

**دانشکده مهندسی نقشه برداری و اطلاعات مکانی دانشگاه تهران**

**گزارش پروژه اول**

**دانشجویان:**

کاروان جلالی

فرشاد شمسی خانی

**استاد:**

دکتر عباس عابدینی

**درس:**

مفاهیم نظری سیستم اطلاعات مکانی

**روش انجام پروژه و پیاده سازی**

**مرحله اول، نصب GeoServer:**

با توجه به هدف پروژه، در ابتدا اقدام به نصب نرم افزار ژئوسرور می کنیم. می دانیم ژئوسرور (GeoServer) یک نرم‌افزار متن‌باز (open-source) و سرور جغرافیایی برای ذخیره، پردازش و ارائه داده‌های مکانی (جغرافیایی) از طریق اینترنت است. ژئوسرور امکان ایجاد و انتشار نقشه‌های تعاملی را فراهم می‌کند و به کاربران اجازه می‌دهد تا داده‌های جغرافیایی خود را با فرمت‌های استاندارد به اشتراک بگذارند و آن‌ها را در سامانه‌های اطلاعات مکانی (GIS) استفاده کنند.

ژئوسرور (GeoServer) چندین روش نصب مختلف دارد که کاربران می‌توانند بسته به نیاز و سیستم عامل خود یکی از آن‌ها را انتخاب کنند:

* نصب به صورت مستقل (Standalone) با فایل ZIP
* نصب به عنوان سرویس (Service) در ویندوز ( روش مورد استفاده در این پروژه)
* نصب با استفاده از WAR در سرورهای وب (مانند Tomcat)
* نصب با Docker

قبل از نصب این نرم افزار باید دقت کنیم که برای اجرای ژئوسرور، نیاز به Java Development Kit (JDK) داریم. ما از JDK 11 چون همخوانی بیشتری با ژئوسرور دارد استفاده کرده ایم.

**مرحله دوم، اجرای GeoServer:**

بعد از طی کردن مراحل نصب، برای اجرای ژئوسرور در منوی Start روی Start geoserver کلیک می کنیم و سپس در یک مرورگر به صورت لوکال توسط آدرس <http://localhost:8080/geoserver> آن را اجرا می کنیم و با وارد کردن مشخصات Login می کنیم.

**مرحله سوم، انتشار لایه ها در GeoServer:**

1. در ابتدا باید یک Workspace ایجاد کنیم. Workspace محیطی است که لایه‌ها و داده‌های ما را گروه‌بندی می‌کند. مراحل به این صورت هست:

* در پنل سمت چپ، روی Workspaces کلیک می کنیم.
* گزینه Add new workspace را انتخاب می کنیم.
* برای Workspace جدید یک نام (Name) انتخاب می کنیم.
* یک URI منحصر به فرد (مانند آدرس وب) برای فضای کاری وارد می کنیم. این URI نیازی به واقعی بودن ندارد و تنها به عنوان شناسه‌ای برای سازماندهی داده‌ها استفاده می‌شود و در نهایت دکمه Submit را می زنیم تا ایجاد شود.

1. در مرحله بعدی باید یک Store ایجاد کنیم. Store مکانی است که داده‌های ما را ذخیره می کند (مانند فایل‌های shapefile، پایگاه داده، یا داده‌های Raster). مراحل:

* در پنل سمت چپ، روی Stores کلیک می کنیم.
* گزینه Add new Store را انتخاب می کنیم.
* نوع داده‌ای که می خواهیم منتشر کنیم را انتخاب می کنیم. (به‌عنوان مثال، اگر یک فایل shapefile داریم، گزینه‌ی Shapefile را انتخاب می کنیم).
* Workspace ای که در مرحله قبل ایجاد کردیم را انتخاب می کنیم.
* در بخش Data Source, فایل های shapefile را آپلود می کنیم یا مسیری که فایل در آن قرار دارد را وارد می کنیم و در نهایت دکمه save را می زنیم.

نکته: در این پروژه ما ShapeFile هایی از استان اصفهان استفاده کرده ایم که قبلا در ArcGis آن ها را Shapefile اصلی ایران Clip کرده ایم.

1. در این مرحله می خواهیم Layers (لایه ها) را منتشر کنیم. پس از ذخیره Store، ژئوسرور به‌صورت خودکار لیستی از لایه‌های موجود در آن Store را نشان می‌دهد. مراحل:

* روی نام لایه‌ای که می خواهیم منتشر کنیم، کلیک می کنیم.
* به تب Publishing می رویم و مشخصات لایه مانند عنوان (Title) و توضیحات (Abstract) را وارد می کنیم.
* به تب Coordinate Reference Systems می رویم و سیستم مختصات لایه را بررسی و در صورت لزوم آن را تنظیم می کنیم. مطمئن می شومی که سیستم مختصات درست برای لایه انتخاب شده است.
* در تب Bounding Boxes، روی گزینه Compute from data و Compute from native bounds کلیک می کنیم تا ژئوسرور محدوده جغرافیایی لایه را به‌صورت خودکار تنظیم کند و در نهایت دکمه Save را می زنیم.

1. برای مشاهده و تست لایه ها می توانیم به پنل سمت چپ برویم و روی Layer Preview کلیک کنیم. لایه مورد نظر را انتخاب می کنیم و از ستون Common Formats گزینه OpenLayers را انتخاب کرده تا لایه در نقشه به‌صورت پیش‌نمایش باز شود.

نکته: این لینک در صفحه HTML صدا زده می شود.

**مرحله چهارم، ایجاد ساختار پایه HTML :**

برای تلفیق لایه ها، داده ها و نقشه ها نیاز به ساختاری داریم تا این اطلاعات در بستر اینترنت یا شبکه از طریق سروری که در مرحله قبل توضیح داده شده به نمایش در آیند. ما در این پروژه به کمک یک صفحه HTML که دارای CSS و SCRIPT هست سعی کرده ایم این مهم را انجام دهیم. بعد از ایجاد ساختاری کلی HTML5 مراحل اضافه کردن اسکریپت های OpenLayer را دنبال می کنیم.

**مرحله پنجم، ایجاد BaseMap :**

برای ایجاد یک نقشه مبنا، ابتدا باید کتابخانه OpenLayers را در صفحه خود وارد کنیم. البته می‌توانید از CDN استفاده کنیم. با مراجعه به آدرس [اینجا](https://openlayers.org/en/latest/examples/simple.html) اسکریپت و کدهای HTML را بر می داریم و در پروژه خودمان استفاده می کنیم. در نهایت، با استفاده از جاوااسکریپت، نقشه را در صفحه نمایش می دهیم. در بخش اسکریپت، ol.Map یک نقشه جدید ایجاد می‌کند. layers لایه‌های نقشه را تعیین می‌کند. view برای تنظیم نمای نقشه استفاده می‌شود که موقعیت اولیه و سطح زوم را در اینجا تعیین می‌کنیم.

**مرحله ششم، ایجاد قابلیت اندازه گیری طول و مساحت:**

برای اضافه کردن قابلیت اندازه‌گیری فاصله و مساحت روی نقشه با استفاده از OpenLayers، نیاز به پیاده‌سازی یک ابزار اندازه‌گیری داریم که به کاربر اجازه دهد با کلیک کردن روی نقشه، فاصله یا مساحت را اندازه‌گیری کند. در اینجا، مراحلی پیاده‌سازی این قابلیت را بیان می کنیم.

* بعد از ایجاد قابلیتی برای انتخاب حالت اندازه‌گیری (فاصله یا مساحت) در HTML، در جاوااسکریپت، ابزار اندازه‌گیری و لایه‌ی مورد نیاز برای نمایش نتایج را ایجاد می کنیم. در این پروژه نیز از مثال های OpenLayers برای پیاده سازی این بخش استفاده کرده ایم. در فایل HTML این پروژه، از یک منوی کشویی (dropdown) با آی دی type برای ایجاد قابلیت های این ویژگی استفاده شده است.
* در بخش اسکریپت، const typeSelect2 به عناصر HTML با شناسه type2 اشاره دارد که یک منوی کشویی است. در تابع addInteraction2 چنانچه const value = typeSelect2.value باشد مقدار انتخاب‌شده در منوی کشویی دریافت می‌شود که اگر مقدار انتخاب‌شده 'None' نباشد، ساخت و ساز رسم فراهم می شود.
* draw2 = new ol.interaction.Draw یک interaction است که برای رسم اشکال روی یک لایه با منبع vectorSource استفاده می‌شود. زمانی که کاربر رسم را به پایان می‌رساند، یک رویداد drawend اجرا می‌شود. در اینجا استایل ویژه‌ای به ویژگی نهایی رسم‌شده اختصاص داده می‌شود. typeSelect2.onchange = function یک interaction برای تغییرات در منوی کشویی است. سپس اینتراکشن addInteraction2 دوباره اجرا می‌شود تا interaction رسم جدید با نوع انتخاب‌شده در منوی کشویی ایجاد و به نقشه اضافه شود.
* **مرحله هفتم، WMS GetFeatureInfo:**

برای ایجاد قابلیت identify لایه های افزوده شده به نقشه، ما در این پروژه از قابلیت GetFeatureInfo از مثال های OpenLayers استفاده کرده ایم. ([اینجا](https://openlayers.org/en/latest/examples/getfeatureinfo-image.html)) این بخش، یک رویداد singleclick روی نقشه اضافه می‌کند که با کلیک کاربر، اطلاعات ویژگی را برای نقطه‌ی کلیک‌شده از سرور درخواست می‌کند. توضیح کوتاه در مورد اسکریپت ها:

map.getView().getResolution() وضوح فعلی نمای نقشه را برمی‌گرداند، که به عنوان یکی از پارامترها برای درخواست GetFeatureInfo به سرور WMS ارسال می‌شود.

evt.coordinate مختصات نقطه کلیک‌شده توسط کاربر است.

EPSG:4326 سیستم مختصات نقشه است.

getFeatureInfoUrl برای تولید URL مربوط به درخواست GetFeatureInfo استفاده می‌شود.

map.getLayers().getArray(): تمام لایه‌های موجود در نقشه را به‌صورت آرایه‌ای برمی‌گرداند. با استفاده از forEach، هر لایه بررسی می‌شود و اگر لایه دارای شناسه (id) باشد، یک گزینه به layerOptions اضافه می‌شود.

* **مرحله هشتم، اجرای پروژه:**

پروژه که شامل لینک لایه های منتشر شده در ژئوسرور، فایل HTML، CSS و فونت است به راحتی اجرا می شود و از نظر عملکرد فنی اختلالی ندارد که عملکرد آن را در قالب یک ویدیو ضمیمه این گزارش کار کرده ایم.

با تشکر – گروه اول