**DOCUMENTACIÓN**

A continuación una breve descripción de las tecnologías que se usan en este proyecto.

**Node Js**: Plataforma construida en Javascript, para aplicaciones de red escalables de fácil construcción. Usado en el presente proyecto para la implementación del servidor. Visitar: <https://nodejs.org/>

**Express:** framework web que proporciona un conjunto robusto de características para aplicaciones web y móviles. Usado para la implementación del servidor junto con Node Js. Visitar: <http://expressjs.com/es/>

**Nunjucks:** Lenguaje de plantillas para javascript. Muy útil al momento de crear las vistas, permitiendo separar la programación del diseño y dejando un código más limpio y de más fácil mantenimiento. Visitar: <https://mozilla.github.io/nunjucks/>

**Html:** Lenguaje para el diseño de vistas web.

**Jquery:** **Versión 1.11.2.** Librería de javascript, que nos permite un manejo más flexible y dinámico de los elementos HTML a través de funciones built-in. Visitar: <https://jquery.com/>

**Jquery UI: Versión 1.11.4.** Conjunto de interacciones de interfaz del usuario, efectos, temas y más, construidas en base a la librería Jquery. Nos permite construir aplicaciones web altamente interactivas. Visitar: <http://jqueryui.com/>

**Bootstrap:** **Versión 3.3.5.** Framwork para desarrollo proyectos responsive y móvil en la web. <http://getbootstrap.com/>

**Openlayers: Versión 3.6.x.** Librería de alto rendimiento para necesidades de mapeo. Visita: <http://openlayers.org/>

**Geoserver:** Servidor de código abierto para compartir datos geoespaciales. <http://geoserver.org/>

SOBRE LA INFORMACIÓN GEOGRÁFICA Y COMERCIAL

La información geográfica se encuentra almacenada en una base de datos geográfica ubicada en la dirección: 20.0.10.73 en el puerto 5434. La base de datos se llama “sisgeco\_localhost” y el esquema es “gis”. A grandes rasgos este esquema contiene información catastral de predios, clientes, sectores, red de agua, alcantarillado y otros.

La información comercial se encuentra en la misma base de datos pero pertenece al esquema “public”. La información proviene de una copia de la base de datos de gisteco.

SOBRE LAS CAPAS GIS

Las capas gis se crean en geoserver en un espacio de trabajo PostGis y se visualizan con openlayers. Cabe destacar los datos geográficos de las capas de geoserver se encuentran en el sistemas EPSG: 32717 y openlayers usa por defecto el sistema EPSG: 3857. En este proyecto se trabajó con el sistema por defecto de openlayers, así que fue necesario la conversión entre sistemas (en el código o las consultas sql) cada vez que era necesario.

Ligada a cada capa hay un estilo que define como lucirán (gráficamente) las capas cuando se muestren con openlayers. La estructura de dichos estilos se encuentra en el archivo “estilos-sld.xml” para mayor detalle y análisis. Cualquier modificación en este archivo no surtirá efecto en la visualización pues no es un archivo de configuración. Si se desea cambiar alguno de los estilos será necesario desplazarse hasta la pestaña “Estilos” de geoserver.

FUNCIONES DEL LADO DEL SERVIDOR

**Colores y significados**

**Verde:** Validación hecha.

**Amarillo:** Validación hecha en otro nivel.

**Ninguno:** Ninguna validación creada.

ARCHIVO: *cliente.js* / RUTA: *modelos/cliente.js*

**getCoordsByNroSuministro**

**Parámetros:** nro\_suministro,res(response object)

**Resultados:** retorna al cliente las coordenadas del cliente en la proyección EPSG: 3857, según su número de suministro.

**Llamadas:** desde “ubicarCliente – mapa.js”

**Posibles acontecimientos**

|  |  |  |  |
| --- | --- | --- | --- |
| Condición | Resultado | | |
| nro\_suministro es vació | Error: no existe la función gis.f\_buscarcliente\_por\_num\_suministro() | |
| No existe el cliente con ese número de suministro. | El mapa se sale de cuadro. | |
| Nro\_suministro tiene caracteres q no son números. | Error de sintaxis. |
| Nro\_suministro tiene más de 8 cifras. | El mapa se sale de cuadro. | |

**getCoordsByCodCatastral**

**Parámetros:** preregion, prezona, presector, premzn, prelote,res (response object)

**Resultados:** retorna al cliente las coordenadas del cliente en la proyección EPSG: 3857, según su código catastral.

**Llamadas:** desde “ubicarClientePorCodCat – mapa.js”

**Posibles acontecimientos**

|  |  |  |
| --- | --- | --- |
| Condición | Resultado | |
| Alguno de los parámetros no es un número | Error: no existe la columna(cd). Ejm: Cuando se ingresa ‘cd’ en la caja prezona | |
| Alguno de los parámetros es vació | Error: error de sintaxis en o cerca de «,». | |
| No existe ningún cliente con ese código catastral. | Error de sintaxis. |
|  |  | |

**verFichaCatastral**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“ficha\_catastral\_piura.html”) con la información básica de la ficha catastral: número de ficha y número de suministro del cliente.

**Llamadas:** desde “info\_cliente.html”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verDatosBasicosFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“datos\_basicos\_ficha\_catastral\_piura.html”) con los datos básicos de la ficha catastral: revisar f\_getdatosbasicos\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verDatosBasicosFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verDatosPropiedadFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“datos\_propiedad\_ficha\_catastral\_piura.html”) con los datos de la propiedad: revisar f\_getdatospropiedad\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verDatosPropiedadFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verConexAguaFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“conexion\_agua\_ficha\_catastral\_piura.html”) con los datos de la conexión de agua: revisar f\_getconexionagua\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verConexAguaFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verConexDesagueFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“conexion\_desague\_ficha\_catastral\_piura.html”) con los datos de la conexión de desague: revisar f\_getconexiondesagua\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verConexDesagueFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verValoresMinFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“valores\_minimos\_ficha\_catastral\_piura.html”) con los valores mínimos admisibles: revisar f\_getvaloresminimos\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verValoresMinFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verDatosComplemFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“datos\_complementarios\_ficha\_catastral\_piura.html”) con los datos complementarios: revisar f\_getdatoscomplem\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verDatosComplemFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verInformacionGeograficaFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“informacion\_geografica\_ficha\_catastral\_piura.html”) con la información geográfica: revisar f\_getinformaciongeografica\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verInformacionGeograficaFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verCroquisFichaCat**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“croquis\_ficha\_catastral\_piura.html”) con las calles que rodean al predio: revisar f\_getcallespredio\_ficha\_cat de “ocurrencias.docx”.

**Llamadas:** desde “verCroquisFichaCat – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verCuentaCorriente**

**Parámetros:** número de suministro, página, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“cuenta\_corriente\_cliente.html”) con detalle de la cuenta corriente del cliente según la página pasada como parámetro: revisar f\_getdetallecuentacorriente\_cliente\_por\_pag de “ocurrencias.docx”.

**Llamadas:** desde “verCuentaCorriente – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
| Página con valor negativo o que no existe | No retorna resultados. |
|  |  |

**getNumPagsCuentaCorriente**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“paginacion\_cuenta\_corriente.html”) con el número de páginas de la cuenta corriente del cliente: revisar f\_getnumpaginas\_cuentacorriente\_cliente de “ocurrencias.docx”.

**Llamadas:** desde “paginacionCuentaCorriente – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verLecturas**

**Parámetros:** número de suministro, página, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“lecturas\_cliente.html”) con las lecturas del cliente según la página pasada como parámetro: revisar f\_getlecturas\_cliente\_por\_pag de “ocurrencias.docx”.

**Llamadas:** desde “verLecturasCliente – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
| Página con valor negativo o que no existe | No retorna resultados. |
|  |  |

**getNumPagsLecturas**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“paginacion\_lecturas.html”) con el número de páginas de las lecturas del cliente: revisar f\_getnumpaginas\_lecturas\_cliente de “ocurrencias.docx”.

**Llamadas:** desde “paginacionLecturas – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verFacturacion**

**Parámetros:** número de suministro, página, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“facturacion\_cliente.html”) con facturación del cliente según la página pasada como parámetro: revisar f\_getfacturacion\_cliente\_por\_pag de “ocurrencias.docx”.

**Llamadas:** desde “verFacturacionCliente – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
| Página con valor negativo o que no existe | No retorna resultados. |
|  |  |

**getNumPagsFacturacion**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“paginacion\_facturacion.html”) con el número de páginas de la facturación del cliente: revisar f\_getnumpaginas\_facturacion\_cliente de “ocurrencias.docx”.

**Llamadas:** desde “paginacionFacturacion – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verDetalleFactura**

**Parámetros:** id\_recibo\_factura, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“detalle\_factura\_cliente.html”) con el detalle de una factura: revisar f\_getdetalle\_factura\_cliente de “ocurrencias.docx”.

**Llamadas:** desde “verDetalleFactura – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Recibo no existe | No retorna resultados. Aún no se han producido excepciones de esta función. |
| Id\_recibo\_factura con formato inválido | No retorna resultados. |
|  |  |

**verConexionAgua**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“conexion\_agua\_cliente.html”) con los datos de la conexión de agua: revisar f\_getconexionagua\_cliente de “ocurrencias.docx”.

**Llamadas:** desde “verConexionAguaCliente – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**verConexionDesague**

**Parámetros:** número de suministro, res (response object), nunjucks (nunjucks object)

**Resultados:** retorna al cliente contenido HTML (“conexion\_desague\_cliente.html”) con los datos de la conexión de agua: revisar f\_getconexiondesague\_cliente de “ocurrencias.docx”.

**Llamadas:** desde “verConexionDesagueCliente – ficha-catastral-piura.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| Número de inscripción con formato inválido | Desconocido. Aún no se han producido excepciones de esta función. |
|  |  |

**comunicarError**

**Parámetros:** res (response object), mensaje

**Observaciones:** Utilizado como respuesta al servidor en caso de no pasar una previa validación

**Resultados:** retorna al cliente el código 400 (Bad Request) con el mensaje pasado como parámetro

**Llamadas:** desde “getCoordsCliente – cliente.js”, “getCoordsClienteByCodCatastral – cliente.js”

ARCHIVO: *utilidades.js* / RUTA: *modelos/gisteco.js*

**getCantClientesActivos**

**Parámetros:** res (response object)

**Resultados:** retorna al cliente la cantidad total de clientes activos: revisar f\_getcant\_clientes\_activos de “ocurrencias.docx”.

**Llamadas:** desde “verTotalClientesInscritos – mapa.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
|  |  |
|  |  |

**distanciaTotalRedAgua**

**Parámetros:** res (response object)

**Resultados:** retorna al cliente distancia en metros de la red de agua: revisar f\_getdistanciatotal\_red\_agua de “ocurrencias.docx”.

**Llamadas:** desde “verTotalDistRedAgua – mapa.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
|  |  |
|  |  |

**distanciaTotalRedDesague**

**Parámetros:** res (response object)

**Resultados:** retorna al cliente distancia en metros de la red de alcantarillado: revisar f\_getdistanciatotal\_red\_desague de “ocurrencias.docx”.

**Llamadas:** desde “verTotalDistRedDesague – mapa.js”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
|  |  |
|  |  |

**getDistritosFromProvincia**

**Parámetros:** id provincia, res (response object), nunjucks

**Resultados:** retorna al cliente contenido HTML (“combo\_distritos.html”) con los distritos de la provincia identificado por **id provincia**: revisar f\_getdistritos\_from\_provincia de “ocurrencias.docx”.

**Llamadas:** desde “análisis-de-usuarios.html”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
|  |  |
|  |  |

ARCHIVO: *utilidades.js* / RUTA: *modulos/utilidades.js*

**mapear**

**Parámetros:** val, index

**Observaciones:** Utilizado para poder marcar el checkbox correcto cuando se muestra la información de ficha catastral.

**Resultados:** retorna “**checked**” cuando **val** e **index** son iguales, en otro caso retorna “”.

**Llamadas:** desde todas las vistas de la ficha catastral: conexion\_agua\_ficha\_catastral\_piura, conexion\_desague\_ficha\_catastral\_piura, etc.

ARCHIVO: *utilidades.js* / RUTA: *modulos/utilidades.js*

**validarNroSuministro**

**Parámetros:** número de suministro

**Resultados:** retorna **0** cuando el **número de suministro** tiene el formato correcto, pero un **mensaje** dirigido al usuario cuando el formato es incorrecto.

**Llamadas:** desde “getCoordsCliente – cliente.js”

**validarCodCatastral**

**Parámetros:** preregion,prezona,presector,premzn,prelote

**Resultados:** retorna **0** cuando el **código catastral** tiene el formato correcto, pero un **mensaje** dirigido al usuario cuando el formato es incorrecto.

**Llamadas:** desde “getCoordsClienteByCodCatastral – cliente.js”

FUNCIONES DEL LADO DEL CLIENTE

ARCHIVO: *validaciones.js* / RUTA: *public/javascripts/validaciones.js*

**validarNroSuministro**

**Parámetros:** número de suministro

**Resultados:** retorna **0** cuando el **número de suministro** tiene el formato correcto, pero un **mensaje** dirigido al usuario cuando el formato es incorrecto.

**Llamadas:** desde “ubicarCliente – mapa.js”

**validarCodCatastral**

**Parámetros:** preregion,prezona,presector,premzn,prelote

**Resultados:** retorna **0** cuando el **código catastral** tiene el formato correcto, pero un **mensaje** dirigido al usuario cuando el formato es incorrecto.

**Llamadas:** desde “ubicarClientePorCodCat – mapa.js”

**numericInputValidation**

**Parámetros:** un caracter

**Resultados:** retorna **true** cuando el **caracter** es un dígito y **false** en caso contrario.

**Llamadas:** desde “ubicarClientePorCodCat – mapa.js”,”buscar\_cliente.html”, “buscar\_cliente\_por\_cod\_cat.html”

ARCHIVO: *ficha-catastral-piura.js* / RUTA: *public/javascripts/ficha-catastral-piura.js*

**verDatosBasicosFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verDatosBasicosFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verDatosPropiedadFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verDatosPropiedadFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verConexAguaFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verConexAguaFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verConexDesagueFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verConexDesagueFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verValoresMinFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verValoresMinFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verDatosComplemFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verDatosComplemFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verInformacionGeograficaFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verInformacionGeograficaFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verCroquisFichaCat**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verCroquisFichaCat

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verConexionAguaCliente**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verConexionAguaCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verConexionDesagueCliente**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verConexionDesagueCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**verCuentaCorriente**

**Parámetros:** número de suministro, página

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verCuentaCorrienteCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**paginacionCuentaCorriente**

**Parámetros:** número de suministro

**Observaciones:** Hace una llamada a través de Ajax a la ruta / getNumPagsCuentaCorrienteCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**cambiarPagCuentaCorriente**

**Parámetros:** Objeto <a>

**Resultados:** llama a la función verCuentaCorriente – ficha-catastral-piura.js para cambiar a la página que indica el objeto <a>

**Llamadas:** desde “ficha-catastral-piura.js”

**verLecturasCliente**

**Parámetros:** número de suministro, página

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verLecturasCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**paginacionLecturas**

**Parámetros:** número de suministro

**Observaciones:** Hace una llamada a través de Ajax a la ruta / getNumPagsLecturasCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**cambiarPagLecturas**

**Parámetros:** Objeto <a>

**Resultados:** llama a la función verLecturasCliente – ficha-catastral-piura.js para cambiar a la página que indica el objeto <a>

**Llamadas:** desde “ficha-catastral-piura.js”

**verFacturacionCliente**

**Parámetros:** número de suministro, página

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verFacturacionCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**paginacionFacturacion**

**Parámetros:** número de suministro

**Observaciones:** Hace una llamada a través de Ajax a la ruta / getNumPagsFacturacionCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

**cambiarPagFacturacion**

**Parámetros:** Objeto <a>

**Resultados:** llama a la función verFacturacionCliente – ficha-catastral-piura.js para cambiar a la página que indica el objeto <a>

**Llamadas:** desde “ficha-catastral-piura.js”

**verDetalleFactura**

**Parámetros:** Objecto <a> que contiene el **número de factura**.

**Observaciones:** Hace una llamada a través de Ajax a la ruta / verDetalleFacturaCliente

**Resultados:** carga en el navegador el HTML que retorna la llamada ajax.

**Llamadas:** desde “ficha-catastral-piura.js”

ARCHIVO: *mapa.js* / RUTA: *public/javascripts/mapa.js*

**cargarCapas**

**Parámetros:** Ninguno.

**Resultados:** Crea todas las capas GIS y Base.

**Llamadas:** desde “mapa.js”

**dibujarMultiLinea**

**Parámetros:** Array de puntos, layer (Objeto de tipo Vector), nombre.

**Resultados:** dibuja una **multilínea** en **layer** al cuál le asigna el **nombre** pasado como parámetro. Esta función no eliminar las figurar dibujadas anteriormente sobre **layer**.

**Llamadas:** desde “drawAditionalFeatures - mapa.js”

**dibujarCirculo**

**Parámetros:** x, y, layer (Objeto de tipo Vector), nombre.

**Resultados:** dibuja un **círculo** en **layer** al cuál le asigna el **nombre** pasado como parámetro. Esta función no eliminar las figurar dibujadas anteriormente sobre **layer**.

**Llamadas:** desde “drawAditionalFeatures - mapa.js”

**clearLayer**

**Parámetros:** layer (Objeto de tipo ol.layer.Vector).

**Resultados:** elimina todos los objetos Feature de **layer**. En otras palabras, elimina todas las figuras que existen en **layer**.

**Llamadas:** desde “drawAditionalFeatures - mapa.js”

**cerrarPopup**

**Parámetros:** Ninguno

**Resultados:** Cierra el popup (<div id = "info-popup"></div> en mapa.html).

**Llamadas:** desde todas las vistas cuyo nombre tiene la siguiente forma: info\_\*\*\*\*\*.html

**getFeatureInfoUrl**

**Parámetros:** layer (objeto de tipo ol.layer.Tile ), cordenadas (objeto ol.coordinate)

**Resultados:** Retorna la **URL** donde se halla la información del objeto (feature) de **layer** que se **intersecta** con las **coordenadas** proporcionadas. Para mayor detalle revisar la función.

**Llamadas:** desde “mapa.js”

**showLayerDataFromUrl**

**Parámetros:** layer (objeto de tipo ol.layer.Tile ),url, cordenadas (objeto ol.coordinate)

**Observaciones:** Hace una llamada a través de Ajax a **url** (proporcionada por getFeatureInfoUrl – mapa.js).

**Resultados:** Retorna la información de la **Url** en formato JSON y la muestra en un popup (<div id = "info-popup"></div>) en las **coordenadas** proporcionadas. Además dibuja figuras adicionales de acuerdo a cuál **capa gis** representa **layer**.

**Llamadas:** desde “mapa.js”

**showMessageOverPopup**

**Parámetros:** html (que puede ser contenido html o no), coordenadas (objeto ol.coordinate)

**Resultados:** Muestra **html** en un popup (<div id = "info-popup"></div>) en las **coordenadas** proporcionadas.

**Llamadas:** desde “showLayerDataFromUrl - mapa.js”

**renderDataFromLayer**

**Parámetros:** layer (objeto de tipo ol.layer.Tile ),data (datos en formato json)

**Observaciones:** Utiliza **nunjucks** para renderizar la **data** en la página html relacionada con **layer**. Para más detalle revisar la función.

**Resultados:** Retorna el contenido renderizado, es decir el contenido **html** con la **data** incrustada dentro de él.

**Llamadas:** desde “showLayerDataFromUrl - mapa.js”

**drawAditionalFeatures**

**Parámetros:** layer (objeto de tipo ol.layer.Tile ),data (datos en formato json)

**Observaciones:** dibuja las una figura (definida por **data**) de acuerdo a **layer**. Para más detalle revisar la función.

**Resultados:** Retorna el contenido renderizado, es decir el contenido **html** con la **data** incrustada dentro de él.

**Llamadas:** desde “showLayerDataFromUrl - mapa.js”

**ubicarCliente**

**Parámetros:** número de suministro

**Observaciones:** Hace una llamada a través de Ajax a / getCoordsCliente

**Resultados:** ubica al cliente en el mapa.

**Llamadas:** desde “buscar\_cliente.html”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| El cliente no existe | Utiliza showQueryResponse – mapa.js para mostrar un mensaje al cliente. |
| El número de inscripción no tiene el formato correcto, como resultado de la validación del lado del servidor. | Utiliza showQueryWarningResponse – mapa.js para mostrar un mensaje al cliente. |
| El número de inscripción no tiene el formato correcto, como resultado de la validación del lado del servidor. | Utiliza showQueryWarningResponse – mapa.js para mostrar un mensaje al cliente. |
|  |  |

**ubicarClientePorCodCat**

**Parámetros:** pre región, pre zona, pre sector, pre manzana, pre lote

**Observaciones:** Hace una llamada a través de Ajax a / getCoordsClienteByCodCatastral

**Resultados:** ubica al cliente en el mapa.

**Llamadas:** desde “buscar\_cliente\_por\_cod\_cat.html”

**Posibles acontecimientos**

|  |  |
| --- | --- |
| Condición | Resultado |
| El cliente no existe | Utiliza showQueryResponse – mapa.js para mostrar un mensaje al cliente. |
| El código catastral no tiene el formato correcto, como resultado de la validación del lado del servidor. | Utiliza showQueryWarningResponse – mapa.js para mostrar un mensaje al cliente. |
| El código catastral no tiene el formato correcto, como resultado de la validación del lado del servidor. | Utiliza showQueryWarningResponse – mapa.js para mostrar un mensaje al cliente. |
|  |  |

**verTotalClientesInscritos**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a / getCantClientesActivos

**Resultados:** muestra el total de clientes inscritos en el navegador.

**Llamadas:** desde “mapa.js”

**verTotalDistRedAgua**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a / distanciaTotalRedAgua

**Resultados:** muestra la distancia total de la red de agua en el navegador.

**Llamadas:** desde “mapa.js”

**verTotalDistRedDesague**

**Parámetros:** Ninguno

**Observaciones:** Hace una llamada a través de Ajax a / distanciaTotalRedAgua

**Resultados:** muestra la distancia total de la red de alcantarillado en el navegado.

**Llamadas:** desde “mapa.js”

**startLoadingAnimation**

**Parámetros:** Ninguno

**Observaciones:** Se utiliza para mostrar un gif y el mensaje: cargando…, cuando se inicia la búsqueda de un cliente.

**Resultados:** muestra un gif y el mensaje cargando...

**Llamadas:** desde “ubicarClientePorCodCat – mapa.js” y “ubicarCliente – mapa.js”

**stopLoadingAnimation**

**Parámetros:** Ninguno

**Observaciones:** Se utiliza para mostrar un gif y el mensaje: cargando…, cuando se inicia la búsqueda de un cliente.

**Resultados:** muestra un gif y el mensaje cargando...

**Llamadas:** desde “ubicarClientePorCodCat – mapa.js” y “ubicarCliente – mapa.js”

**showQueryResponse**

**Parámetros:** Mensaje

**Resultados:** muestra un mensaje en #myModal .msj

**Llamadas:** desde “ubicarClientePorCodCat – mapa.js” y “ubicarCliente – mapa.js”

**showQueryWarningResponse**

**Parámetros:** Mensaje

**Resultados:** muestra un mensaje en #myModal .msj con un estilo error-msj

**Llamadas:** desde “ubicarClientePorCodCat – mapa.js” y “ubicarCliente – mapa.js”

**verEnStreetViewHandler**

**Parámetros:** x, y

**Resultados:** muestra una vista panorámica del punto x, y (coordenadas en el sistema EPSG: 3857)

**Llamadas:** desde “info-cliente.html”