

ARQUITECTURA FRONT-END.

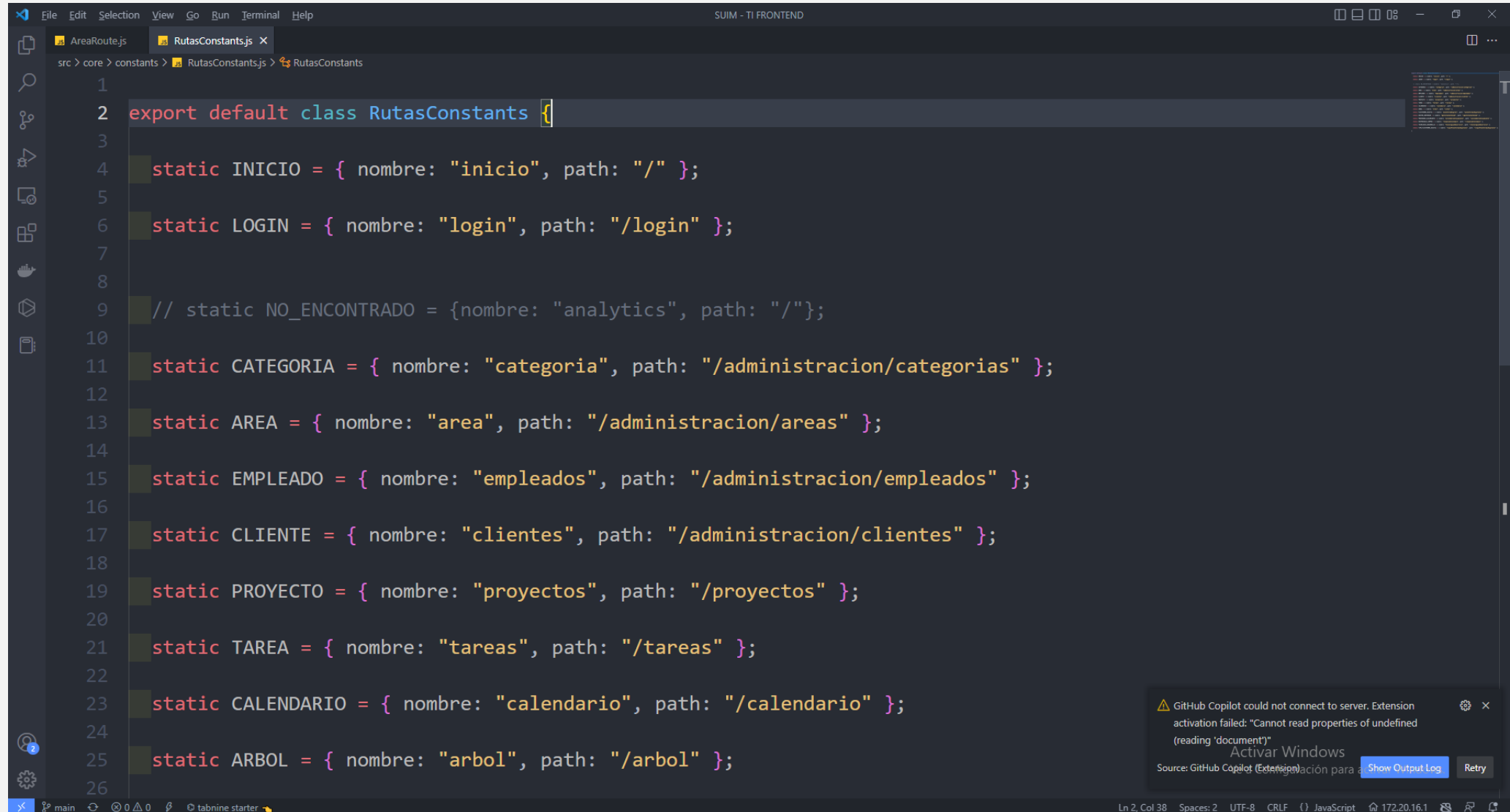
EJEMPLO DE CRUD BÁSICO.

1. CREACIÓN DE CONSTANTES PARA LAS RUTAS DEL BACK-END(CONEXIÓN AL SERVICIO) EN `URLSERVICES.JS`

```
/*  
 * SERVICIOS DE AREA  
 */  
static AREA_LISTAR_TODOS = API_PATH + "area/listarTodos";  
static AREA_LISTAR_PAGINADO = API_PATH + "area/listarPaginado";  
static AREA_REGISTRAR = API_PATH + "area/registrar";  
static AREA_OBTENER = API_PATH + "area/obtener";  
static AREA_ACTUALIZAR = API_PATH + "area/actualizar";  
static AREA_ELIMINAR = API_PATH + "area/eliminar";
```

2. CREACIÓN DE CONSTANTES PARA LAS RUTAS DEL FRONT

RUTASCONSTANS.JS



```
1
2 export default class RutasConstants {
3
4   static INICIO = { nombre: "inicio", path: "/" };
5
6   static LOGIN = { nombre: "login", path: "/login" };
7
8   // static NO_ENCONTRADO = {nombre: "analytics", path: "/"};
9
10  static CATEGORIA = { nombre: "categoria", path: "/administracion/categorias" };
11
12  static AREA = { nombre: "area", path: "/administracion/areas" };
13
14  static EMPLEADO = { nombre: "empleados", path: "/administracion/empleados" };
15
16  static CLIENTE = { nombre: "clientes", path: "/administracion/clientes" };
17
18  static PROYECTO = { nombre: "proyectos", path: "/proyectos" };
19
20  static TAREA = { nombre: "tareas", path: "/tareas" };
21
22  static CALENDARIO = { nombre: "calendario", path: "/calendario" };
23
24  static ARBOL = { nombre: "arbol", path: "/arbol" };
25
26 }
```

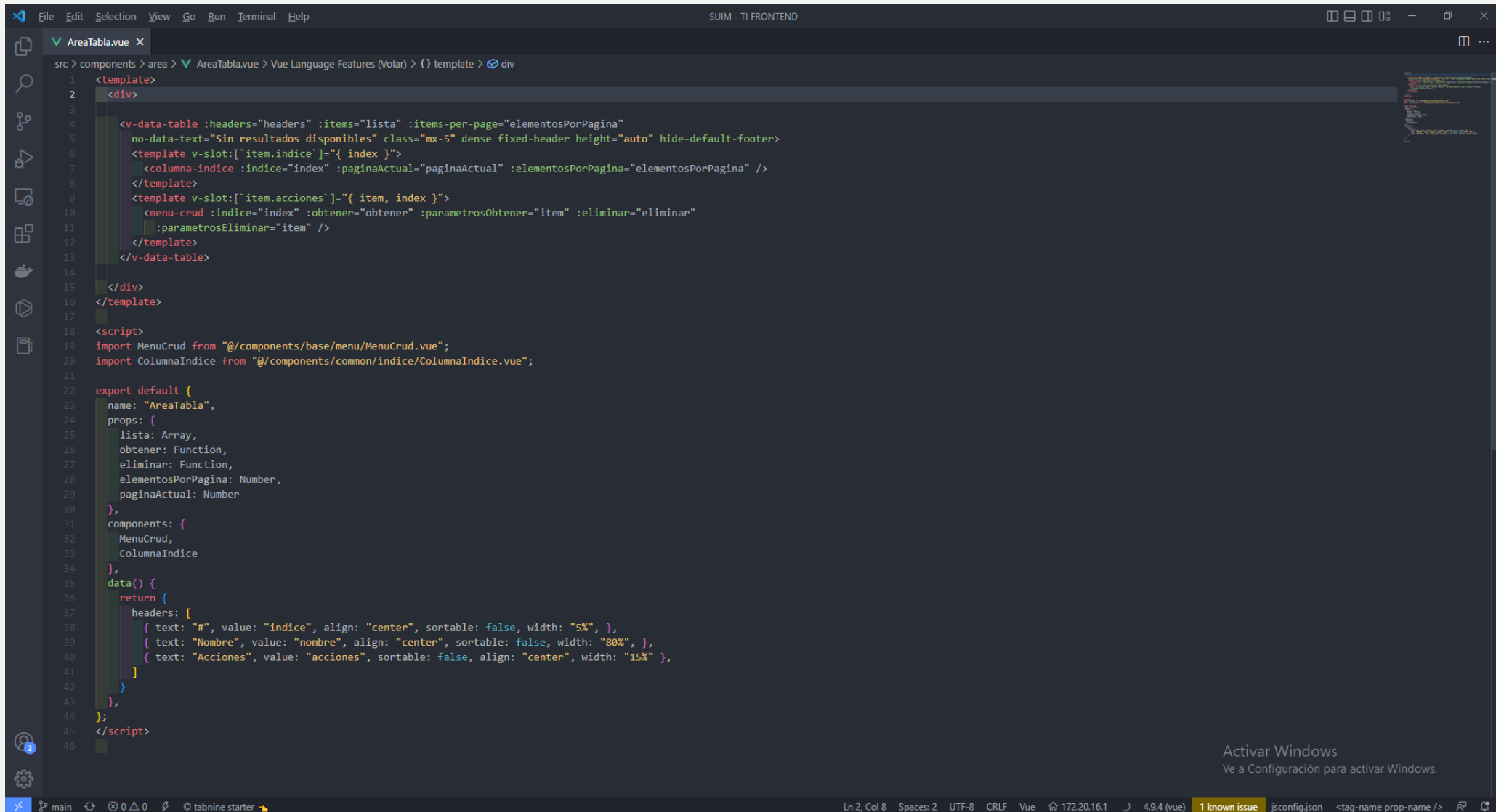
Ln 2, Col 38 Spaces: 2 UTF-8 CRLF JavaScript 172.20.16.1

GitHub Copilot could not connect to server. Extension activation failed: "Cannot read properties of undefined (reading 'document')"
Activar Windows
Source: GitHub Copilot (Extension) activación para: Show Output Log Retry

2. CREACIÓN DE SERVICE.JS (CONEXIÓN AL SERVICIO)

```
1 import UrlServices from "@core/constants/UrlServices";
2 import BaseService from "@core/services/BaseService";
3
4 export default class AreaService extends BaseService {
5
6     static listarPaginado(parametros) {
7         return this.POST_SEC(UrlServices.AREA_LISTAR_PAGINADO, parametros);
8     }
9
10    static listarTodos() {
11        return this.POST_SEC(UrlServices.AREA_LISTAR_TODOS);
12    }
13
14    static registrar(parametros) {
15        return this.POST_SEC(UrlServices.AREA_REGISTRAR, parametros);
16    }
17
18    static obtener(parametros) {
19        return this.POST_SEC(UrlServices.AREA_OBTENER, parametros);
20    }
21
22    static actualizar(parametros) {
23        return this.PUT_SEC(UrlServices.AREA_ACTUALIZAR, parametros);
24    }
25
26    static eliminar(parametros) {
27        return this.POST_SEC(UrlServices.AREA_ELIMINAR, parametros);
28    }
29
30 }
```

4. CREACIÓN DE TABLA(GENERACIÓN DE VISTA) COMPONENTE DENTRO DE LA CARPETA COMPONENTS/XXX/XXXTABLA.VUE

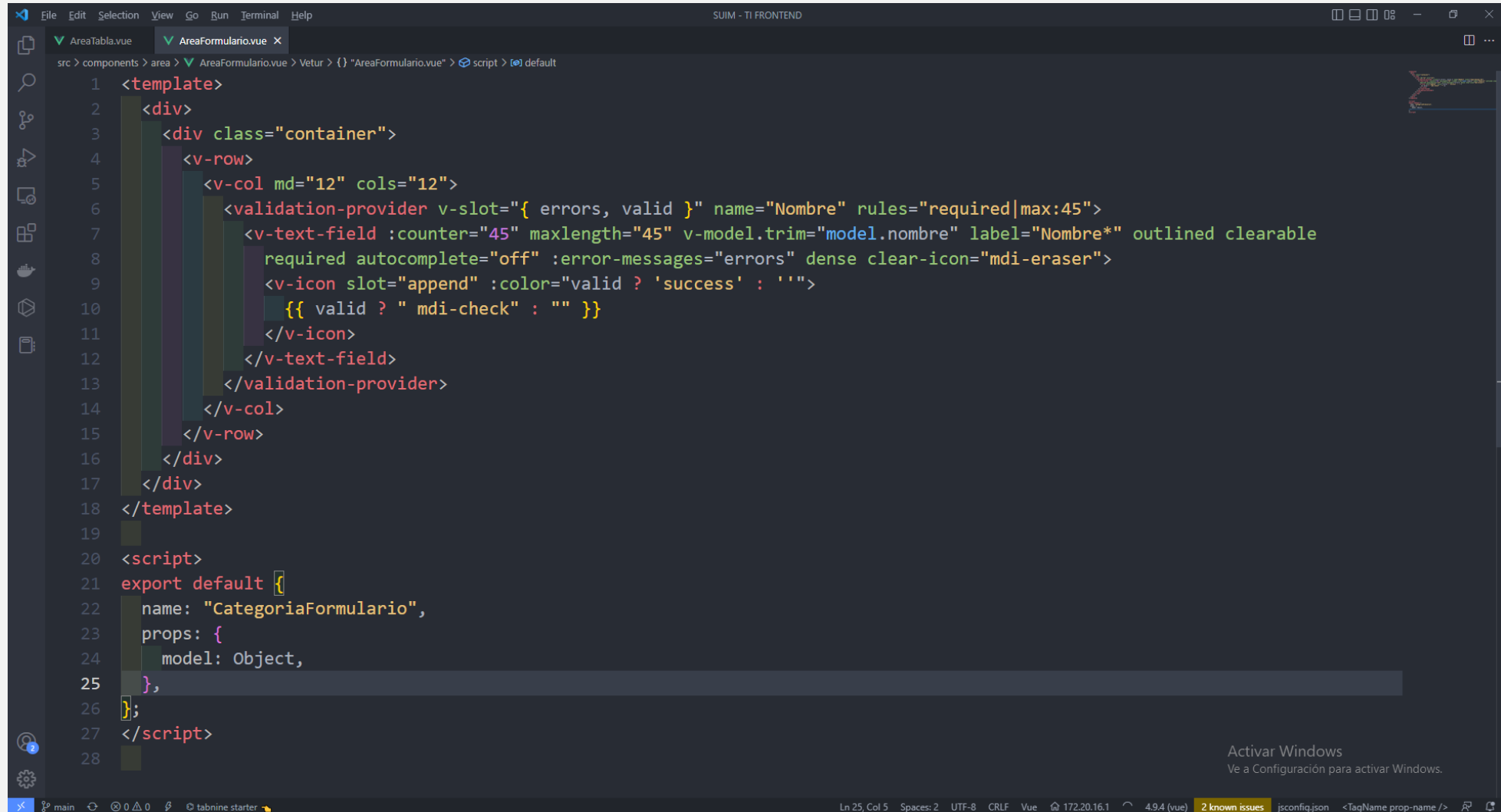


```
1 <template>
2   <div>
3
4     <v-data-table :headers="headers" :items="lista" :items-per-page="elementosPorPagina"
5       no-data-text="Sin resultados disponibles" class="mx-5" dense fixed-header height="auto" hide-default-footer>
6       <template v-slot:[`item.indice`]-="{ index }">
7         <columna-indice :indice="index" :paginaActual="paginaActual" :elementosPorPagina="elementosPorPagina" />
8       </template>
9       <template v-slot:[`item.acciones`]-="{ item, index }">
10        <menu-crud :indice="index" :obtener="obtener" :parametrosObtener="item" :eliminar="eliminar"
11          :parametrosEliminar="item" />
12      </template>
13    </v-data-table>
14  </div>
15</template>
16
17<script>
18import MenuCrud from "@components/base/menu/MenuCrud.vue";
19import ColumnaIndice from "@components/common/indice/ColumnaIndice.vue";
20
21export default {
22  name: "AreaTabla",
23  props: {
24    lista: Array,
25    obtener: Function,
26    eliminar: Function,
27    elementosPorPagina: Number,
28    paginaActual: Number
29  },
30  components: {
31    MenuCrud,
32    ColumnaIndice
33  },
34  data() {
35    return {
36      headers: [
37        { text: "#", value: "indice", align: "center", sortable: false, width: "5%", },
38        { text: "Nombre", value: "nombre", align: "center", sortable: false, width: "80%", },
39        { text: "Acciones", value: "acciones", sortable: false, align: "center", width: "15%", },
40      ]
41    }
42  },
43};
44</script>
```

Activar Windows
Ve a Configuración para activar Windows.

Ln 2, Col 8 Spaces: 2 UTF-8 CRLF Vue 172.20.16.1 4.9.4 (vue) 1 known issue jsconfig.json <tag-name prop-name />

5. CREACIÓN DE FORMULARIO(GENERACIÓN DE VISTA) COMPONENTE DENTRO DE LA CARPETA COMPONENTS/XXX/XXXFORMULARIO.VUE

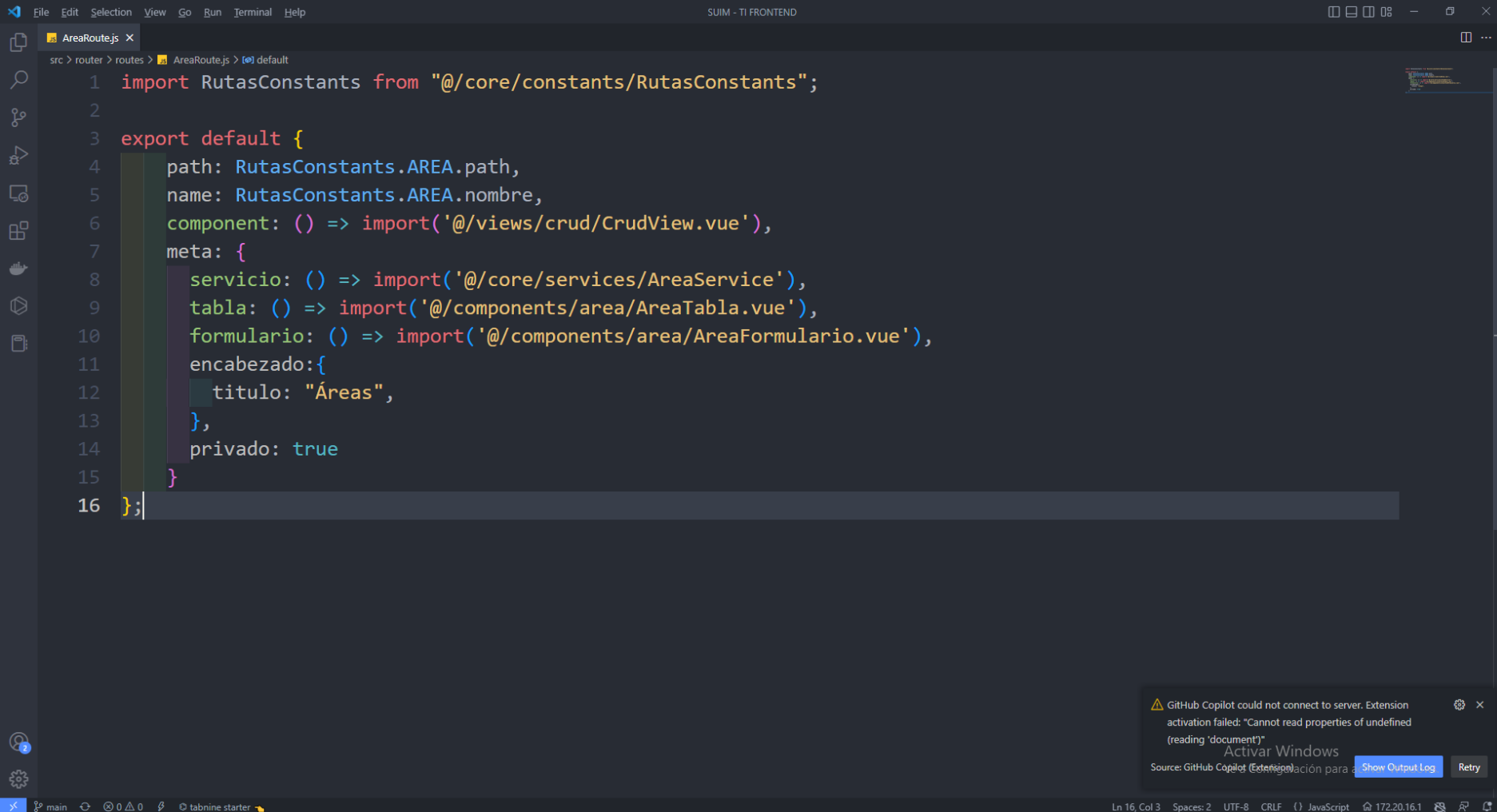


```
1 <template>
2   <div>
3     <div class="container">
4       <v-row>
5         <v-col md="12" cols="12">
6           <validation-provider v-slot="{ errors, valid }" name="Nombre" rules="required|max:45">
7             <v-text-field :counter="45" maxlength="45" v-model.trim="model.nombre" label="Nombre*" outlined clearable
8               required autocomplete="off" :error-messages="errors" dense clear-icon="mdi-eraser">
9               <v-icon slot="append" :color="valid ? 'success' : ''">
10                 {{ valid ? " mdi-check" : "" }}
11               </v-icon>
12             </v-text-field>
13           </validation-provider>
14         </v-col>
15       </v-row>
16     </div>
17   </div>
18 </template>
19
20 <script>
21 export default {
22   name: "CategoriaFormulario",
23   props: {
24     model: Object,
25   },
26 };
27 </script>
28
```

Activar Windows
Ve a Configuración para activar Windows.

6. CREACIÓN DE RUTA (GENERACIÓN ROUTE)

ARCHIVO QUE CONECTA TODOS LOS ARCHIVOS PREVIOS
ROUTER/ROUTES/XXXROUTE.JS



```
1 import RutasConstants from "@core/constants/RutasConstants";
2
3 export default {
4   path: RutasConstants.AREA.path,
5   name: RutasConstants.AREA.nombre,
6   component: () => import('@views/crud/CrudView.vue'),
7   meta: {
8     servicio: () => import('@core/services/AreaService'),
9     tabla: () => import('@components/area/AreaTabla.vue'),
10    formulario: () => import('@components/area/AreaFormulario.vue'),
11    encabezado: {
12      titulo: "Áreas",
13    },
14    privado: true
15  }
16 };
```

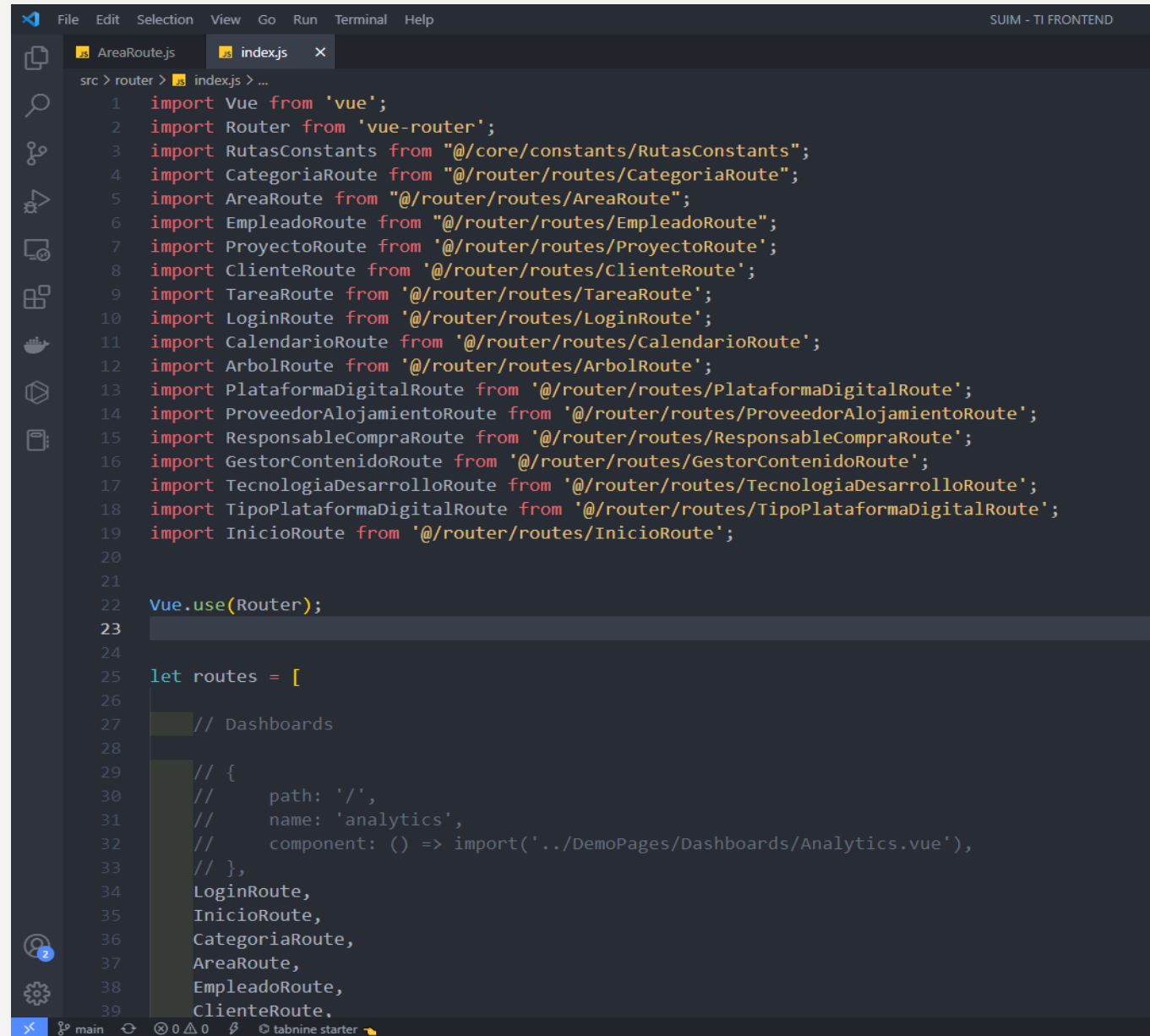
Ln 16, Col 3 Spaces: 2 UTF-8 CRLF JavaScript 172.20.16.1

GitHub Copilot could not connect to server. Extension activation failed: "Cannot read properties of undefined (reading 'document')"

Activar Windows

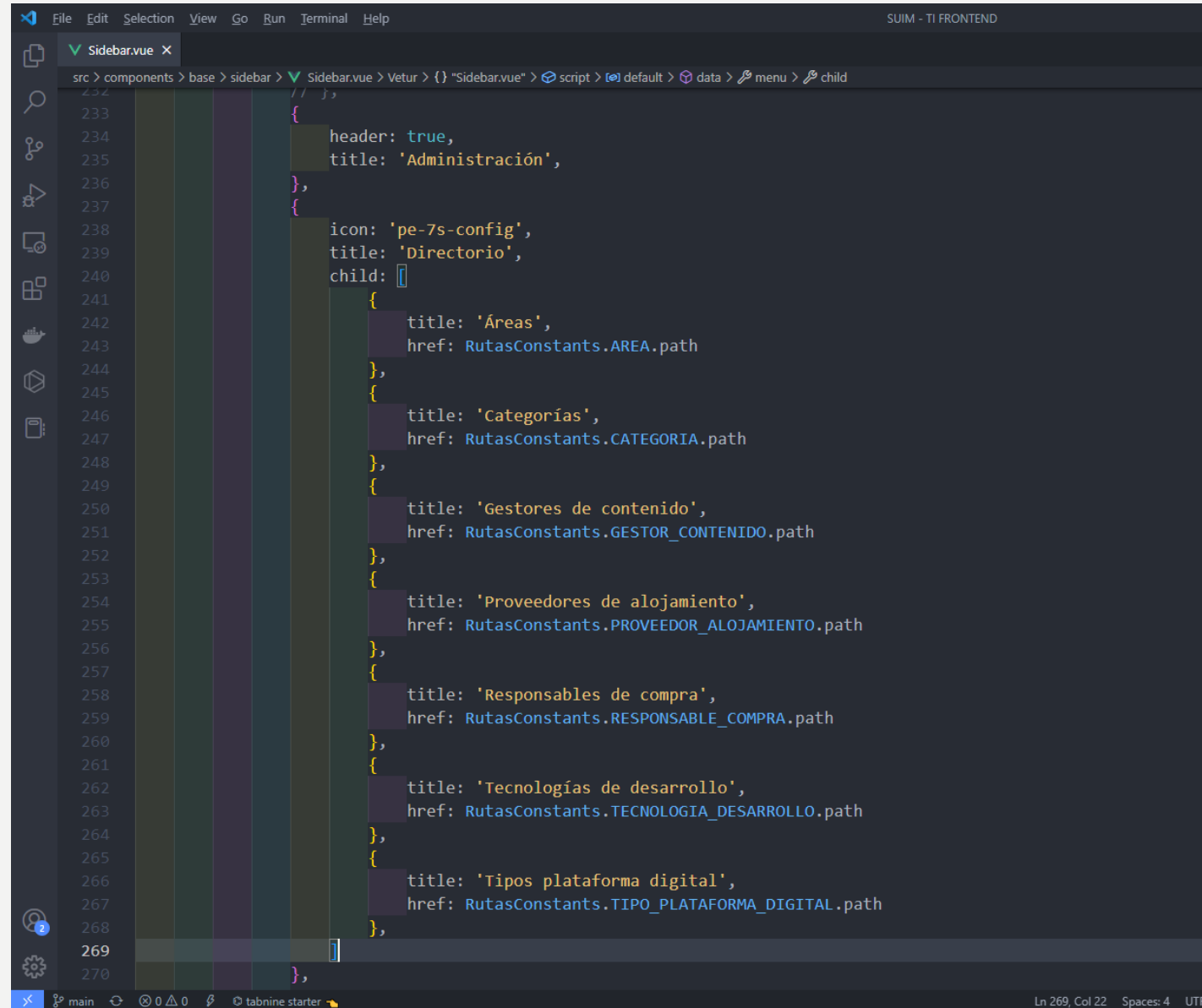
Source: GitHub Copilot (Extension) Activación para: Show Output Log Retry

7. IMPORTACIÓN DE RUTA EN ROUTER/INDEX.JS (GENERACIÓN ROUTE)



```
File Edit Selection View Go Run Terminal Help SUIM - TI FRONTEND
src > router > index.js > ...
1 import Vue from 'vue';
2 import Router from 'vue-router';
3 import RutasConstants from "@core/constants/RutasConstants";
4 import CategoriaRoute from "@router/routes/CategoriaRoute";
5 import AreaRoute from "@router/routes/AreaRoute";
6 import EmpleadoRoute from "@router/routes/EmpleadoRoute";
7 import ProyectoRoute from "@router/routes/ProyectoRoute";
8 import ClienteRoute from "@router/routes/ClienteRoute";
9 import TareaRoute from "@router/routes/TareaRoute";
10 import LoginRoute from "@router/routes/LoginRoute";
11 import CalendarioRoute from "@router/routes/CalendarioRoute";
12 import ArbolRoute from "@router/routes/ArbolRoute";
13 import PlataformaDigitalRoute from "@router/routes/PlataformaDigitalRoute";
14 import ProveedorAlojamientoRoute from "@router/routes/ProveedorAlojamientoRoute";
15 import ResponsableCompraRoute from "@router/routes/ResponsableCompraRoute";
16 import GestorContenidoRoute from "@router/routes/GestorContenidoRoute";
17 import TecnologiaDesarrolloRoute from "@router/routes/TecnologiaDesarrolloRoute";
18 import TipoPlataformaDigitalRoute from "@router/routes/TipoPlataformaDigitalRoute";
19 import InicioRoute from "@router/routes/InicioRoute";
20
21
22 Vue.use(Router);
23
24
25 let routes = [
26   // Dashboards
27   // {
28   //   path: '/',
29   //   name: 'analytics',
30   //   component: () => import('../DemoPages/Dashboards/Analytics.vue'),
31   // },
32   LoginRoute,
33   InicioRoute,
34   CategoriaRoute,
35   AreaRoute,
36   EmpleadoRoute,
37   ClienteRoute,
```


8. ENLACE DE REDIRECCIÓN A LA VISTA (PRUEBA DE FUNCIONALIDAD) SIDEBAR.VUE



```
233 {
234   header: true,
235   title: 'Administración',
236 },
237 {
238   icon: 'pe-7s-config',
239   title: 'Directorio',
240   child: [
241     {
242       title: 'Áreas',
243       href: RutasConstants.AREA.path
244     },
245     {
246       title: 'Categorías',
247       href: RutasConstants.CATEGORIA.path
248     },
249     {
250       title: 'Gestores de contenido',
251       href: RutasConstants.GESTOR_CONTENIDO.path
252     },
253     {
254       title: 'Proveedores de alojamiento',
255       href: RutasConstants.PROVEEDOR_ALOJAMIENTO.path
256     },
257     {
258       title: 'Responsables de compra',
259       href: RutasConstants.RESPONSABLE_COMPRA.path
260     },
261     {
262       title: 'Tecnologías de desarrollo',
263       href: RutasConstants.TECNOLOGIA_DESARROLLO.path
264     },
265     {
266       title: 'Tipos plataforma digital',
267       href: RutasConstants.TIPO_PLATAFORMA_DIGITAL.path
268     },
269   ],
270 }
```



EXTRAS

COMPONENTES BASE.

1.1. CRUDVIEW.VUE; COMPONENTES Y VARIABLES

```
139 export default {
140   name: "CrudView",
141   components: {
142     TituloPagina,
143     Loader
144   },
145   data() {
146     return {
147       titulo: "Sin titulo",
148       subtitulo: "",
149       busqueda: "",
150       accion: 1,
151       enviando: false,
152       dialogo: false,
153       indice: -1,
154       accion: 1,
155       lista: [],
156       model: {},
157       tabla: null,
158       servicio: null,
159       formulario: null,
160       busqueda: "",
161       total: 0,
162       paginaActual: 1,
163       elementosPorPagina: 10,
164       retrasoBusqueda: null,
165       conteoPagina: 1,
166       ocultarBotoAgregar: false
167     }
168   },
```

1.2. CRUDVIEW.VUE; EVENTOS

```
watch: {  
  busca(nuevoValor, anteriorValor) {  
    this.eliminarRetrasoBusqueda();  
    this.retrasoBusqueda = setTimeout(() => {  
      this.paginaActual = 1;  
      this.listarPaginado();  
    }, 400);  
  },  
  paginaActual() {  
    this.listarPaginado();  
  }  
},  
async mounted() {  
  this.construirVista();  
},
```

1.3. CRUDVIEW.VUE; MÉTODOS DE LÓGICA DE COMPONENTE

```
methods: {
  async construirVista() {
    this.titulo = this.$route?.meta?.encabezado?.titulo || "Sin titulo";
    this.subtitulo = this.$route?.meta?.encabezado?.subtitulo || "";
    this.servicio = (await this.$route.meta.servicio())?.default || null;
    this.tabla = (await this.$route.meta.tabla())?.default || null;
    this.formulario = (await this.$route.meta.formulario())?.default || null;
    this.ocultarBotoAgregar = this.$route.meta.ocultarBotoAgregar || false;
    this.listarPaginado();
  },
  mostrarAccion(accion) {
    return this.accion == accion;
  },
  cambiarAccion(accionNueva) {
    if (accionNueva == 1) {
      this.model = {};
      if (this.$route?.meta?.modeloInicial != undefined) {
        this.model = Object.assign(this.$route?.meta?.modeloInicial, {});
      }
      this.$refs.observer.reset();
      this.accion = accionNueva;
    } else {
      if (accionNueva == 2 && this.$route?.meta?.modeloInicial != undefined) {
        this.model = this.$route?.meta?.modeloInicial();
      }
      this.accion = accionNueva;
    }
  },
  hacerAccion() {
    switch (this.accion) {
      case 2:
        this.registrar();
        break;
      case 3:
        this.actualizar();
        break;
      default:
        break;
    }
  },
  cambiarPagina(page) {
    if (this.paginaActual != page) {
      this.paginaActual = page;
      this.listarPaginado();
    }
  },
  eliminarRetrasoBusqueda() {
    if (this.retrasoBusqueda != null) {
      clearTimeout(this.retrasoBusqueda);
      this.retrasoBusqueda = null;
    }
  },
}
```

1.4. CRUDVIEW.VUE; MÉTODOS DEL CRUD

```
listarPaginado() { ...  
},  
registrar() { ...  
},  
actualizar() { ...  
},  
obtener(parametros, i) { ...  
},  
eliminar(parametros, i) { ...  
},  
actualizarEstado(parametros, i) { ...  
},
```

1.5. CRUDVIEW.VUE; MÉTODO DE LISTADO PAGINADO

```
236     listarPaginado() {
237         if (!this.enviando) {
238
239             this.enviando = true;
240
241             let parametros = {
242                 busqueda: this.busqueda,
243                 maximoResultados: this.elementosPorPagina,
244                 paginaActual: this.paginaActual
245             };
246
247             this.servicio.listarPaginado(parametros).then((res) => {
248
249                 let { elementos, total } = res.data;
250                 this.lista = elementos;
251                 this.total = total;
252                 this.conteoPagina = Math.ceil(total / this.elementosPorPagina);
253
254             }).catch((e) => {
255                 this.$utileria.errorhttp(e);
256             }).finally(()=>{
257                 this.enviando = false;
258             });
259         }
260     },
```

1.6. CRUDVIEW.VUE; MÉTODO DE REGISTRO

```
261 registrar() {
262
263     if (!this.enviando) {
264         this.enviando = true;
265         this.servicio.registrar(this.model).then((response) => {
266             this.enviando = false;
267             let { data, status } = response;
268             if (status == 200) {
269                 this.listarPaginado();
270                 this.$utileria.notificacion("Se ha registrado correctamente.", "SUCCESS");
271                 this.cambiarAccion(1);
272             } else {
273                 data = data || "Ocurrió un error durante el registro.";
274                 this.$utileria.notificacion(data, "ERROR");
275             }
276         }).catch((e) => {
277             this.enviando = false;
278             this.$utileria.errorhttp(e);
279         });
280     }
281
282 },
```


1.7. CRUDVIEW.VUE; MÉTODO DE CONSULTA INDIVIDUAL

```
311  ✓ obtener(parametros, i) {
312
313  ✓  if (!this.enviando) {
314
315    this.enviando = true;
316
317  ✓    this.servicio.obtener(parametros).then((response) => {
318      this.enviando = false;
319      let { data, status } = response;
320  ✓    if (status == 200) {
321      this.model = data;
322      this.index = Number(i);
323      this.cambiarAccion(3);
324  ✓    } else {
325      data = data || "Ocurrió un error al obtener la información.";
326      this.$utileria.notificacion(data, "ERROR");
327    }
328    }).catch((e) => {
329      this.enviando = false;
330      this.$utileria.errorhttp(e);
331    });
332
333  }
334
335  },
```

1.8. CRUDVIEW.VUE; MÉTODO DE MODIFICACIÓN

```
283  actualizar() {
284
285    if (!this.enviando) {
286
287      this.enviando = true;
288
289      this.servicio.actualizar(this.model).then((response) => {
290        this.enviando = false;
291        let { data, status } = response;
292        if (status == 200) {
293
294          this.listarPaginado();
295          this.$utileria.notificacion("Se ha modificado correctamente.", "SUCCESS");
296          this.cambiarAccion(1);
297          this.index = -1;
298
299        } else {
300          data = data || "Ocurrió un error durante la modificación.";
301          this.$utileria.notificacion(data, "ERROR");
302        }
303
304      }).catch((e) => {
305        this.enviando = false;
306        this.$utileria.errorhttp(e);
307      });
308    }
309
310  },
```

1.9. CRUDVIEW.VUE; MÉTODO DE ELIMINACIÓN

```
    },
    eliminar(parametros, i) {
      if (!this.enviando) {
        this.enviando = true;

        this.servicio.eliminar(parametros).then((response) => {
          this.enviando = false;
          let { data, status } = response;
          if (status == 200) {
            this.listarPaginado();
            this.$utileria.notificacion("Se ha eliminado correctamente.", "SUCCESS");
          } else {
            data = data || "Ocurrió un error al eliminar la información.";
            this.$utileria.notificacion(data, "ERROR");
          }
        }).catch((e) => {
          this.enviando = false;
          this.$utileria.errorhttp(e);
        });
      }
    },
  },
```

1.10. CRUDVIEW.VUE; MÉTODO DE CAMBIO DE ESTATUS

```
358  ✓ actualizarEstado(parametros, i) {
359  ✓    if (!this.enviando) {
360
361    this.enviando = true;
362
363  ✓    this.servicio.actualizarEstado(parametros).then((response) => {
364    this.enviando = false;
365    let { data, status } = response;
366  ✓    if (status == 200) {
367    this.listarPaginado();
368    this.$utileria.notificacion("Se ha actualizado correctamente.", "SUCCESS");
369  ✓    } else {
370    data = data || "Ocurrió un error al eliminar la información.";
371    this.$utileria.notificacion(data, "ERROR");
372    }
373    }).catch((e) => {
374    this.enviando = false;
375    this.$utileria.errorhttp(e);
376    });
377
378    }
379  },
```