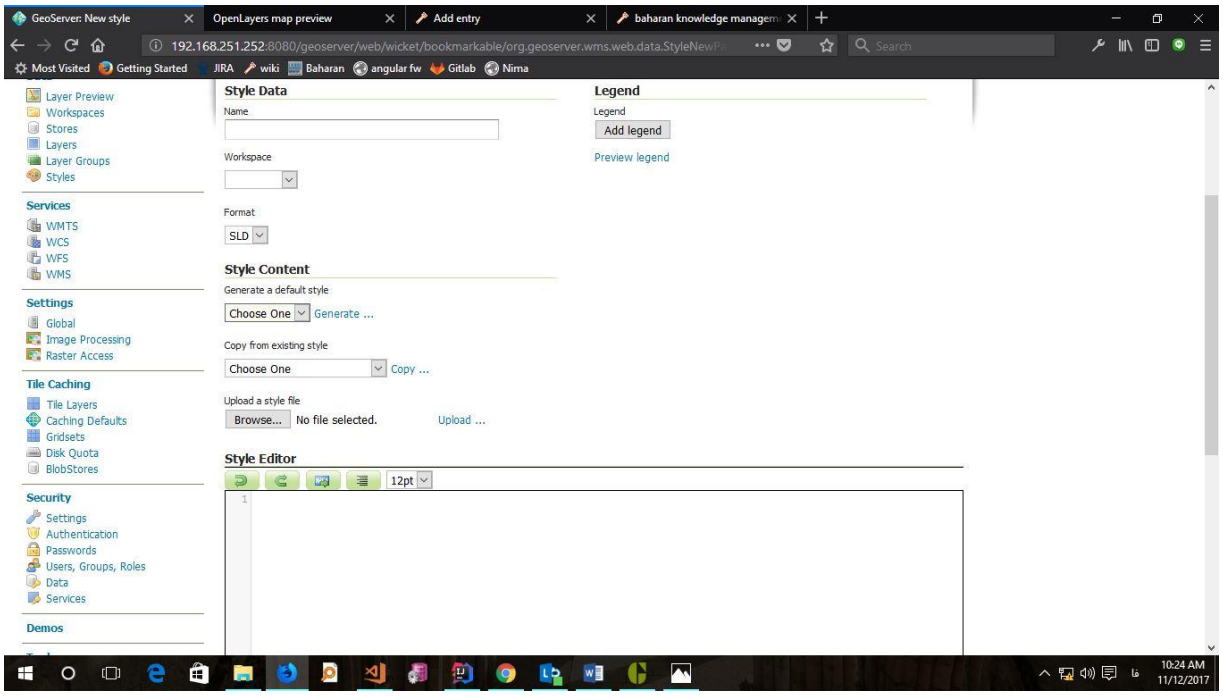
استایل ها نقش مهمی را در نمایش نقشه ها بر عهده دارند که انجام این کار در جئوسرور با فایل xml صورت میپذیرد . به عنوان مثال برای نمایش نقشه ماهواره ای ما با استفاده از ترکیب دو لایه وکتوری مسیر های کشور که از سایت OSM داشتیم و فایل رستری تهران استفاده کردیم و در نهایت با اضافه کردن استایل ها نمایی همانند شکل زیر را ایجاد کردیم تا به هنگام نمایش نقشه ماهواره ای شهر، مسیر ها و نام خیابان ها قابل مشاهده باشند.

نحوه اضافه کردن استایل به لایه های وکتوری

برای این کار از پنل سمت چپ به بخش styles بروید. در این جا لیست استایل های موجود را مشاهده میکنید . روی گزینه add a new style کلیک کنید تا بتوانید استایل جدید را ایجاد کنید.

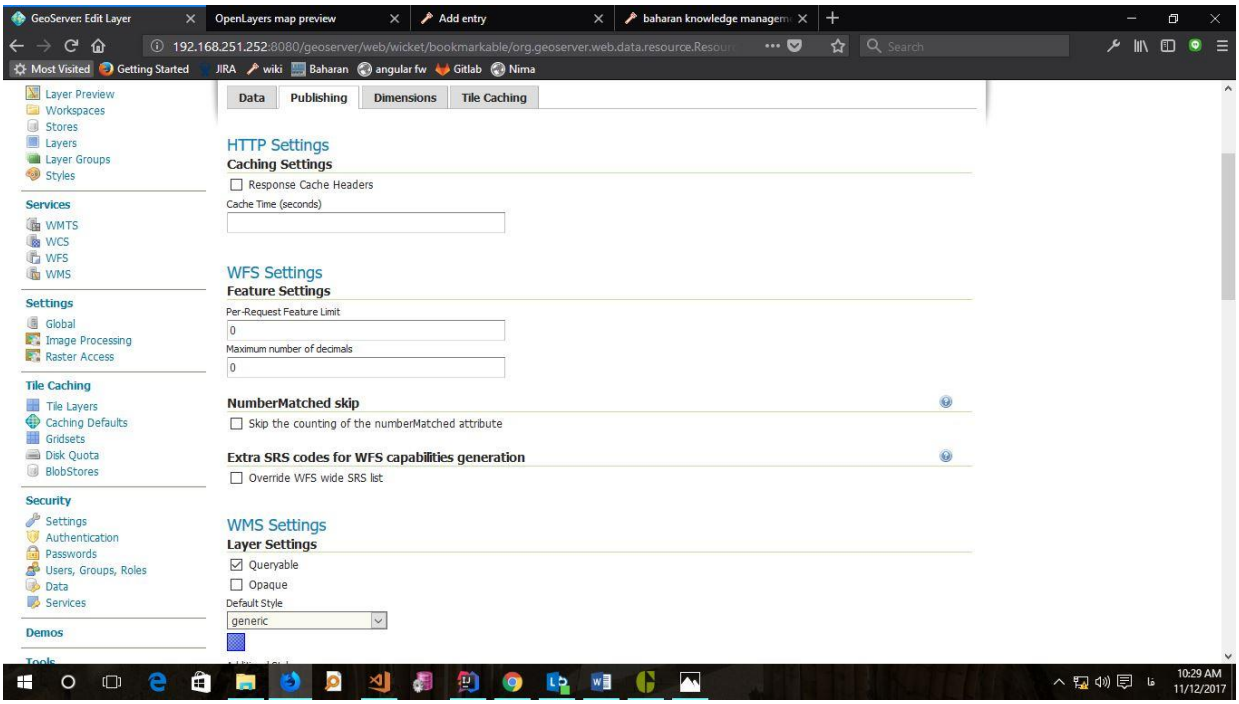


نام دلخوا و workspace را انتخاب کنید و در بخش style editor میتوانید xml استایل های خود را paste کرده و ذخیره کنید

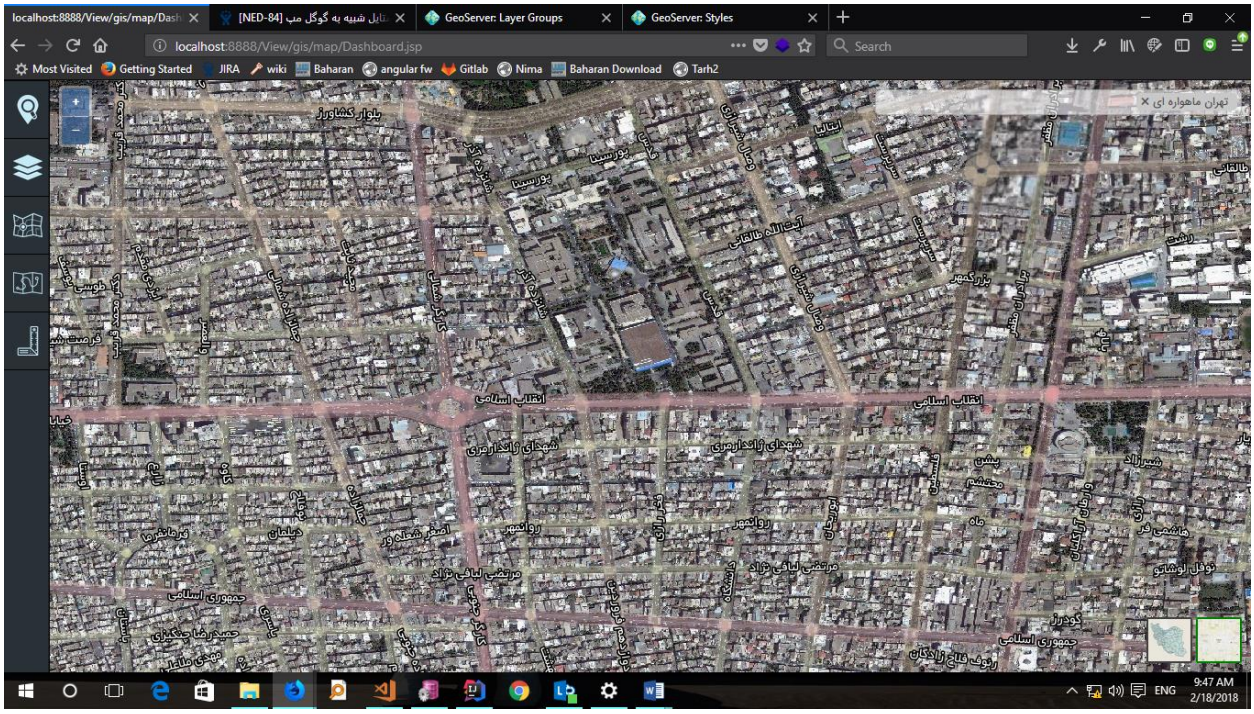


Refrence : geoServer styles cookbook

برای اعمال این استایل باید به بخش layers برگردید بعد از باز کردن یک لایه به صورت جدید یا ویرایش وارد تب دوم با نام publishing میشویم



و از قسمت default style همان استایلی که خودمان ساخته ایم را می آوریم و به لایه اضافه میکنیم و سپس ذخیره میکنیم حالا وقتی لایه را در بخش layer preview انتخاب میکنیم استایل مورد نظرمان به آن اضافه شده است



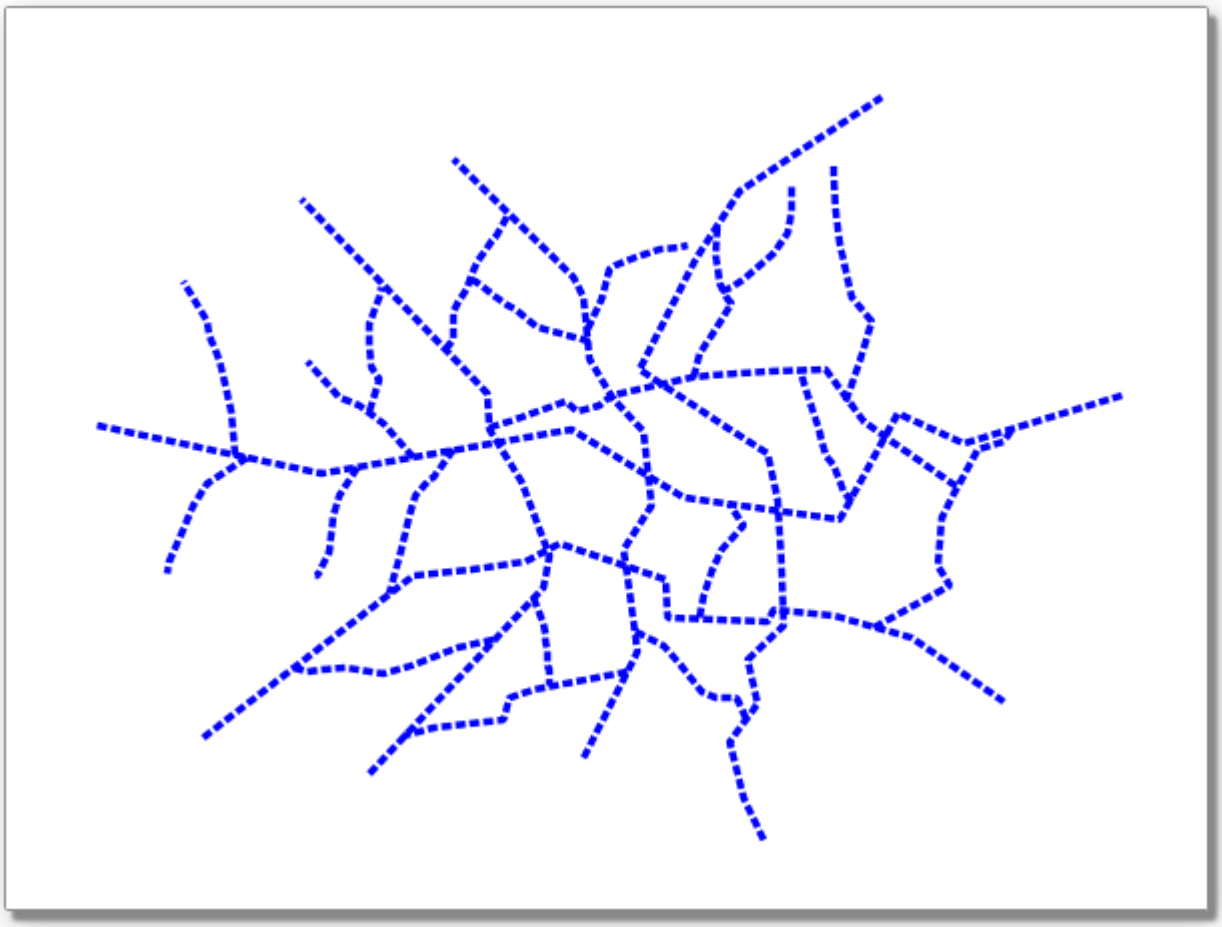
به صورت کلی عناصر نقشه از سه جزء اصلی نقطه ، خط و چند ضلعی تشکیل میشوند که در کنار هم و یا به صورت تکی نقشه هایی با شمایل خاص ، ایجاد میکنند.

نمونه ای از استایل ها در جئو سرور

مثال نقطه با دور خط

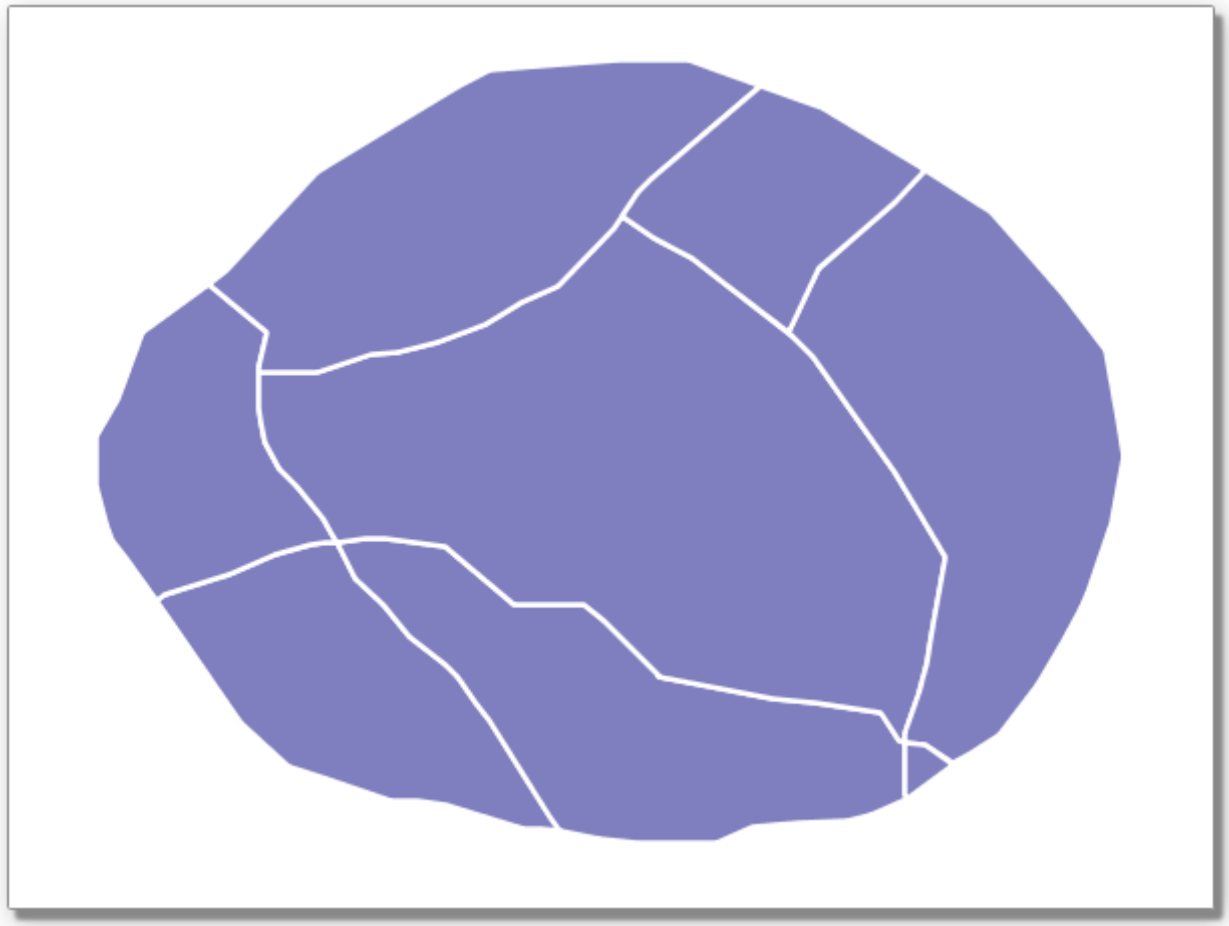
```
<FeatureTypeStyle>
  <Rule>
    <PointSymbolizer>
      <Graphic>
        <Mark>
          <WellKnownName>circle</WellKnownName>
          <Fill>
            <CssParameter name="fill">#FF0000</CssParameter>
          </Fill>
          <Stroke>
            <CssParameter name="stroke">#000000</CssParameter>
            <CssParameter name="stroke-width">2</CssParameter>
          </Stroke>
        </Mark>
        <Size>6</Size>
      </Graphic>
    </PointSymbolizer>
  </Rule>
</FeatureTypeStyle>
```

Dashed line



```
<FeatureTypeStyle>
  <Rule>
    <LineSymbolizer>
      <Stroke>
        <CssParameter name="stroke">#0000FF</CssParameter>
        <CssParameter name="stroke-width">3</CssParameter>
        <CssParameter name="stroke-dasharray">5 2</CssParameter>
      </Stroke>
    </LineSymbolizer>
  </Rule>
</FeatureTypeStyle>
```

Transparent polygon

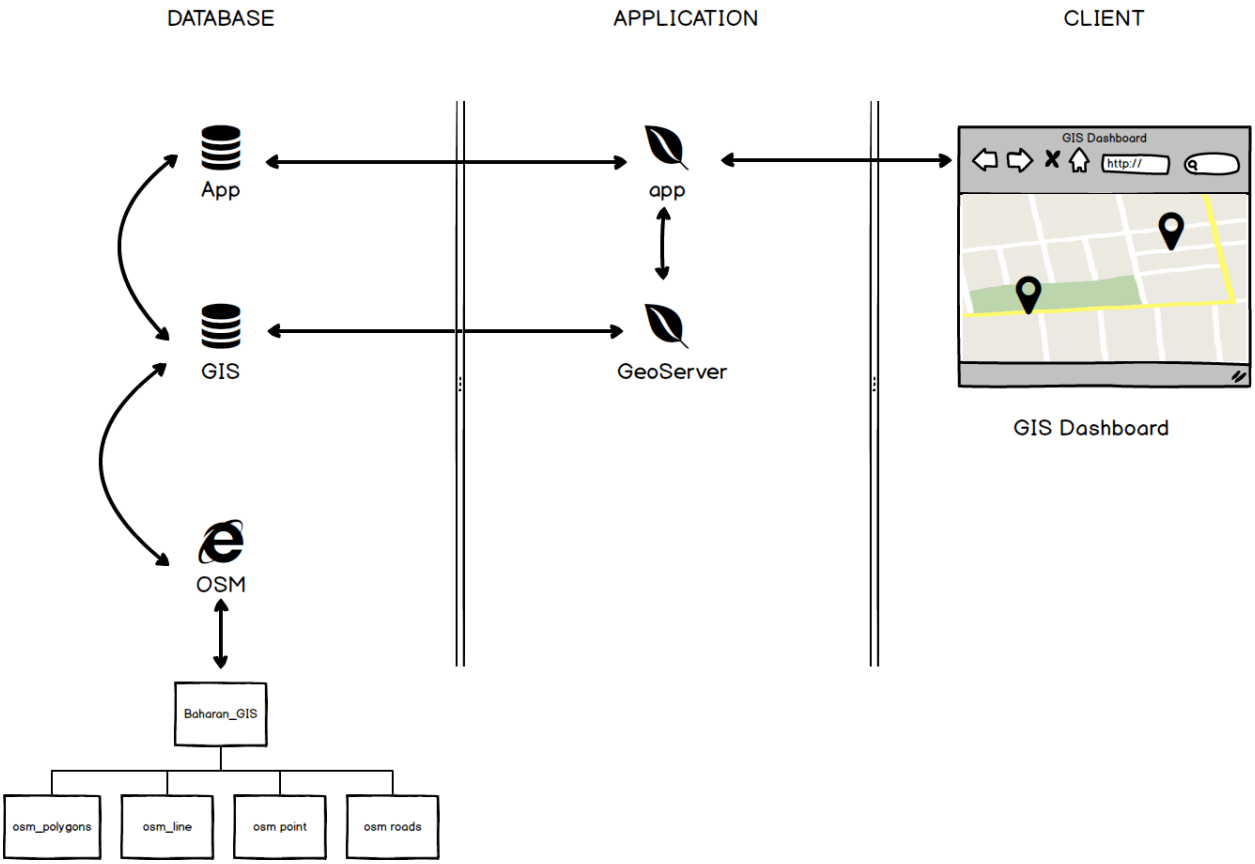


```
<FeatureTypeStyle>
  <Rule>
    <PolygonSymbolizer>
      <Fill>
        <CssParameter name="fill">#000080</CssParameter>
        <CssParameter name="fill-opacity">0.5</CssParameter>
      </Fill>
      <Stroke>
        <CssParameter name="stroke">#FFFFFF</CssParameter>
        <CssParameter name="stroke-width">2</CssParameter>
      </Stroke>
    </PolygonSymbolizer>
  </Rule>
</FeatureTypeStyle>
```

فصل ۲

معماری

معماری سامانه

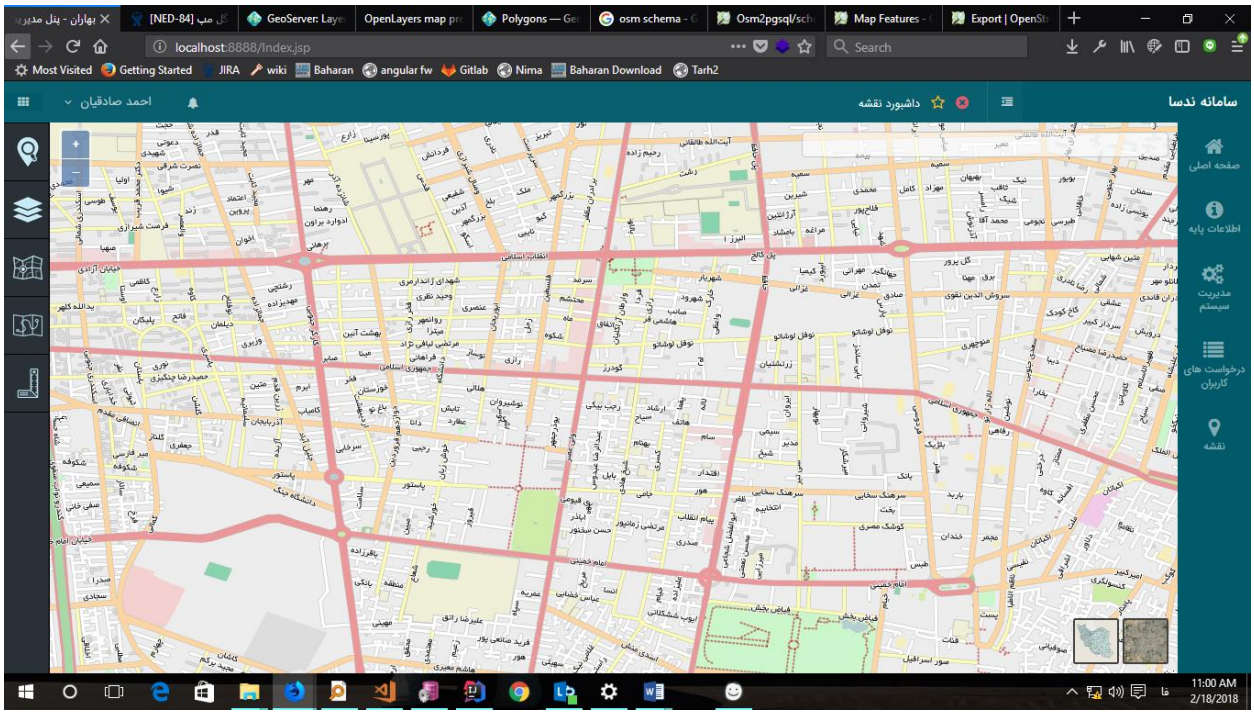


Openstreetmap

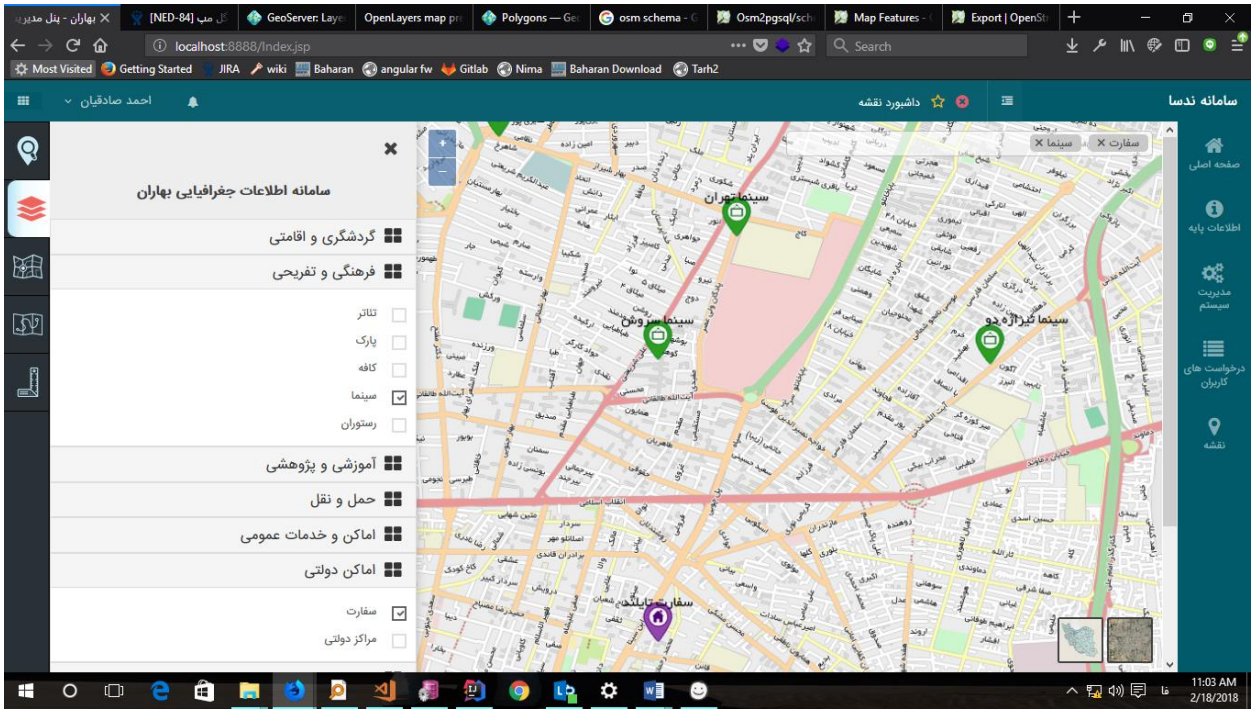
سایت osm که سایتی opensource هست اطلاعات جی آی اسی خودش رو در اختیار همه افراد قرار میده تا بتونن از اون استفاده کنند. شما میتوانید از انواع مختل دیتاهایی که در سایت موجود است استفاده کنید از shape فال ها گرفته تا فرمت pbf که برای تبدیل به دیتابیس mysql مورد استفاده قرار میگیرد.

دیتابیس

نقشه فعلی که در سامانه جی آی اس بهاران استفاده شده است از لایه های polygon,line,road میباشد که با استایل های خاصی به عنوان یک layergroup با هم ترکیب شده اند و به شکل زیر در آمده اند .



و همینطور د بخش لایه ها از جدول point استفاده کرده ایم که با کمک CQL هر کدام از مناطقی که مد نظر بوده را در قالب لایه ای خاص به نمایش میگذاریم.



دلیل استفاده از دیتابیس ، دسترسی بهتر و آزادی مانور در انجام پروژه های جی آی اسی بود است که باعث میشود در انجام نیاز های مختلف کارفرما با مشکلات کمتری مواجه شد . برای مثال فرض کنید بخواهیم بر اساس یک مختصات خاص(مثلا تقسیمات کشوری) جستجویی خاصی را انجام بدهیم که با شرایط فعلی سامانه و استفاده از اوراکل به عنوان دیتابیس جی آی اس به راحتی این کار با استفاده از oracle Spatial و داده های جئومتری (POINT,POLYGON,LINE) قابل انجام است و به کمک این قابلیت اوراکل قادر هستید داده های جغرافیای مربوط به بیزنس خود را در کنار دیگر فیلد های مدل خود ذخیره کنید . همچنین hibernate ۵ از oracle spatial پشتیبانی میکند و شما میتوانید به راحتی کوئری های جئومتری خود را با استفاده از هایبرنیت اجرا کنید.

فصل ۳

نحوه استفاده از جی آی اس

استفاده از داده های جئومتری در مدل ها

```java
News
package org.baharan.nedsa.model.news;

import com.vividsolutions.jts.geom.Point;
import org.baharan.framework.model.BaseEntity;
import org.baharan.framework.model.core.baseInfo.BaseInformation;

/**
 * Created by Nariman on 12/17/2017.
 */
public class News extends BaseEntity<Long> {
    private String title;
    private String description;
    private String summary;
    private String tag;
    private BaseInformation subject;
    private BaseInformation classification;
    private BaseInformation priority;
    private Point location;
    private String code;

    public String getTitle() { return title; }

    public void setTitle(String title) { this.title = title; }

    public String getDescription() { return description; }

    public void setDescription(String description) { this.description = description; }

    public String getSummary() { return summary; }

    public void setSummary(String summary) { this.summary = summary; }

    public BaseInformation getSubject() { return subject; }

    public void setSubject(BaseInformation subject) { this.subject = subject; }

    public BaseInformation getClassification() { return classification; }
```

```xml
nedsa - [G:\Nedsa\nedsa] - [nedsa] - ...\src\main\resources\hibernate\nedsa\news\News.hbm.xml - IntelliJ IDEA 2016.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
nedsa > src > main > resources > hibernate > nedsa > news > News.hbm.xml
NewsTrack.java | Track.hbm.xml | News.hbm.xml | Track.java | NewsController.java | NewsRepository.java | FillStyle.java | MapLocationController.java | MapLocationService.java

<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
        "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<!--@formatter:off-->
<hibernate-mapping>
    <class   name="org.baharan.nedsa.model.news.News"  table="Nedsa_News"  >
        <id name="id" type="long" >
            <column name="ID"  />
            <generator class="sequence" >
                <param name="sequence">SEQ_Nedsa_News</param>
            </generator>
        </id>
        <property          name="title"                    column="title"             type="string"                                                    not-null="false"
        <property          name="description"              column="description"       type="string"                                                    not-null="false"
        <property          name="summary"                  column="summary"           type="string"                                                    not-null="false"
        <property          name="tag"                      column="tag"               type="string"                                                    not-null="false"
        <property          name="code"                     column="code"              type="string"                                                    not-null="false"
        <many-to-one       name="subject"                  column="subject_Id"        entity-name="org.baharan.framework.model.core.baseInfo.BaseInformation" not-null="true"
        <many-to-one       name="classification"           column="classification_Id" entity-name="org.baharan.framework.model.core.baseInfo.BaseInformation" not-null="true"/
        <many-to-one       name="priority"                 column="priority_Id"       entity-name="org.baharan.framework.model.core.baseInfo.BaseInformation" not-null="false"
        <property          name="location"                 column="location"          type="org.hibernate.spatial.GeometryType"                        not-null="false"

        <many-to-one       name="createdBy"                column="CreatedBy"         entity-name="org.baharan.framework.model.security.User"          not-null="true"
        <many-to-one       name="updatedBy"                column="UpdatedBy"         entity-name="org.baharan.framework.model.security.User"          not-null="true"
        <property          name="createdDate"              column="CreatedDate"                                                                        not-null="true"
        <property          name="updatedDate"              column="UpdatedDate"                                                                        not-null="true"
        <property          name="ip"                       column="IP"                type="string"                                                    not-null="true"
    </class>
    <!--@formatter:on-->
</hibernate-mapping>

alter session set current_schema = GIS (today 11:33 AM)
```

چند نمونه از کوئری های جئومتری

```java
package org.baharan.nedsa.dao.impl.map;

import ...

@Repository
public class CountryDivisionRepository extends GenericRepository<CountryDivision, Long> implements ICountryDivisionRepo

    @Override
    protected Class<CountryDivision> getDomainClass() { return CountryDivision.class; }

    @Override
    public String getCode(Point point, BorderType borderType) {
        Session session = getSession();
        String hql = " select d.code " +
                " from CountryDivision d " +
                " where within(:point, d.geom) = true " +
                " and d.borderType = :borderType ";
        Query query = session.createQuery(hql);
        query.setParameter("point", point);
        query.setParameter("borderType", borderType);
        query.setMaxResults(1);
        return (String) query.uniqueResult();
    }
}
```

کوئری بالا کد تقسیمات کشوری مربوط به نقطه ای که در نقشه انتخاب شده باشد را بر میگرداند

```java
        query.setFirstResult(0);
        return (MapLocation) query.uniqueResult();
    }

    @Override
    public MapLocation getPointCity(Point point) {
        Session session = this.getSession();
        String hql =
                " from " + domainClass.getName() + " e " +
                        " where contains(e.geom, :point) = true and " +
                        " e.type = :locationType";

        Query query = session.createQuery(hql);
        query.setParameter("point", point);
        query.setParameter("locationType", LocationType.PROVINCE);
        query.setMaxResults(1);
        query.setFirstResult(0);
        return (MapLocation) query.uniqueResult();
    }
```

کوئری بالا مشخص میکند نقطه انتخابی مربوط به کدام شهر است.

جدول function های hibernate spatial

Function
Basic functions on Geometry
int dimension(Geometry)
String geometrytype(Geometry)
int srid(Geometry)
Geometry envelope(Geometry)
String astext(Geometry)
byte[] asbinary(Geometry)
boolean isempty(Geometry)
boolean issimple(Geometry)
Geometry boundary(Geometry)
Functions for testing Spatial Relations between geometric objects
boolean equals(Geometry, Geometry)
boolean disjoint(Geometry, Geometry)
boolean intersects(Geometry, Geometry)
boolean touches(Geometry, Geometry)
boolean crosses(Geometry, Geometry)
boolean within(Geometry, Geometry)
boolean contains(Geometry, Geometry)
boolean overlaps(Geometry, Geometry)
boolean relate(Geometry, Geometry, String)
Functions that support Spatial Analysis
double distance(Geometry, Geometry)
Geometry buffer(Geometry, double)
Geometry convexhull(Geometry)
Geometry intersection(Geometry, Geometry)
Geometry geomunion(Geometry, Geometry)
Geometry difference(Geometry, Geometry)
Geometry symdifference(Geometry, Geometry)
Common non-SFS functions

boolean dwithin(Geometry, Geometry, double)
Geometry transform(Geometry, int)
Spatial aggregate Functions
Geometry extent(Geometry)

معرفی نوع داده Geometry :

این نوع داده شامل ۵ پارامتر ، شامل موارد زیر میباشد.

- **SDO_GTYPE**

۴ رقم با فرمت DLTT که نمایانگر نوع جئومتری میباشد

  - D : Dimention (مقادیر ۲ و ۳ و ۴ را میگیرد مثلا ۲ بعدی و...)
  - L: Linear Refrencing System (مقدار آن اکثرا ۰ است)
  - T : Type Of Geometry (۰۱ نقطه است ، ۰۲ خط یا کمان است ، ۰۳ چند ضلعی یا سطح)

به عنوان مثال ۲۰۰۲ به معنی یک خط یا کمان دو بعدی است

- **SDO_SRID**

سیستم مختصات جغرافیایی بین المللی (معمولا از استاندارد ۴۳۲۶ استفاده میشود)

- **SDO_POINT**

در صورت که نوع جئومتری شما یک نقطه باشد این مقدار مخالف null میباشد برای مثال {۵۱,۳۵،۳۵,۷۲,null} که مقدار آخر(z) که بعد سوم محسوب میشود null است.

- **SDO_ELEM_INFO**

توضیحی است برای پارامتر بعدی (SDO_ORDINATES) شامل سه قسمت میباشد

  - **SDO_STARTING_OFFSET**

    نشان دهنده آفست شروع آرایه Ordinate میباشد

  - **SDO_ETYPE**

    نمایانگر شکل (shape) ordinate میباشد برای مثال

    - ۲: خط صاف
    - ۱۰۰۳ : چند ضلعی (polygon)

  - **SDO_INTERPOLATION**

این پارامتر دو استفاده دارد

§ برای آبجکت های ساده مانند چند ضلعی ، نوع چند ضلعی را مشخص میکند. برای مثال

- ۱ : خط
- ۲ : کمان (arc)
- ۳ : مستطیل (rectangle)
- ۴ : دایره (circle)

§ برای آبجکت های پیچیده ، تعداد عضو های آرایه ordinates را مشخص میکند برای مثال چند ضلعی را در نظر بگیرید که بیش از یک خم بسته دارد.

- **SDO_ORDINATE**

آرایه ای از مختصات ها برای آبجکت های غیر از point . در واقع جئوسرور با خواند اطلاعات این فیلد آبجکت را رسم میکند و متا دیتای آبجک هم طبق توضیحات بالا در SDO_ELEM_INFO ذخیره شده است.

نمونه داده های جئومتری ذخیره شده در دیتابیس

POINT

{۲۰۰۱،۴۳۲۶,{۴۴,۹۶۸۷۰۱۰۲۷۳۵۴،۳۷,۲۱۰۵۹۱۵۸۱۶۴۶۱,null},null,null}

LINE

{۲۰۰۲،۴۳۲۶,null,{۱،۲،۱},{۵۱,۴۷۳۸۸۶۸۲۰۴۲۲۲،۲۸,۰۶۱۴۴۶۶۶۹۷۰۲۳،۵۱,۴۷۳۸۴۰۱۰۸۰۲۷۵،۲۸,۰
۶۱۹۷۱۹۹۹۰۴۱۹،۵۱,۴۷۳۷۲۳۴۱۶۸۷۲۱،۲۸,۰۶۵۱۲۳۷۶۲۶۱۹۷,...,۵۱,۴۷۳۳۳۰۵۸۳۵۹۸۳،۲۸,۰۷۶۴۹۱
۴۵۹۶۷۵۷}}

POLYGON

{۲۰۰۳،۴۳۲۶,null,{۱،۱۰۰۳،۱},{۵۵,۳۸۶۴۵۱۹۶۸۶۸۳۹،۳۳,۳۶۴۹۲۳۶۲۵۸۷۵۱،۵۵,۳۸۸۱۱۸۷۰۲۸۶۲۱،
۵۵,۸۵۱۱۹۵۴۷۰۷۵۴۷،۳۲,۷۶۲۶۸۳۳۰۳۶۸۱۶,...}}

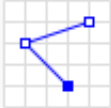
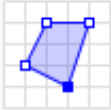
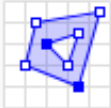
WKT(Well Known Text)

استانداردی برای نمایش داده های وکتور جئومتری بر روی نقشه میباشد . در این سامانه از openLayers library برای ایجاد ارتباط با geoserver و کار کردن با نقشه استفاده شده است . openlayers توابعی را در اختیار ما قرار میدهد تا به راحتی داده های geometry که از سمت دیتابیس می آید به فرمت wkt تبدیل کرده و در نقشه نمایش دهیم و همچنین داده داده های جئومتری از جنس wkt که در طول استفاده از نرم افزار از آن ها استفاده میکنید را به فرمت مورد نظر برای ذخیره سازی در دیتابیس تبدیل میکند.

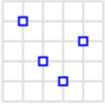
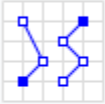
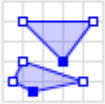
علت اصلی استفاده از wkt همزبانی بین لایه مختلف سیستم است . در واقع ما برای نمایش و ارتباط با نقشه ها باید در لایه های مختلف با زبان های مختلفی صحبت کنیم ، در front با wkt در backend با هایبرنیت و در دیتابیس با oracle geometry که برای ایجاد ارتباط بین این زبان ها از wkt استفاده میکنیم چون زبانی است که هایبرنیت از آن پشتیبانی میکند.

نمونه مختصات ها در استاندارد wkt

Geometry primitives (2D)

Type	Examples	
Point		POINT (30 10)
LineString		LINESTRING (30 10, 10 30, 40 40)
Polygon		POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))
		POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))

Multipart geometries (2D)

Type	Examples	
MultiPoint		MULTIPOINT ((10 40), (40 30), (20 20), (30 10))
		MULTIPOINT (10 40, 40 30, 20 20, 30 10)
MultiLineString		MULTILINESTRING ((10 10, 20 20, 10 40), (40 40, 30 30, 40 20, 30 10))
MultiPolygon		MULTIPOLYGON (((30 20, 45 40, 10 40, 30 20)), ((15 5, 40 10, 10 20, 5 10, 15 5)))
		MULTIPOLYGON (((40 40, 20 45, 45 30, 40 40)), ((20 35, 10 30, 10 10, 30 5, 45 20, 20 35), (30 20, 20 15, 20 25, 30 20)))

کلاس WktUtility

برای سهولت تبدیل نوع geometry و wkt ، کلاس wktUtility به فریم ورک اضافه شده است

نمونه

```
44        }
45
46        @Override
47        public MapLocation getNearestRoute(String pointWkt) {
48            return this.getNearestRoute(WktUtility.wktToGeometry(pointWkt, Point.class));
49        }
50
```

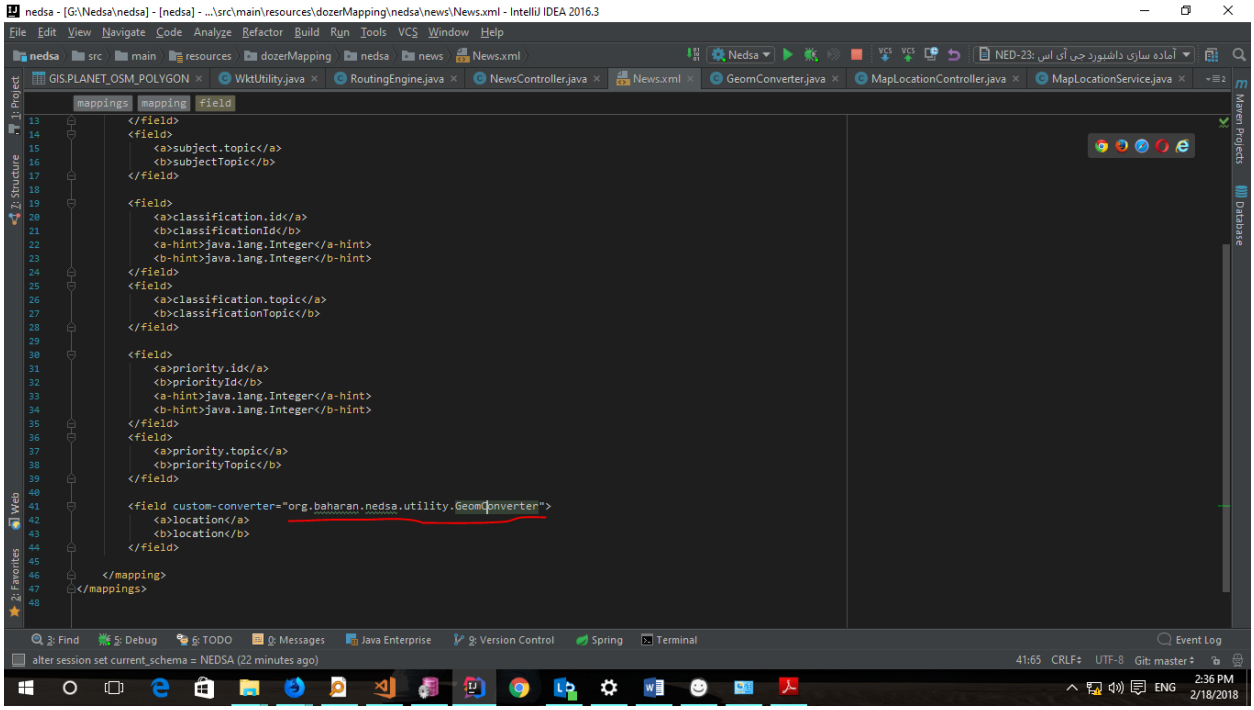
در مثال بالا نام نزدیک ترین منطقه ی نزدیک به نقطه انتخاب شده در textbox مربوط به نقطه شروع یا پایان مسیر یابی درج میشود. همان طور که مشاهده میکنید نقطه انتخاب شده از سمت کلاینت که به صورت wkt ارسال شده است با استفاده از کلاس معرفی شده به نوع داده جئومتری تبدیل شد.

نمونه

```
34
35        @GetMapping(value = "/powerLocation")
36        public String powerLocation() {
37            return WktUtility.geometryToWkt(service.powerLocation());
38        }
39
40    }
41
```

نمونه کد بالا bound (حاشیه دور) استانی که کاربر مربوطه از آنجا لاگین کرده است را با فرمت wkt بر میگرداند تا به هنگام لود نقشه به صورت اتوماتیک ، استان مورد نظری که کاربر متعلق به آنجا هست لود شود.

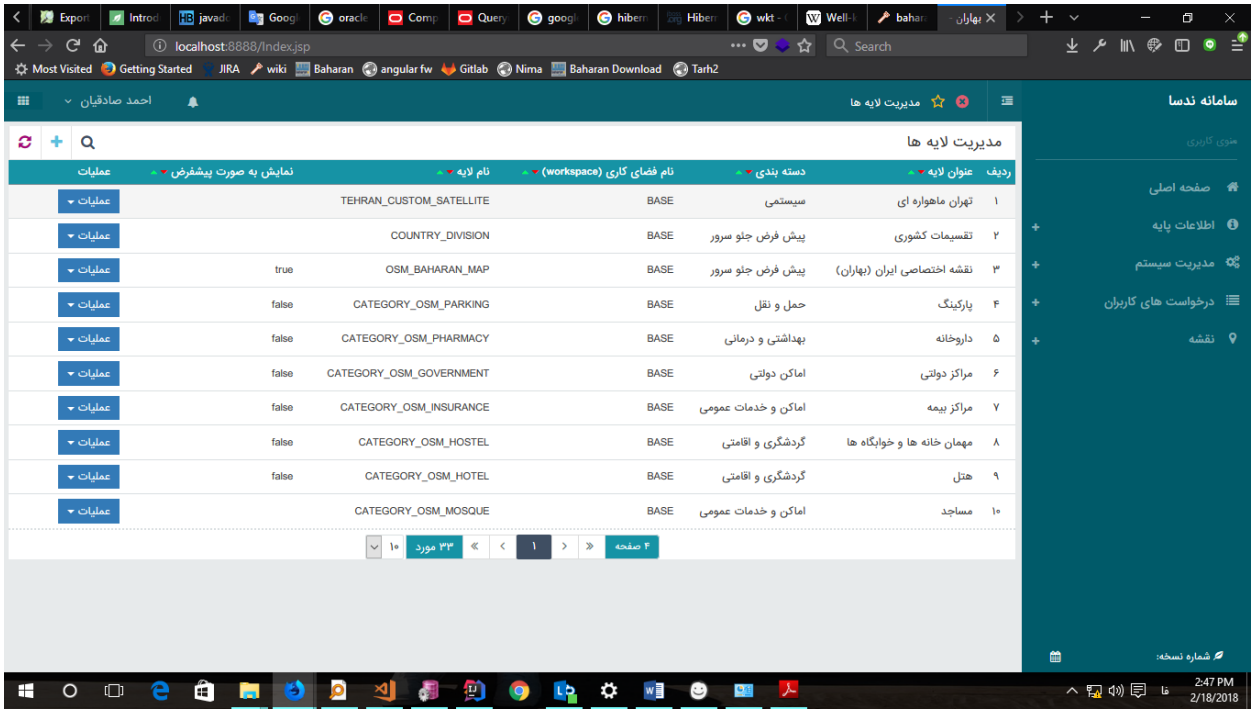
نمونه ای از فایل xml مدل جئومتری

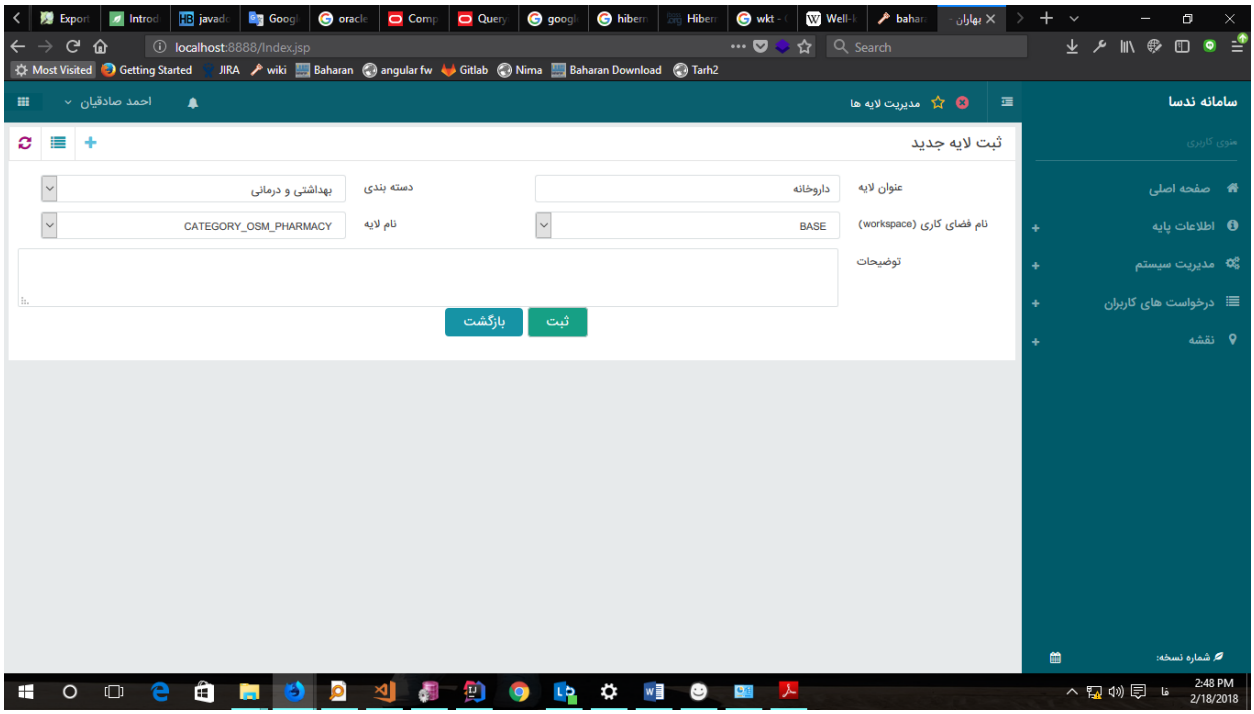


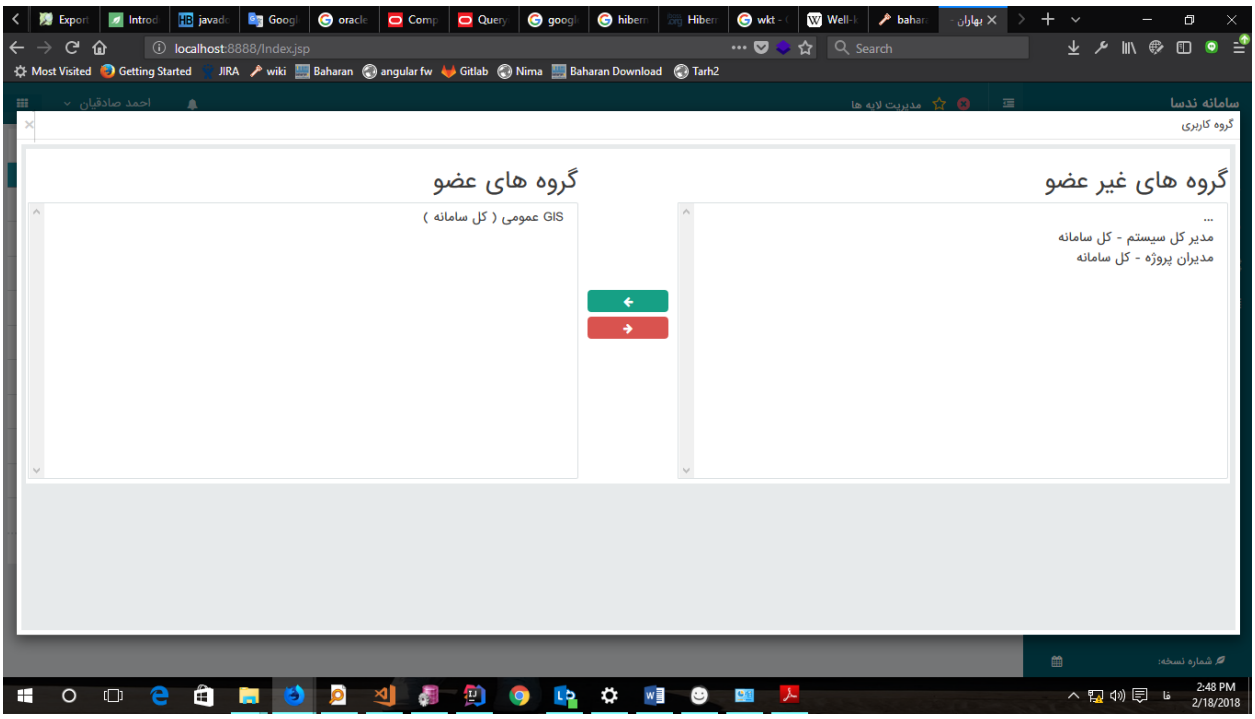
برای تبدیل wkt و geometry توسط modelMapper ها ، همان طور که در بالا مشاهده میکنید از کلاس GeomConverter استفاده میکنیم که در فریم ورک اضافه شده است.

مدیریت لایه ها

برای مدیریت لایه هایی که در جئوسرور اضافه میکنیم و دسترسی های مربوط به آن ها روندی را آماده کردیم که اطلاعات کلیه لایه ها از دیتابیس خوانده شود و ارتباط زیادی بین اپلیکیشن و جئوسروربه غیر از نمایش نقشه نباشد.در واقع با این رویکرد از تعریف کردن کاربر های اضافه در جئوسرور جلوگیری کردیم و مدیریت دسترسی را درون اپلیکیشن و نسبت به یوزر های سیستم مدیریت کرده ایم.


localhost:8888/Index.jsp
سامانه ندسا
مدیریت لایه ها
صفحه اصلی
اطلاعات پایه
مدیریت سیستم
درخواست های کاربران
نقشه
مدیریت لایه ها
ردیف
عنوان لایه
دسته بندی
نام فضای کاری (workspace)
نام لایه
نمایش به صورت پیشفرض
عملیات
۱ تهران ماهواره ای سیستمی BASE TEHRAN_CUSTOM_SATELLITE
۲ تقسیمات کشوری پیش فرض جئو سرور BASE COUNTRY_DIVISION
۳ نقشه اختصاصی ایران (بهاران) پیش فرض جئو سرور BASE OSM_BAHARAN_MAP true
۴ پارکینگ حمل و نقل BASE CATEGORY_OSM_PARKING false
۵ داروخانه بهداشتی و درمانی BASE CATEGORY_OSM_PHARMACY false
۶ مراکز دولتی اماکن دولتی BASE CATEGORY_OSM_GOVERNMENT false
۷ مراکز بیمه اماکن و خدمات عمومی BASE CATEGORY_OSM_INSURANCE false
۸ مهمان خانه ها و خوابگاه ها گردشگری و اقامتی BASE CATEGORY_OSM_HOSTEL false
۹ هتل گردشگری و اقامتی BASE CATEGORY_OSM_HOTEL false
۱۰ مساجد اماکن و خدمات عمومی BASE CATEGORY_OSM_MOSQUE
۴ صفحه
۳۳ مورد
۱۰
شماره نسخه:
2:47 PM
2/18/2018


localhost:8888/Index.jsp
سامانه ندسا
ثبت لایه جدید
عنوان لایه
داروخانه
دسته بندی
بهداشتی و درمانی
نام فضای کاری (workspace)
BASE
نام لایه
CATEGORY_OSM_PHARMACY
توضیحات
ثبت
بازگشت
شماره نسخه:
2:48 PM
2/18/2018



هر لایه ای که در geoserver تعریف شده باشد برای نمایش داده شدن باید در بخش مدیریت لایه ها تعریف شود و سپس از طریق گزینه عملیات میتوانیم گروه هایی که میبایست این لایه ها را مشاهده کنند را تخصیص میدهیم ؛ اگر کاربر وارد شده در داشبورد به لایه مورد نظر دسترسی داشته باشد میتواند از منو لایه آن را روی نقشه نمایش دهد.

تقسیمات کشوری

برای ایجاد هماهنگی و یکسان سازی در سیستم جغرافیایی از کدینگ تقسیمات کشوری ارائه شده وزارت کشور استفاده است تا با استفاده از یکتایی کد ها در کوئری های جغرافیای عملکرد سریع تری را داشته باشیم.

به عنوان مثال یکی از نیاز های مشترک مطرح شده در سامانه ها نمایش اطلاعات (استایل)به صورت متفاوت در سطوح بزرگنمایی مختلف بود . برای نمونه کاربر این نیاز را مطرح میکرد که اگر از zoom level بالا به نقشه نگاه میکند (سطحی که استان ها یا شهر را نشان میدهد) به جای نمایش پراکندگی تمامی ماشین ها تنها کافیست رقم تعداد ماشین های درون آن شهر نمایش داده شود و اگر درون استان یا شهر zoom کردیم پراکندگی نقطه ای ماشین ها در سطح شهر و یا استان را نمایش دهد.

```java
55      @Override
56      public List<NewsViewModel> listByBound(String polygon, Integer zoom) {
57          List<NewsViewModel> newsViewModels = null;
58          if (zoom <= 10) {
59              newsViewModels = iNewsRepository.groupByBound(BorderType.CITY);
60          } else {
61              newsViewModels = ModelMapper.maplist(iNewsRepository.listByBound(WktUtility.wktToGeometry(polygon, Polygon.class)), NewsViewModel.class);
62          }
63          newsViewModels.stream().forEach(n -> this.setNewsStyle(n, zoom));
64          return newsViewModels;
65      }
66
```

```java
33
34          @Override
35          public List<News> listByBound(Polygon polygon) {
36              String hql = " from " + domainClass.getName() + " e " +
37                      " where within(e.location , :polygon) = true ";
38              HashMap<String, Object> params = new HashMap<>();
39              params.put("polygon", polygon);
40              return super.getAll(hql, params);
41          }
42
43
44          @Override
45          public List<NewsViewModel> groupByBound(BorderType borderType) {
46              Session session = getSession();
47              String hql = " select d.code as code, cast(count(e) as string) as title" +
48                      " from News e, CountryDivision d " +
49                      " where d.code = e.code " +
50                      " and d.borderType = :borderType " +
51                      " group by d.code";
52              Query query = session.createQuery(hql);
53              query.setParameter("borderType", borderType);
54              query.setResultTransformer(Transformers.aliasToBean(NewsViewModel.class));
55              return query.list();
56          }
57
58      }
59
```

همان طور که مشاهده میکنید در نمونه کد بالا با استفاده از کد تقسیمات کشوری که به هنگام ذخیره هر آبجکت ، در جدول مربوطه ذخیره شده است ، نیاز مطرح شده را حل کرده ایم و همچنین برای کاربر این امکان وجود دارد تا با استفاده از کلاس Style که به فریم ورک اضافه شده است ، نحوه نمایش بر روی نقشه را هم با فرمت خاصی به ما بدهد تا در نقشه به نمایش بگذاریم که در ادامه به بررسی آن خواهیم پرداخت.

جدول COUNTRY_DIVISION شامل اطلاعات کدینگ و مرز جغرافیای مربوط به تقسیمات کشوری ایران در سطح شهر و استان میباشد.فیلد border_type نشانگر شهر و یا استان بودن این رکورد هاست (۰ به معنی استان و ۱به معنی شهرستان ها)

کلاس Style

برای نمایش آبجکت ها بر روی نقشه کلاسی به نام Style ساخته شده است که به شما این امکان را میدهد تا در لایه سرویس خود مشخص کنید که میخواهید آبجکت شما بر روی نقشه به چه شکلی دیده شود . شما میتوانید برای برای نمایش یک نقطه در شرایط مختلف بیزنسی آیکن های متفاوتی را با اندازه های مختلفی نمایش دهید یا برای نمایش یک منطقه مشخص کنید که رنگ دور خط منطقه چه رنگی باشد و داخل منطقه انتخاب شده با چه رنگی پر شود.

نمونه

```
63          newsViewModels.stream().forEach(n -> this.setNewsStyle(n, zoom));
64          return newsViewModels;
```

```
75      private void setNewsStyle(NewsViewModel news, int zoom) {
76          Style style = null;
77          String iconClassified = "/View/gis/map/images/megaphone(1).png";
78          String iconUnclassified = "/View/gis/map/images/megaphone.png";
79          if (zoom == -1 || zoom > 10) {
80              if (news.getClassificationId() != null && news.getClassificationId().equals(803)) {
81                  style = new Style(new IconStyle(iconClassified,  scale: 1F));
82              } else {
83                  style = new Style(new IconStyle(iconUnclassified,  scale: 1F));
84              }
85          } else {
86              style = new Style(new TextStyle(
87                      news.getTitle(),  fillColor: "blue",  strokeColor: "white",  scale: 3,  textAlign: "center"));
88          }
89
90          ObjectMapper mapper = new ObjectMapper();
91          try {
92              String json = mapper.writeValueAsString(style);
93              news.setStyle(json);
94          } catch (JsonProcessingException e) {
95              e.printStackTrace();
96          }
97      }
98  }
99
```

به عنوان مثال در بالا همان طور که مشاهده میکنید گفت شده است که اگر zoom leve از ۱۰ بزرگتر بود و طبقه بندی عادی بود یک آیکن رو نمایش دهد و اگر غیر عادی بود آیکن دیگری با رنگ قرمز نمایش دهد. همین طور در ادامه اگر zoom level از ۱۰ کوچکتر بود از textStyle برای نمایش جمع کل آبجکت ها استفاده کند که به آن رنگ و متن و مقیاس نمایش را ارسال کرده است.

انواع سازنده های کلاس Style

- public Style(StrokeStyle stroke, TextStyle text){}
- public Style(TextStyle text){}
- public Style(IconStyle icon){}
- public Style(StrokeStyle stroke, FillStyle fill, TextStyle text){}
- public Style(StrokeStyle stroke, FillStyle fill, IconStyle icon, TextStyle text){}

انواع کلاس های استفاده شده مرتبط با کلاس Style

StrokeStyle

کلاس دور خط برای چند ضلعی ها و اشکال هندسی دارای محیط کاربرد دارد

Constructors

- public StrokeStyle(String color, String width){}

- public StrokeStyle(String color, String lineCap, String lineDash, String lineDashOffset, String width, String lineJoin){}

Properties

- **color** : هم میتوانیم نام رنگ مورد نظر را اینجا بنویسیم مثل (red,green,...) و هم میتوانیم از کد hex استفاده کنیم مانند #fff

- **width** : اندازه خط را مشخص میکند که با عدد مشخص میشود

- **lineCap** : انتهای خطوط را به صورت round و یا اشکال دیگر در می آورد ، مقادیر : round,square,butt

- **lineDash** : آرایه ای از اعداد برای فاصله بین خطوط منقطع عدد اول نمایانگر طول خط و عدد دوم فاصله بین دو خط [۵،۱۰]
- **lineDashOffset**
- **lineJoin**: نقاط اتصال خطوط را به اشکال خاصی تبدیل میکند ، مقادیر : `bevel, round, miter`

TextStyle

برای نمایش متنی یا عددی مورد استفاده قرار میگیرد

Constructors

- public TextStyle(String content)
- public TextStyle(String content, String fillColor, Integer scale)
- public TextStyle(String content, String fillColor, String strokeColor, Integer scale, String textAlign)
- public TextStyle(String content, String font, String fillColor, String strokeColor, Integer scale, String textAlign, String backgroundFill, String backgroundStroke)

Properties

- **scale**: مقدار عددی از ۱ به بالا که اندازه متن را مشخص میکند.
- **textAlign**: محل قرار گیری متن را مشخص میکند ، مقادیر : 'left', 'right', 'center',

IconStyle

برای نمایش آیکن های خاص با فرمت png مورد استفاده قرار میگیرد.

Constructors

- public IconStyle(String src)
- public IconStyle(String src, Float scale)

Properties

- **src** : آدرس محل ذخیره شدن عکس
- **Scale** : عددی اعشاره بین ۰ تا ۱ برای مثال ۰,۵ یا ۱

FillStyle

برای پر کردن رنگ داخل یک چند ضلعی و یا شکل هندسی بسته به کار می رود.

Constructors

- public FillStyle(String color)

نحوه ارتباط سامانه ها با Baharan_GIS

برای این کار کافی است با زبانی که در ادامه به شما معرفی میکنیم با ما ارتباط برقرار کنید تا اطلاعات مدل های شما که شامل فیلد های جئومتری هستند را بر روی نقشه ببینید و بر روی آن ها عملیات های موردنظرتان را انجام دهید.

Map-config.js

برای نمایش اطلاعات خود کافی است اطلاعات این فایل را با استانداردی که برای شما مشخص کردیم پر کنید تا Baharan_GIS با استفاده از آن اطلاعات را خوانده و پردازش کند.

```
10
11  var mapConfig = {
12      defaultCenterByPower: true, //optional
13      defaultLayers: ['OSM_BAHARAN_MAP'], //optional
14      defaultBounds: [43.3699439472087, 25.0592443984591, 63.4014839521755, 39.7816502038349], //optional
15      objects: [{
16          name: 'news',
17          title: 'اخبار',
18          url: {
19              list: '/rest/nedsa/news/getAll',
20              search: '/rest/nedsa/news/map/grid',
21              load: '/rest/nedsa/news/load',
22              listByBound: '/rest/nedsa/news/listByBound'
23          },
24          icon: 'fa fa-newspaper-o',
25          tooltip: {
26              title: 'title',
27              detail: 'description'
28          },
29          schema: [{
30              key: 'title',
31              type: 'string',
32              title: 'عنوان خبر',
33              search: true, //optional, default = false
34              view: true //optional, default = false
35          }, {
36              key: 'description',
37              type: 'string',
38              title: 'شرح خبر',
39              search: true
40          }, {
41              key: 'summary',
42              type: 'string',
43              title: 'خلاصه'
44          }, {
45              key: 'location',
46              type: 'geom',
47              title: 'موقعیت جغرافیایی'
48          }, {
```

```
46                  type: 'geom',
47                  title: 'موقعیت جغرافیایی'
48              }, {
49                  key: 'code',
50                  title: 'کد تقسیمات'
51              }],
52              commands: [{
53                  title: 'نمایش جزییات',
54                  icon: '/View/gis/map/images/details.png',
55                  url: '/View/nedsa/news/NewsInfo.jsp'
56              }, {
57                  title: 'دستورات',
58                  icon: '/View/gis/map/images/path.png',
59                  url: '/View/nedsa/order/Index.jsp'//required if you haven't trackUrl
60              },{
61                  title: 'مسیریابی',
62                  icon: '/View/gis/map/images/track.png',
63                  trackUrl: '/rest/nedsa/news/NewsTrack/listByDate'//required if you haven't url
64              }]
65          }, {
66              name: 'cars',
67              title: 'خودرو',
68              url: {
69                  list: '/rest/nedsa/news/getAll',
70                  search: '/rest/nedsa/news/list/grid',
71                  load: '/rest/nedsa/news/load',
72                  listByBound: '/rest/nedsa/news/listByBound'
73              },
74              icon: 'fa fa-car',
75              tooltip: 'description',
76              schema: [{
77                  key: 'title',
78                  type: 'string',
79                  title: 'عنوان خبر',
80                  search: true, //optional, default = false
81                  view: true //optional, default = false
82              }, {
83                  key: 'description',
84                  type: 'string',
```

defaultCenterByPower

این فیلد از نوع Boolean میباشد . اگر مقدار را برابر true قرار دهید به هنگام لود شدن نقشه ، چک میشود که رده کاربری که لاگین کرده است در کدام استان قرار دارد و با توجه به آن zoom scale نسبت به استان رده نمایش داده میشود

defaultCenterPoint

اگر بخواهید از نقطه ای خاص نقشه شما لود شود به صورت زیر این مشخصه را پر میکنید

defaultCenterPoint : Array [٥١,٤٢٦١٩٧٨٥٢٦٠٧١٦, ٣٥,٧٠٢٠٢٦٩٣٦٦٥٠١١]

defaultLayers

آرایه ای از نام آرایه ها که در مدیریت لایه ها قرار دارد را میتواند به عنوان ورودی بگیرد. اگر نام لایه خاصی را بخواهید بلافاصله در هنگام لود شدن نقشه نمایش دهید باید نام آن را در این قسمت واردن کنید.

defaultBounds

یک آرایه ۴ تایی میباشد که اگر مقدار defaultCenterByPower را false بگذاریم نقشه را از این مختصات نمایش خواهد داد.

objects

آرایه ای از آبجکت هایی که میخواهید در بخش دسته بندی در داشبورد نقشه نمایش داده شود را در اینجا باید تعریف کرد

name

نام آبجکت مورد نظ را به انگلیسی در این قسمت درج کنید

title

نام مدل خود را به زبان فارسی برای نمایش در بخش دسته بندی (داشبورد نقشه) وارد کنید.

localhost:8888/View/gis
Lines — GeoServer
OpenLayers v4.6.4
localhost:8888/View/gis/map/Dashboard.jsp
Most Visited
Getting Started
JIRA
wiki
Baharan
angular
سامانه اطلاعات جغرافیایی بهاران
اخبار
خودرو

url

در این قسمت آدرس rest های مورد نیاز برای لود کردن دیتا های خود را برای ما مینویسید

list: آدرس **getAll/**

Search: آدرس **/list/grid**

load : آدرس**/load/{id}**

listByBound: آدرس **/listByBound/{polygon}/{zoom}**

متد listByBound متدی است که آبجکت های شما رو نسبت به bound ای که در صفحه مشاهده میکنید(یعنی اندازه ای از نقشه که درون مانیتور قابل مشاهده است) لود میکند و با هر زوم و یا درگ کردن یک rest به سمت سرور رفته و داده های همان ناحیه لود میشود که در قسمت تقسیمات کشوری کد مربوط به آن مورد بررسی قرار گرفت.

icon

این قسمت را با یکی از فونت های fontAwesome پر کنید تا در کنار نام دسته بندی نمایش داده شود

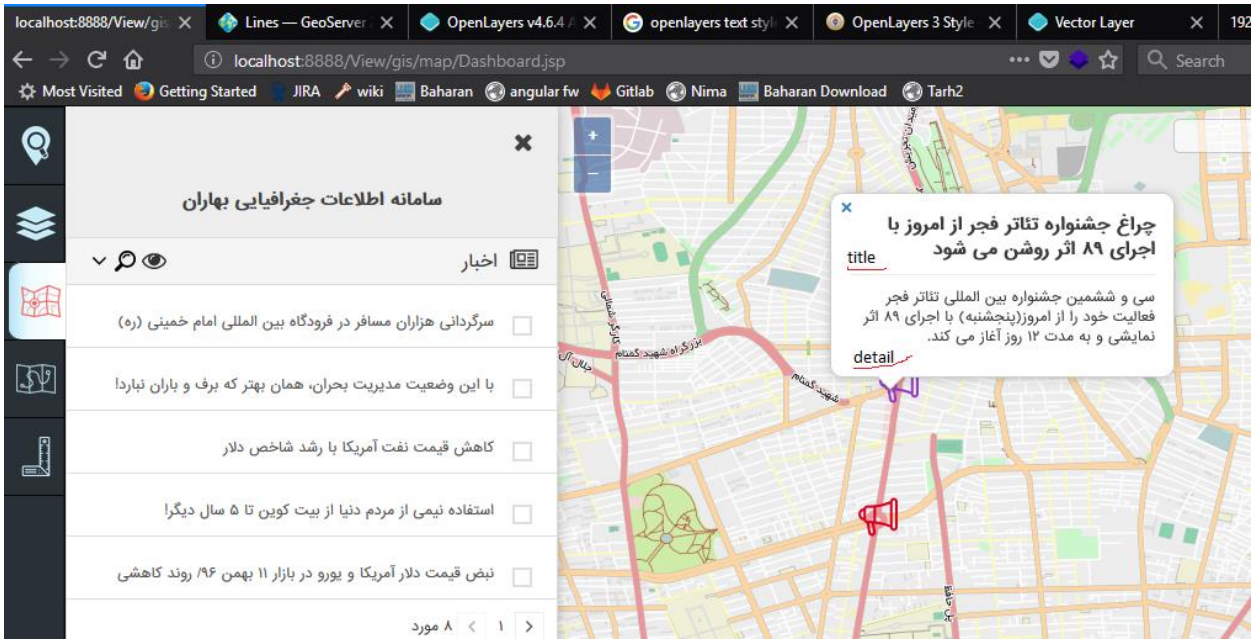
localhost:8888/View/gis
Lines — GeoServer
OpenLayers v4.6.4
localhost:8888/View/gis/map/Dashboard.jsp
Most Visited
Getting Started
JIRA
wiki
Baharan
angular
سامانه اطلاعات جغرافیایی بهاران
اخبار
خودرو

tooltip

این قسمت شامل دو property میباشد . به هنگامی نمایش آبجکت ها بر روی نقشه ، اصولا وقتی با اشاره گر بر روی اشکال مربوطه بر روی نقشه قرار میگیریم دوست داریم اطلاعات مختصری راجع به آن آبجکت را به کاربر نمایش دهیم به همین منظور در این قسمت باید مشخص کنیم که کدام فیلد از مدل به عنوان title و کدام فیلد به عنوان detail نمایش داده شود.

title : نام فیلدی از مدل اصلی (هم نام با property مربوطه در کلاس جاوایی) که میخواهیم به عنوان title نمایش داده شود.

detail : نام فیلدی از مدل اصلی (هم نام با property مربوطه در کلاس جاوایی) که میخواهیم به عنوان detail نمایش داده شود.



schema

در این قسمت باید نام فیلد هایی از مدل خود که در قسمت نقشه به آن نیاز دارید و در rest هایی که آدرسشان را در بالا دادیم دخیل هستند را با فرمتی مطابق نمونه برای ما وارد کنید .

این مشخصه میتواند آرایه ای از آبجکت های شما را شامل شود (قابل مشاهده در نمونه بالا)

key : نام فیلد به انگلیسی

type : جنس (type) فیلد به انگلیسی برای مثال string,geom,number

title : متن فارسی فیلد برای نمایش جزئیات

search : این فیلد از نوع Boolean میباشد. اگر مقدار آن true باشد یعنی مخواهید از این فیلد به عنوان فیلد قابل جستجو استفاده کنید و لیست خود را محدود کنید

لازم به ذکر است این فیلد optional است و مقدار پیش فرض آن false میباشد

نکته : در حال حاضر تنها فیلد های از نوع string قابل جستجو میباشند.

view : اگر میخواهید این فیلد در بخش دسته بندی نمایش داده شود باید این مقدار را برابر true قرار دهید. لازم به ذکر است که این مشخصه optional میباشد و مقدار پیش فرض این فیلد false میباشد

commands

قطعا یکی از feature هایی که بر روی آیجکت های روی نقشه مدنظر تمامی کاربران است امکان کلیک راست است. قطعا شما با استفاده از این command ها میخواهید عملیات مختلفی را انجام دهید که در ادامه به بررسی آن میپردازیم.

در این مشخصه شما میتوانید آرایه ای از فعالیت (action) های خود را وارد کنید.

title : در این قسمت متنی که میخواهید به هنگام کلیک راست بر روی نقشه نمایش داده شود را پر میکنید.

icon : در این قسمت آدرس فایل png که مخواهید در کنار متن نمایش داده شود را وارد میکنید.

url : آدرس صفحه jsp که میخواهید پس از فشردن کلید به شما نمایش داده شود را در این قسمت وارد میکنید

نکته : به هنگام فشردن دکمه ما id شئ مورد نظر را برای شما ارسال میکنیم تا با استفاده از آن عملیات خود را انجام دهید و ما صفحه را برای شما لود کنیم.

trackUrl : یکی از امکانات مورد نظر کاربران نمایش اطلاعات tracking آبجکت مورد نظر بر روی نقشه میباشد.برای این منظور دیگر نیازی به پر کردن مشخصه url نیست و به جای آن در این فیلد آدرس rest مربوط به لود کردن دیتای tracking را وارد میکنیم.

نکته : از دو فیلد url و trackUrl تنها یکی از آن ها میتواند در آبجکت command وجود داشته باشد .

Tracking

برای انجام عملیات مسیر یابی باید جداول خاصی تولید شود که در ادامه به بررسی آن خواهیم پرداخت

اگر میخواهید اطلاعات ردگیری آبجکت خاصی را بر روی نقشه نمایش دهید لازم است که مدلی به شکل زیر تولید کنید.

NewsTrack.java × | Track.java × | NewsController.java × | NewsRepository.java × | style-pars

```
1   package org.baharan.nedsa.model.news;
2
3   import ...
5
6   public class NewsTrack extends Track {
7       private News news;
8       private Point location;
9
10      public News getNews() { return news; }
13
14      public void setNews(News news) { this.news = news; }
17
18      @Override
19      public Point getLocation() { return location; }
22
23      @Override
24      public void setLocation(Point location) { this.location = location; }
27  }
28
```

```java
package org.baharan.nedsa.model.map;

import ...

public class Track extends BaseEntity<Long> {
    private Point location;

    public Point getLocation() { return location; }

    public void setLocation(Point location) { this.location = location; }
}
```

NewsTrack.hbm

```xml
<?xml version="1.0"?>
<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
                                   "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd">
<hibernate-mapping>
    <class name="org.baharan.nedsa.model.news.NewsTrack" table="Nedsa_News_Track"  >
        <id name="id" type="long" >
            <column name="ID"  />
            <generator class="sequence" >
                <param name="sequence">SEQ_Nedsa_News_Track</param>
            </generator>
        </id>
        <property          name="location"        column="location"          type="org.hibernate.spatial.GeometryType"                                  not-null="false" />
        <many-to-one       name="news"            column="news_id"           entity-name="org.baharan.nedsa.model.news.News"        not-null="true"     update="false"  />

        <many-to-one       name="createdBy"       column="CreatedBy"         entity-name="org.baharan.framework.model.security.User" not-null="true"     update="false"  />
        <many-to-one       name="updatedBy"       column="UpdatedBy"         entity-name="org.baharan.framework.model.security.User" not-null="true"                     />
        <property          name="createdDate"     column="CreatedDate"                                                               not-null="true"     update="false"  />
        <property          name="updatedDate"     column="UpdatedDate"                                                               not-null="true"                     />
        <property          name="ip"              column="IP"                type="string"                                           not-null="true"                     />
    </class>
</hibernate-mapping>
```

همان طور که مشاهده میکنید مدل NewsTrack که از مدلTrack که به فریم ورک اضافه شده است ارث میبرد و این حداقل نیاز برای ساخت یک مدل tracking محسوب میشود

برای خواندن و نمایش اطلاعات tracking متدی به GenericService اضافه میشود که با استفاده از آن اطلاعات tracking خوانده میشود و در نقشه نمایش داده میشود.

نمونه کد

controller

```java
@PostMapping(value = "/listByDate")
public List<TrackViewModel> listByDate(@RequestBody TrackViewModel trackViewModel) {
    return ModelMapper.mapList(iNewsTrackService.listByDate(trackViewModel), TrackViewModel.class);
}
```

service

```java
@Override
public List<Track> listByDate(TrackViewModel trackViewModel) {
    Calendar calendar = Calendar.getInstance();
    Date start = null;
    Date end = null;
    if(trackViewModel.getFromDate() != null){
        calendar.setTime(DateUtility.solarToDate(trackViewModel.getFromDate()));
        if(trackViewModel.getFromTime() != null){
            int startHour = Integer.valueOf(trackViewModel.getFromTime().split( regex: ":")[0]);
            int startMinute = Integer.valueOf(trackViewModel.getFromTime().split( regex: ":")[1]);
            calendar.set(Calendar.HOUR, startHour);
            calendar.set(Calendar.MINUTE, startMinute);
            start = calendar.getTime();
        }
    }
    if(trackViewModel.getToDate() != null){
        calendar.setTime(DateUtility.solarToDate(trackViewModel.getToDate()));
        if(trackViewModel.getToTime() != null){
            int endHour = Integer.valueOf(trackViewModel.getToTime().split( regex: ":")[0]);
            int endMinute = Integer.valueOf(trackViewModel.getToTime().split( regex: ":")[1]);
            calendar.set(Calendar.HOUR, endHour);
            calendar.set(Calendar.MINUTE, endMinute);
            end = calendar.getTime();
        }
    }
    return iNewsTrackRepository.listByDate(trackViewModel.getModelId(),start, end);
}

}
```

repository

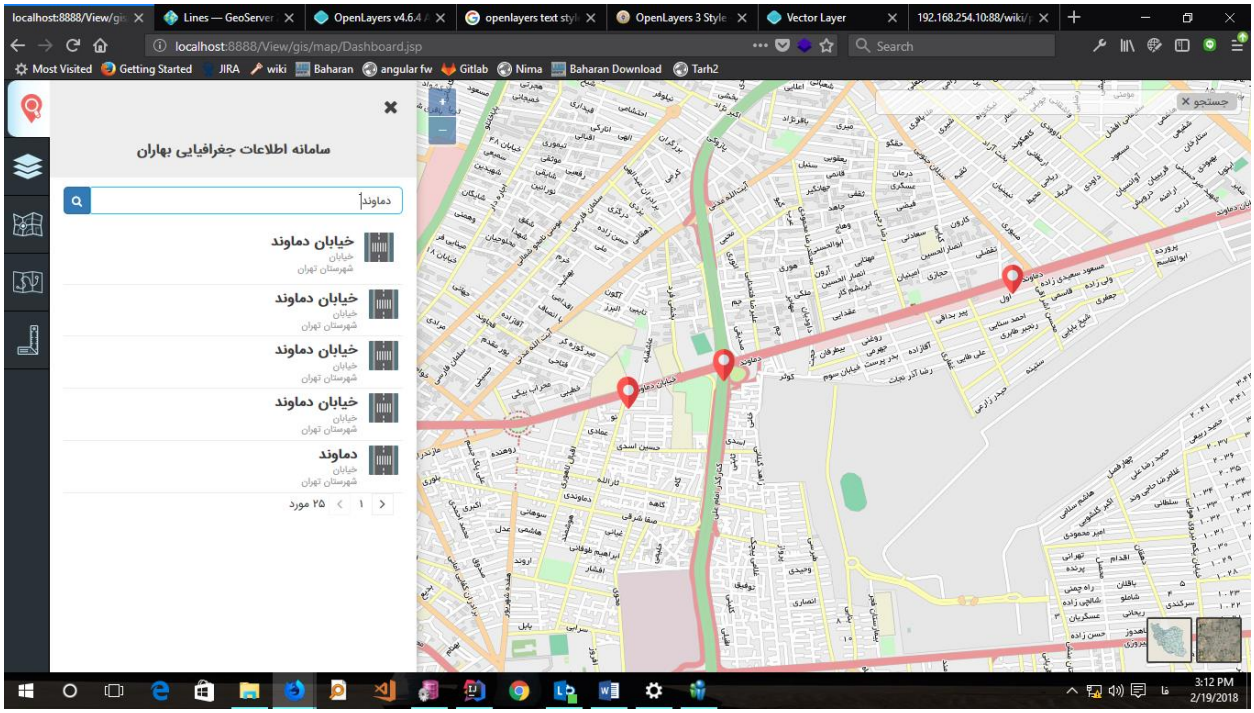
```java
31      @Override
32      public List<Track> listByDate(Long newsId, Date fromDate, Date toDate) {
33          String hql = " from " + domainClass.getName() + " e where e.news.id = :newsId";
34          HashMap<String, Object> params = new HashMap<>();
35          params.put("newsId", newsId);
36          if (fromDate != null) {
37              hql += " and e.createdDate >= :fromDate ";
38              params.put("fromDate", fromDate);
39          }
40          if (toDate != null) {
41              hql += " and e.createdDate <= :toDate ";
42              params.put("toDate", toDate);
43          }
44
45          return super.getAll(hql, params);
46      }
47
48  }
49
```

فصل ۴

راهنمای نقشه

راهنمای نقشه

جستجو

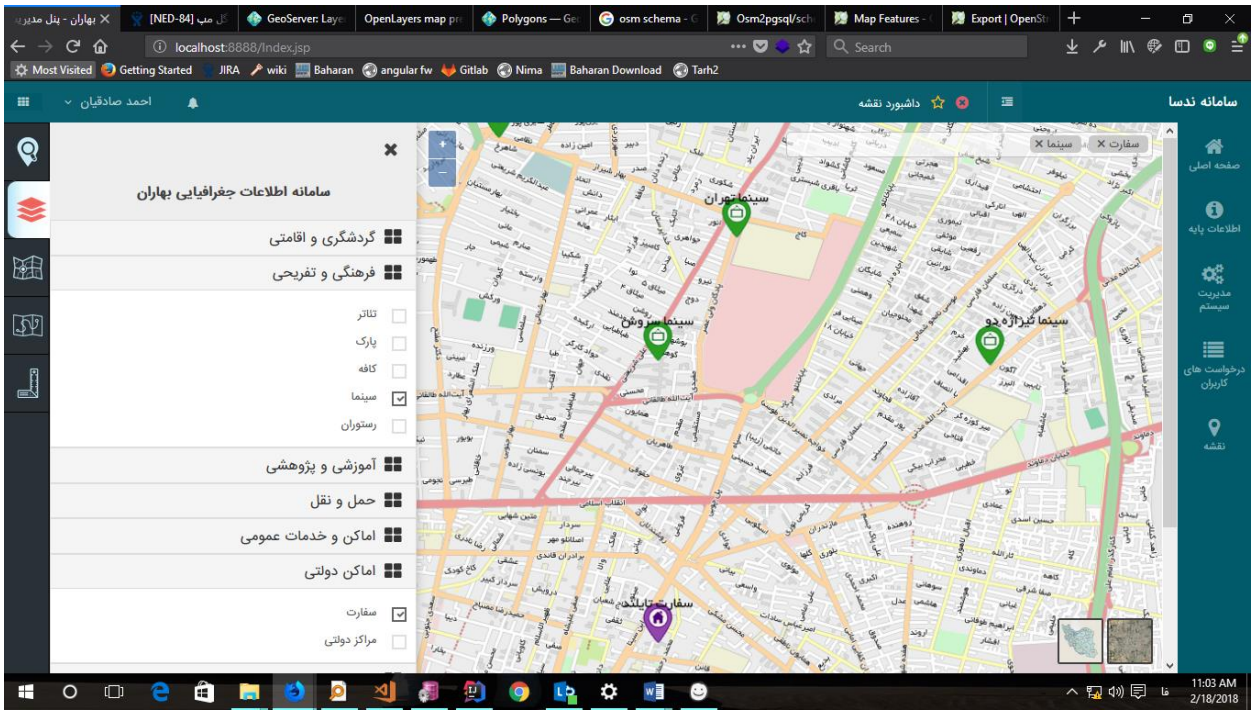


در این بخش میتوانید به جستجوی مناطق مورد نظر بر روی نقشه بپردازید . پس از نوشتن نام محل مورد نظر کلید Enter را فشار دهید تا جستجو انجام شود.

مکانیزم جستجو بر اساس نزدیک ترین محل ها به مکان فرد لاگین کرده در سیستم و اولویت هایی خاص مرتب و نمایش داده میشوند و دارای paging میباشند .

برای این منظور view ای خاص برای جستجوی مناطق از دیتا های موجود دانلود شده از سایت osm ساخته شد تا این کار و در نهایت یک materialize view برای کش کردن آن استفاده شده است.

لایه ها



در این بخش لایه هایی را که به آن ها دسترسی دارید را مشاهده میکنید و با کلیک روی هرکدام از checkbox ها میتوانید اطلاعات آن بخش را مشاهده کنید

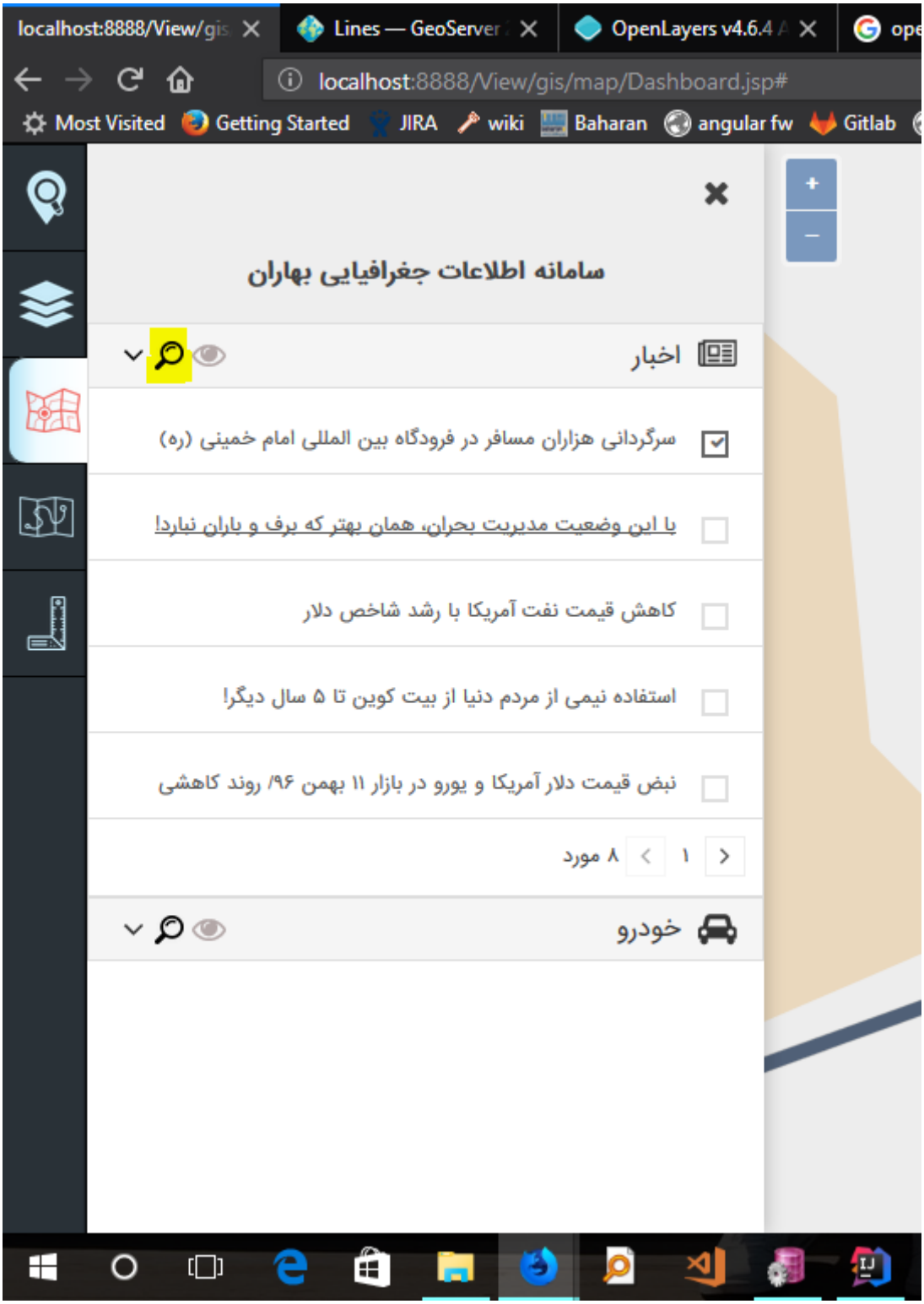
دسته بندی

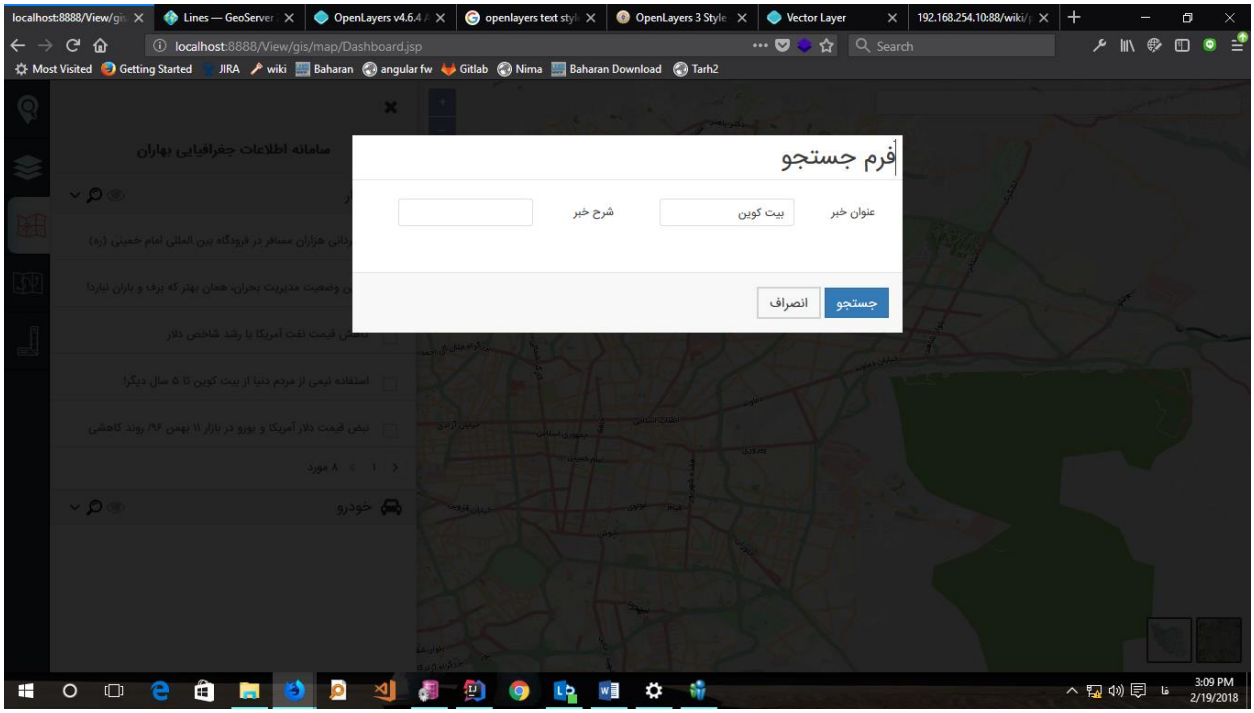
در این بخش شامل لایه های مربوط به سامانه خود هستید که در این بخش چند قابلیت وجود دارد

جستجو

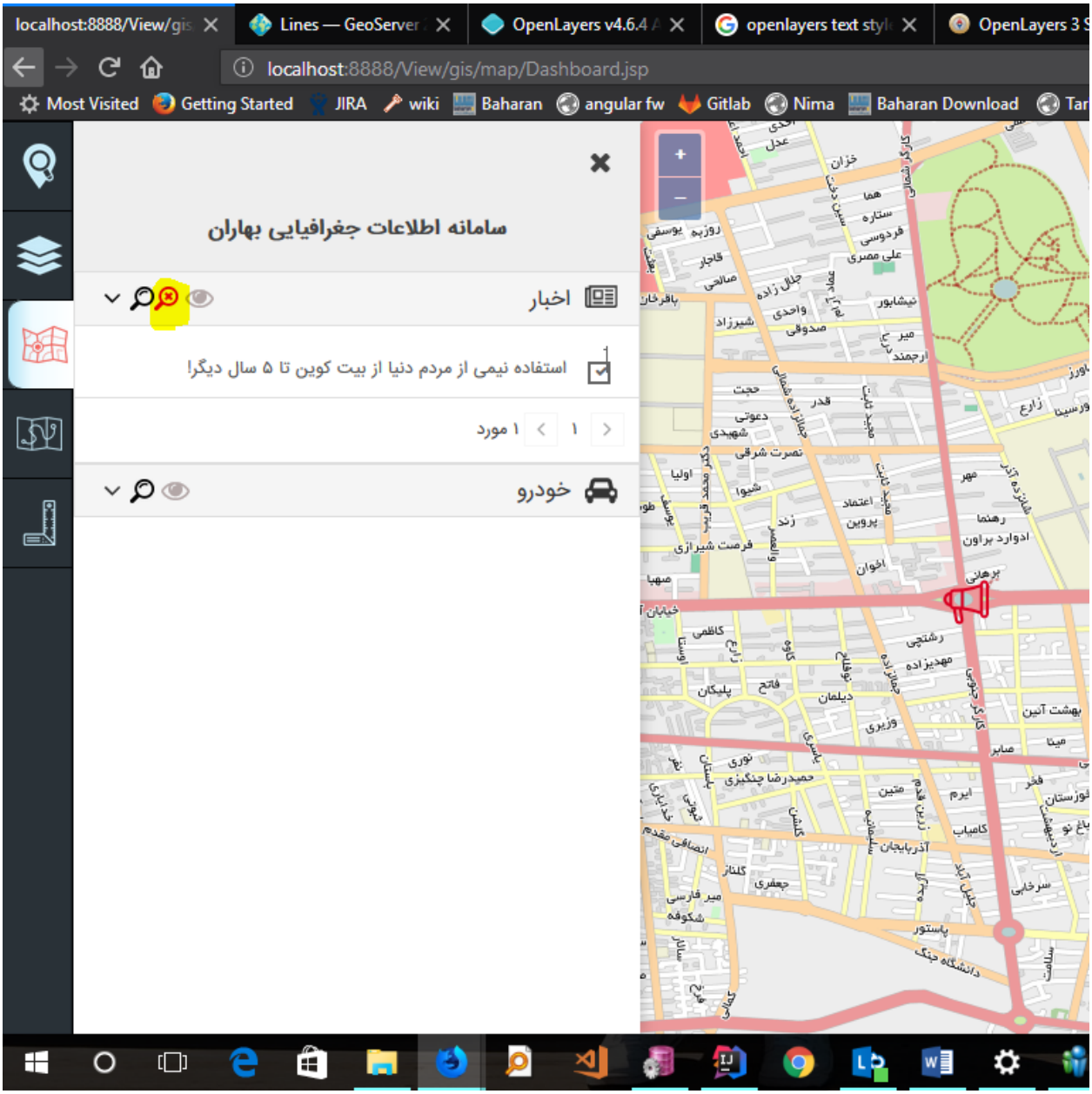
با زدن آیکن جستجو کادری برا شما باز میشود که به شما امکان وارد کردن فیلتر ها را میدهد سپس با انتخاب کلید جستجو فیلتر برای ما اعمال میشود

نمایش اطلاعات با قابلیت paging همراه است و در پایین رکورد ها تعداد ، شماره صفحه وامکان عوض کردن صفحه وجود دارد


localhost:8888/View/gis
Lines — GeoServer
OpenLayers v4.6.4
localhost:8888/View/gis/map/Dashboard.jsp#
Most Visited
Getting Started
JIRA
wiki
Baharan
angular fw
Gitlab
سامانه اطلاعات جغرافیایی بهاران
اخبار
سرگردانی هزاران مسافر در فرودگاه بین المللی امام خمینی (ره)
با این وضعیت مدیریت بحران، همان بهتر که برف و باران نبارد!
کاهش قیمت نفت آمریکا با رشد شاخص دلار
استفاده نیمی از مردم دنیا از بیت کوین تا ۵ سال دیگر!
نبض قیمت دلار آمریکا و یورو در بازار ۱۱ بهمن ۹۶/ روند کاهشی
۱
۸ مورد
خودرو

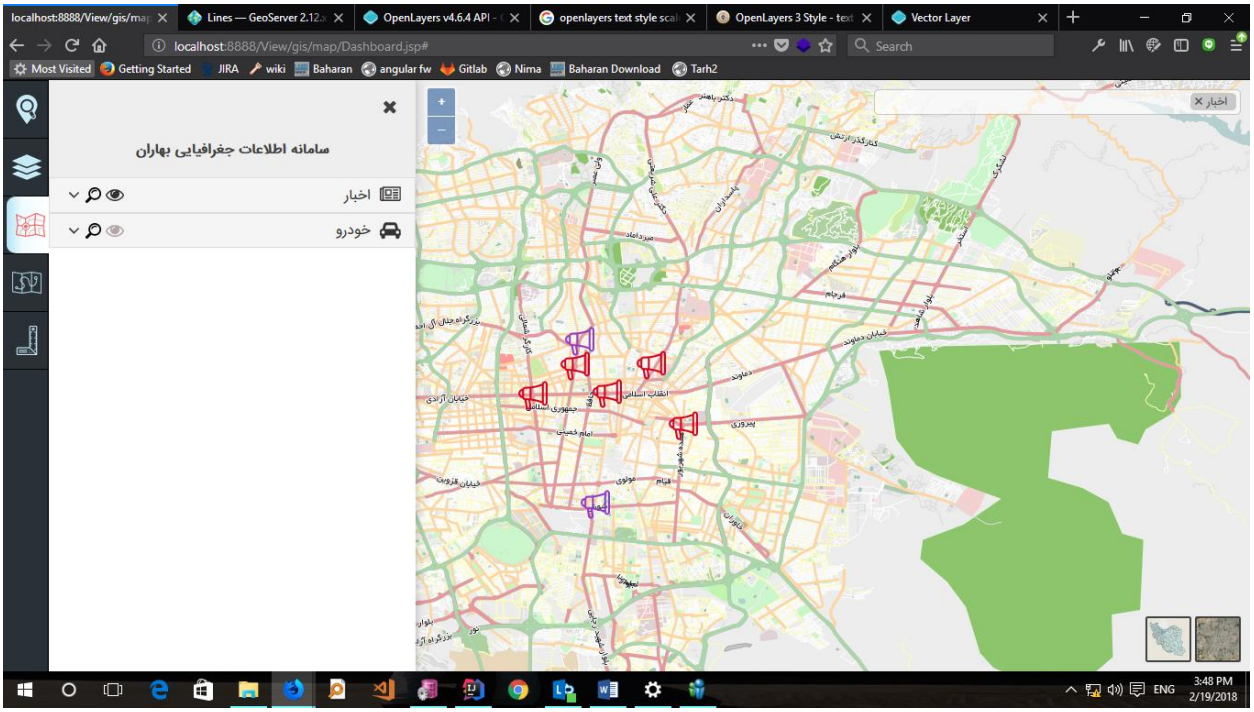


بعد از انجام عملیات جستجو برای غیر فعال کردن عملیات جستجو کافی است تا آیکن جستجو با رنگ قرمز را انتخاب کنید تا فیلتر های مربوطه حذف شوند.

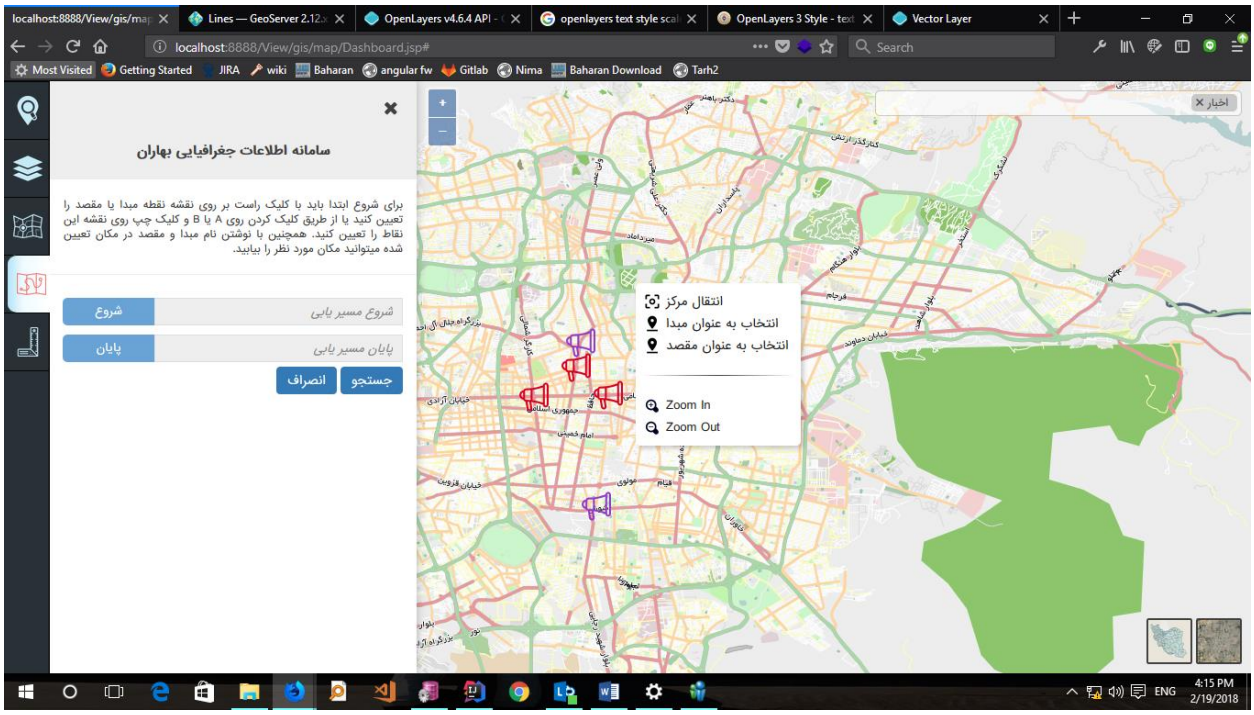

localhost:8888/View/gis
Lines — GeoServer
OpenLayers v4.6.4
openlayers text styl
OpenLayers 3
localhost:8888/View/gis/map/Dashboard.jsp
Most Visited
Getting Started
JIRA
wiki
Baharan
angular fw
Gitlab
Nima
Baharan Download
سامانه اطلاعات جغرافیایی بهاران
اخبار
استفاده نیمی از مردم دنیا از بیت کوین تا ۵ سال دیگر!
۱ مورد
خودرو

علامت چشم

با انتخاب این علمات پراکندگی مدل شما نسبت به boun ای که در تصویر میبینید به شما نمایش داده میشود .و با زوم و درگ کردن اطلاعات آپدیت میشوند



مسیریابی

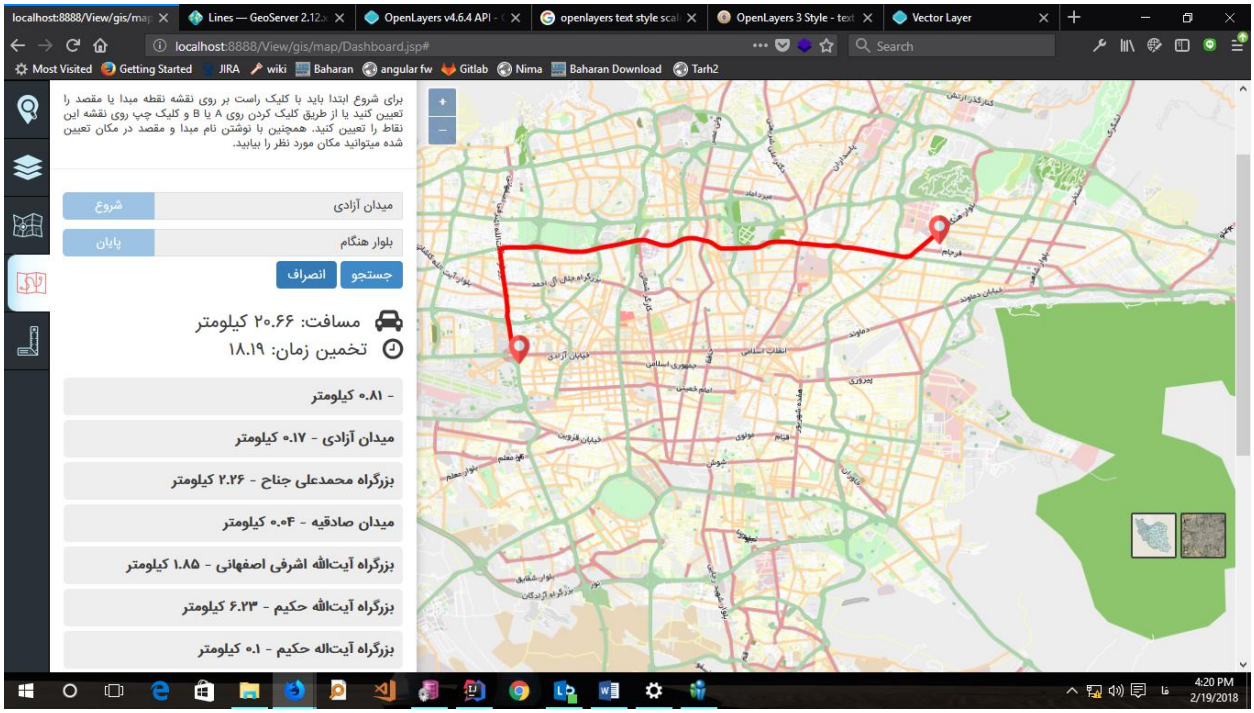


در این بخش شما میتوانید عملیات مسیر یابی را انجام دهید که کوتاه ترین مسیر برای رفتن از نقطه ای به نقطه دیگر را به شما نشان میدهد

برای این منظور به دو روش میتوانید عمل کنید

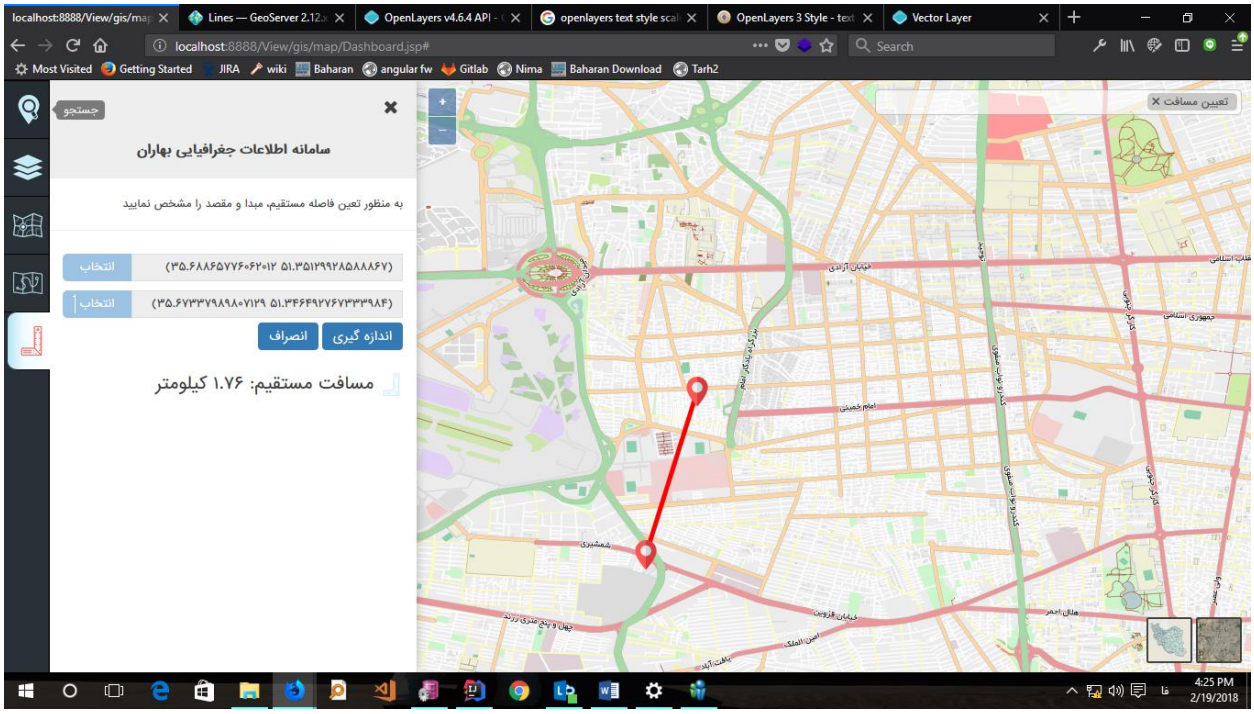
۱. کلیک راست کردن بر روی نقشه و انتخاب مبدا یا مقصد
۲. انتخاب دکمه شروع یا پایان و کلیک کردن بر روی مکان مورد نظر در نقشه

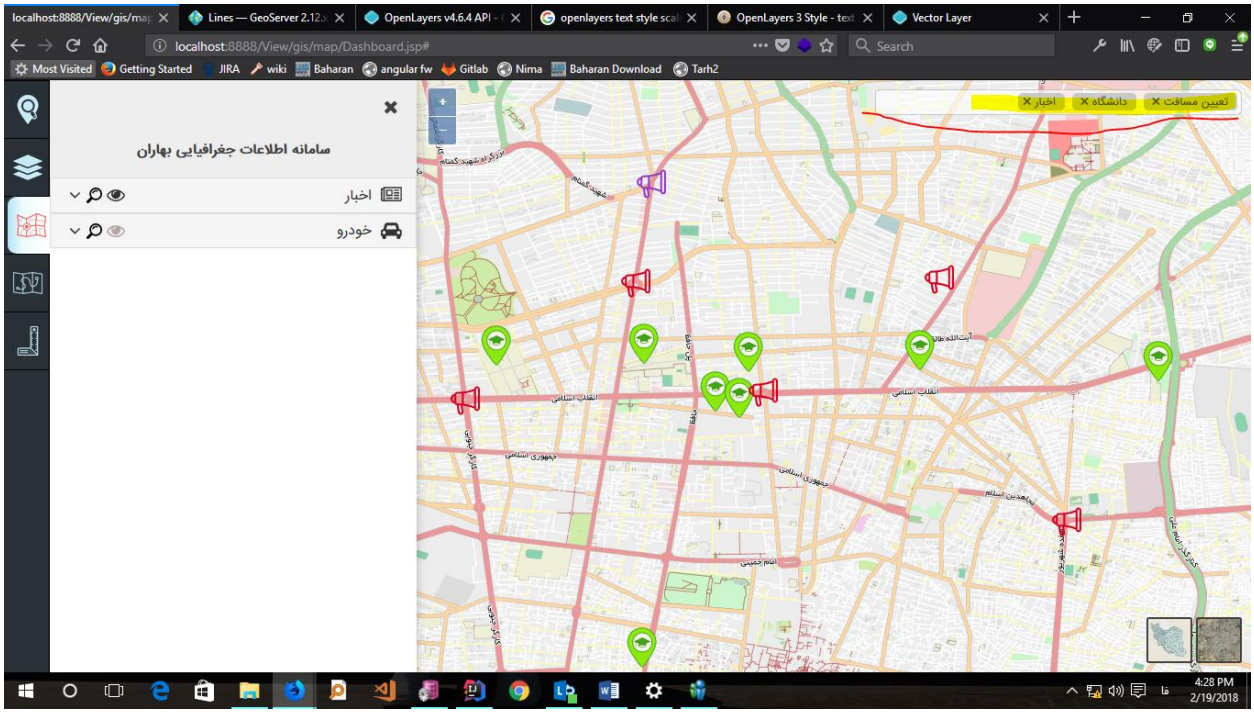
سپس با زدن کلید جستجو عملیات جستجو انجام شده و به شما نمایش داده میشود.


localhost:8888/View/gis/map/Dashboard.jsp#
Most Visited
Getting Started
JIRA
wiki
Baharan
angular fw
Gitlab
Nima
Baharan Download
Tarh2
برای شروع ابتدا باید با کلیک راست بر روی نقشه نقطه مبدا یا مقصد را تعیین کنید یا از طریق کلیک کردن روی A یا B و کلیک چپ روی نقشه این نقاط را تعیین کنید. همچنین با نوشتن نام مبدا و مقصد در مکان تعیین شده میتوانید مکان مورد نظر را بیابید.
شروع
میدان آزادی
پایان
بلوار هنگام
جستجو
انصراف
مسافت: ۲۰.۶۶ کیلومتر
تخمین زمان: ۱۸.۱۹
- ۰.۸۱ کیلومتر
میدان آزادی - ۰.۱۷ کیلومتر
بزرگراه محمدعلی جناح - ۲.۲۶ کیلومتر
میدان صادقیه - ۰.۰۴ کیلومتر
بزرگراه آیت‌الله اشرفی اصفهانی - ۱.۸۵ کیلومتر
بزرگراه آیت‌الله حکیم - ۶.۲۳ کیلومتر
بزرگراه آیت‌اله حکیم - ۰.۱ کیلومتر
4:20 PM
2/19/2018

تعیین مسافت

در این بخش با کلیک بر روی مبدا و مقصد و کلیک کردن بر روی نقشه محل های انخاب شده مشخص میشوند و با زدن کلید اندازه گیری نتیجه مسافت به کیلومتر برای شما نمایش داده میشود.



نوار Navigation

در این قسمت این شما میتوانید لیست لایه هایی که در حال نمایش هستند را مشاهده کنید و در صورت لزوم با دسترسی سریع تری با کلیک بر روی علامت ضربدر نمایش آن لایه را لغو کنید