

**INFORME QUE RINDE LA UNIDAD TÉCNICA DE SERVICIOS DE  
INFORMÁTICA PARA DAR CUMPLIMIENTO A LO ESTABLECIDO EN  
EL PUNTO QUINTO DEL ACUERDO OPLEV/CG095/2023 EMITIDO  
POR EL CONSEJO GENERAL DEL ORGANISMO PÚBLICO LOCAL  
ELECTORAL DEL ESTADO DE VERACRUZ.**





El 16 de agosto del 2023, el Consejo General del OPLE Veracruz mediante el acuerdo OPLEV/CG095/2023 aprobó modificar el plan de trabajo de la Unidad de Fiscalización para la revisión del primer informe semestral de avance del ejercicio 2023, de las asociaciones políticas estatales con registro ante el Organismo Público Local Electoral del estado de Veracruz.

En el punto Quinto del Acuerdo referido, se instruyó a la Unidad Técnica de Servicios de Informática realizar *“las acciones necesarias para determinar el uso de dicho Sistema durante el segundo semestre del año en curso, para lo cual dichas áreas deberán informar los resultados correspondientes a más tardar cuatro días hábiles previos al 31 de agosto de 2023, fecha en la que el Consejo General determinará lo conducente.”*

Atento a lo anterior, me permito informar que dada la complejidad técnica que implica realizar un análisis detallado de procesos y flujo de información del Sistema de Contabilidad en Línea (SiCLi) no es técnicamente posible en el tiempo señalado para garantizar su correcto funcionamiento en todos y cada uno de sus procesos para el segundo semestre del año 2023, lo anterior derivado principalmente de los siguientes motivos:

- a) La complejidad del proceso contable aplicado en el SiCLi;
- b) La cantidad de líneas de código que lo conforman; y
- c) Los requerimientos de infraestructura, de procesamiento de datos y almacenamiento.

Aunado a lo anterior, se debe tener en consideración que los trabajos de programación del Sistema de Contabilidad en Línea iniciaron desde el año 2018, y han trabajado en la generación de código diferentes personas en distintos lapsos de tiempo, situación que vuelve más complejo saber los *estilos* o formas de trabajo con los que se elaboraron los diferentes módulos y sub-módulos con los que cuenta el Sistema Informático.

Dicha complejidad se suma al inminente Inicio del Proceso Electoral Local Ordinario 2023-2024, y la demanda de recursos humanos y materiales que se ocuparán para atender las obligaciones de este Órgano Electoral y aún más importante, para garantizar a la ciudadanía veracruzana el libre ejercicio de sus derechos político-electorales.





Para abundar en lo anterior, es oportuno detallar los siguientes aspectos:

**a) Líneas de código que conforman el Sistema de Contabilidad en Línea**

El SiCLi cuenta con 39,077 líneas de código mismas que se encuentran distribuidas de la siguiente manera:

Tipo	Líneas de código	Descripción
Controladores:	16,037	Módulos de programación que permiten la interacción de información entre las tablas de la bases de datos del sistema y las páginas web del sistema (Vistas, Partial views, Layouts y Mails)
Clases	731	Una clase es un elemento de la programación orientada a objetos que actúa como una plantilla y va a definir las características y comportamientos de una entidad. La clase va a ser como un molde a partir del cual vamos a poder definir entidades, las clases afectan directamente a los controladores, un controlador para operar puede emplear varias clases.
Vistas	13,755	Las vistas son la parte pública que el usuario de nuestro sistema va a poder ver, conocidas como las páginas web.
Partial views	7,134	Una vista parcial permite agrupar un trozo de código de HTML, CSS etc. que pueda ser reutilizado en otras vistas.
Layouts	328	Es una plantilla general que podemos heredar atributos en otros sitios y evitar repetir código, evitando la redundancia.
Mails	1,092	Código conformado por instrucciones que se utilizan para generar y enviar los correos electrónicos que el sistema debe reportar en determinados procesos.

Por lo anterior, para realizar una correcta revisión que otorgue como resultado tener la certeza de la correcta funcionalidad del SiCLi, es necesario poder revisar cada una de las líneas de código (39,077) y procesos de programación involucrados para cada uno de los procedimientos que se realicen dentro de él.

Así mismo, se requiere poder dar un seguimiento de acción en tiempo real de los módulos que comprenden este sistema, es decir, se tendría que abrir el código y validar que el flujo sea congruente en cada línea del código, y dar seguimiento de flujo en una vista usuario y programador, vigilando el comportamiento de sistema, así como observar el procedimiento de almacenamiento temporal y permanente, y verificar la creación de las variables temporales y de registros definitivos.

#### **b) Complejidad del Sistema de Contabilidad en Línea**

Solamente la parte visual del SiCLi, que es la que utiliza el usuario final, está conformado por 7 módulos y 27 submódulos los cuales requieren de conocimiento contable especializado para su programación y en consecuencia su análisis o revisión.

Es decir, se requiere que un Ingeniero o programador lea y comprenda todos los detalles técnicos contables que involucran los procedimientos que realiza en SiCLi, o en caso opuesto, se requiere que un experto en contabilidad aprenda el lenguaje de programación para realizar una auditoría al sistema y detectar alguna incidencia con ambos conocimientos.

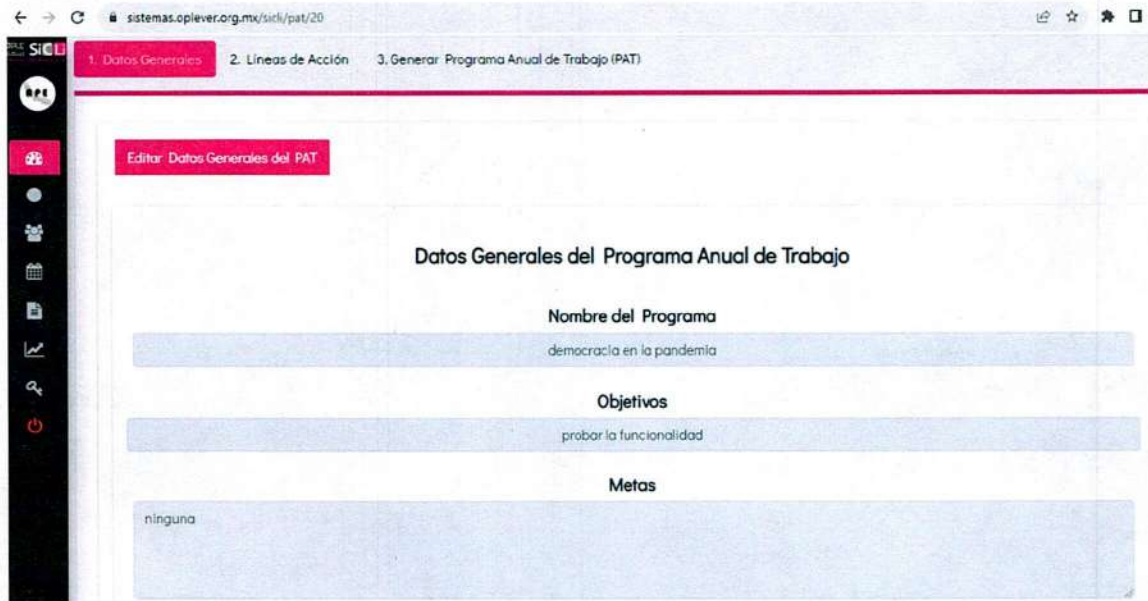
Por otra parte, debe considerarse también que no es posible cuantificar, de forma precisa, el almacenamiento que puede requerirse en el transcurso de un año contable, ya que las Asociaciones Políticas Estatales ingresan su documentación con diferentes características y cantidades (ejemplos: resolución, tipo de documento, cantidad de documentos, etc.) ocasionando que sea muy variable el espacio requerido en el repositorio de Sistema.

Ejemplos de algunos procesos y flujos de información complejos con los que cuenta el Sistema:



PAT

1. Deben capturarse los datos generales



Editar Datos Generales del PAT

### Datos Generales del Programa Anual de Trabajo

**Nombre del Programa**  
democracia en la pandemia

**Objetivos**  
probar la funcionalidad

**Metas**  
ninguna

2. Captura de Líneas de acción



Agregar Línea de Acción

### Líneas de Acción

Mostrar 10 registros

Excel Buscar:

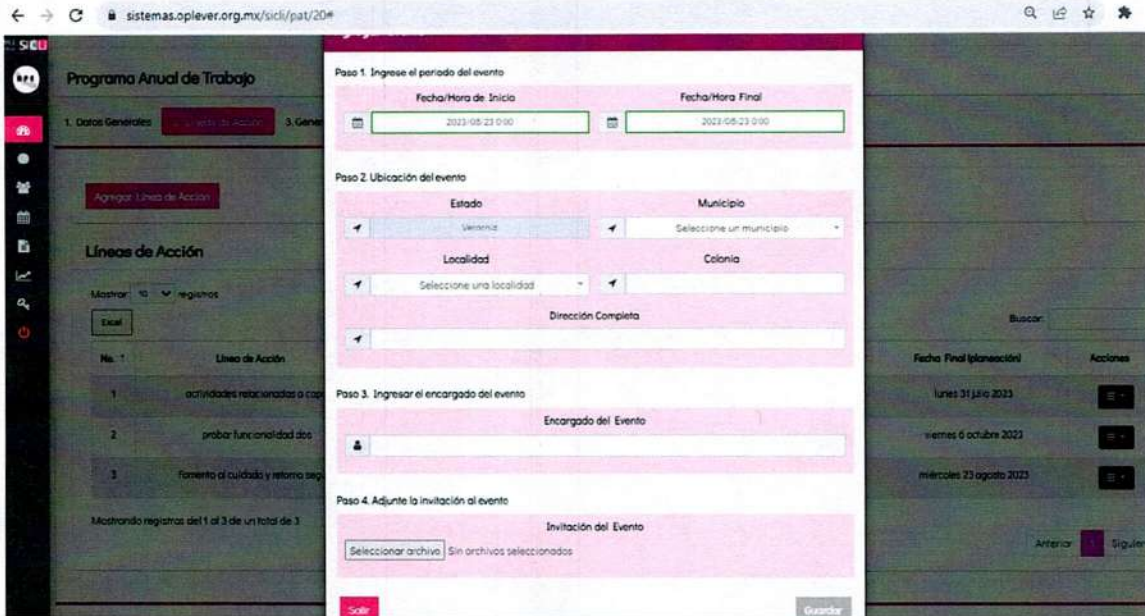
No. 1	Línea de Acción	Monto	Rubro	Fecha de Inicio (planeación)	Fecha Final (planeación)	Acciones
1	actividades relacionadas a capa	\$ 5,458.88	Educación y Capacitación	Jueves 13 Julio 2023	lunes 31 Julio 2023	☰
2	probar funcionalidad dos	\$ 5,600.00	Gastos por Tareas Editoriales	domingo 1 octubre 2023	viernes 6 octubre 2023	☰
3	Fomento al cuidado y retorno seguro	\$ 1.00	Educación y Capacitación	sábado 1 Julio 2023	miércoles 23 agosto 2023	☰

Mostrando registros del 1 al 3 de un total de 3

Anterior **1** Siguiente



### 3. Captura de eventos y evidencias



**Programa Anual de Trabajo**

1. Datos Generales

2. Ubicación del Evento

3. Generar

Agregar Línea de Acción

**Líneas de Acción**

Mostrar: 10 registros

Exact

No. 1	Línea de Acción
1	actividades relacionadas a capa
2	probar funcionalidad dos
3	fomento al cuidado y retorno seg

Mostrando registros del 1 al 3 de un total de 3

**Paso 1. Ingrese el periodo del evento**

Fecha/Hora de Inicio: 2023-08-23 0:00

Fecha/Hora Final: 2023-08-23 0:00

**Paso 2. Ubicación del evento**

Estado: Veracruz

Municipio: Seleccione un municipio

Localidad: Seleccione una localidad

Colonia: Seleccione una colonia

Dirección Completa: Seleccione una dirección

**Paso 3. Ingresar el encargado del evento**

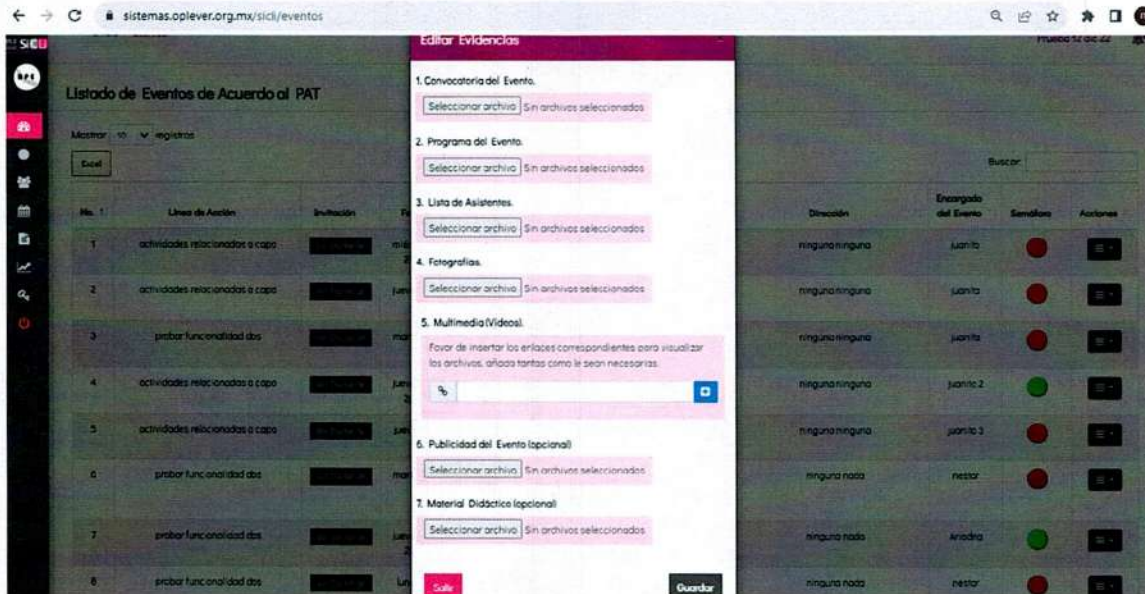
Encargado del Evento: Seleccione un encargado

**Paso 4. Adjunte la invitación al evento**

Invitación del Evento: Seleccione un archivo

Sin archivos seleccionados

Guardar



**Editar Evidencias**

1. Convocatoria del Evento. Seleccione un archivo. Sin archivos seleccionados

2. Programa del Evento. Seleccione un archivo. Sin archivos seleccionados

3. Lista de Asistentes. Seleccione un archivo. Sin archivos seleccionados

4. Fotografías. Seleccione un archivo. Sin archivos seleccionados

5. Multimedia (Videos). Favor de insertar los enlaces correspondientes para visualizar los archivos, añada tantas como le sean necesarias.

6. Publicidad del Evento (opcional). Seleccione un archivo. Sin archivos seleccionados

7. Material Didáctico (opcional). Seleccione un archivo. Sin archivos seleccionados

Sin archivos seleccionados

Guardar

**Listado de Eventos de Acuerdo al PAT**

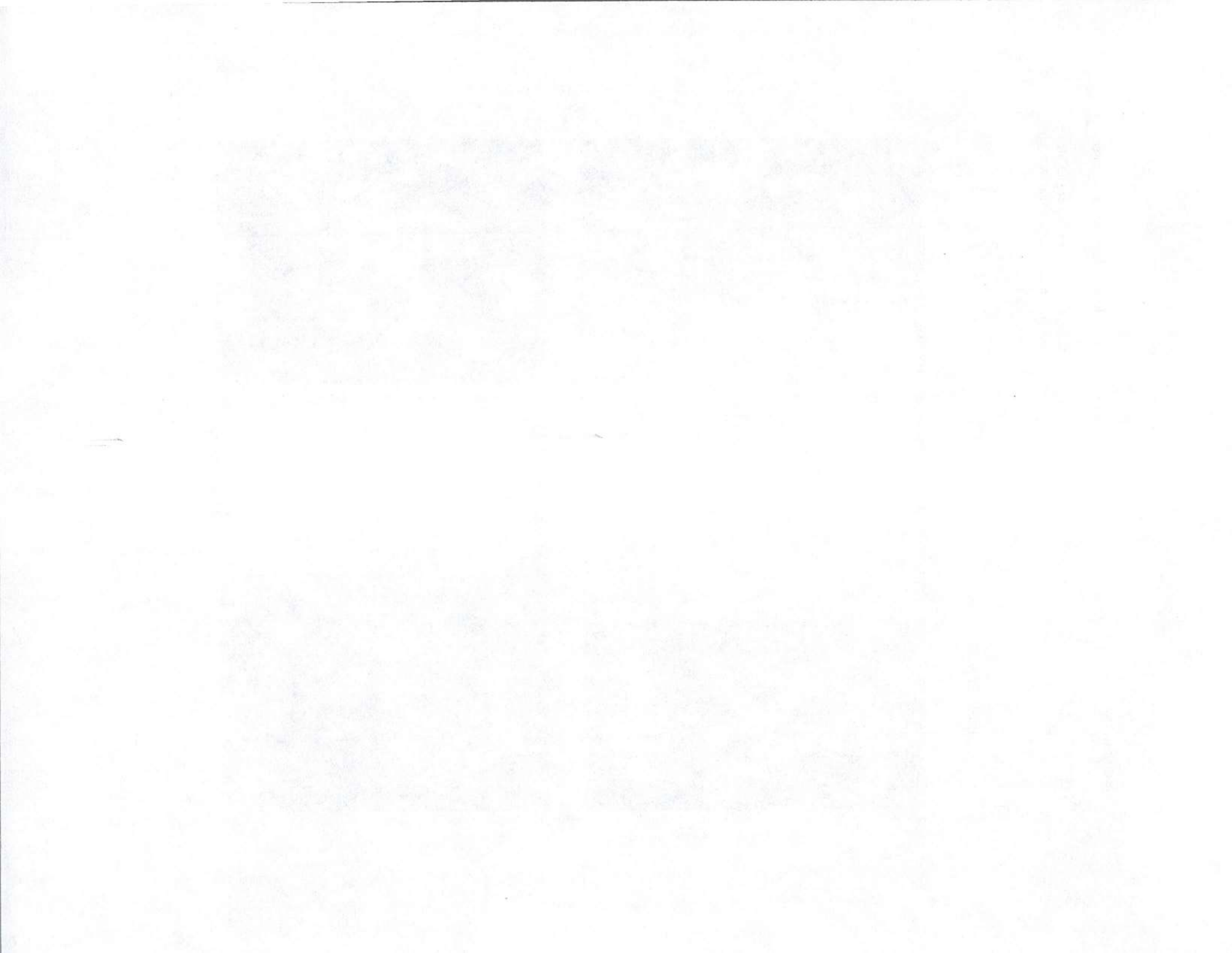
Mostrar: 10 registros

Exact

No. 1	Línea de Acción	Invitación	Encargado	Semiforo	Acciones
1	actividades relacionadas a capa		ninguna/ninguna	Juanita	
2	actividades relacionadas a capa		ninguna/ninguna	Juanita	
3	probar funcionalidad dos		ninguna/ninguna	Juanita	
4	actividades relacionadas a capa		ninguna/ninguna	Juanita 2	
5	actividades relacionadas a capa		ninguna/ninguna	Juanita 3	
6	probar funcionalidad dos		ninguna nada	nestor	
7	probar funcionalidad dos		ninguna nada	Aradra	
8	probar funcionalidad dos		ninguna nada	nestor	





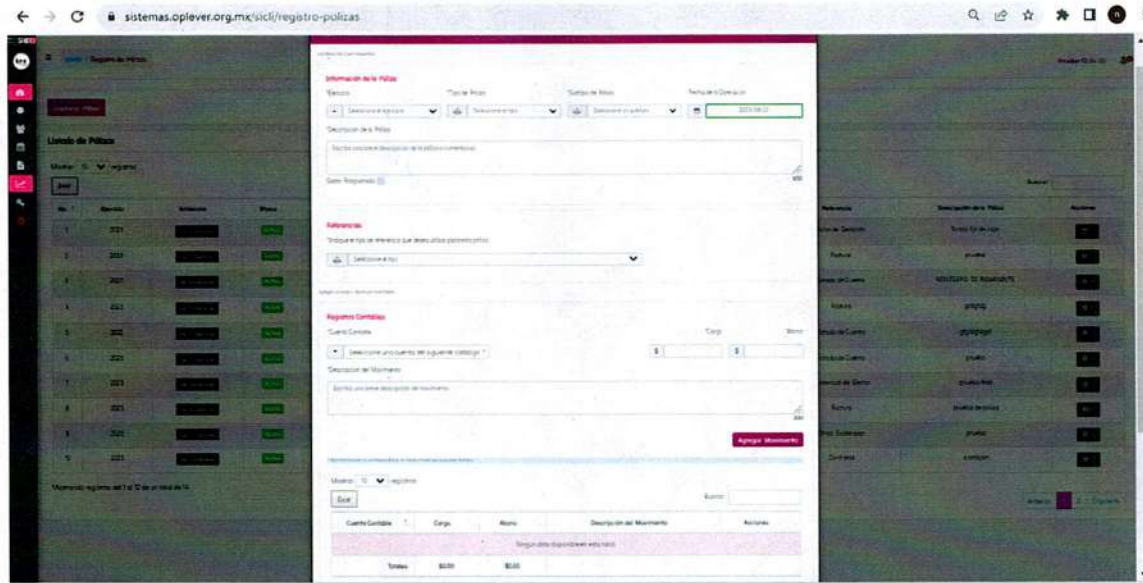


4. Generación de cronograma

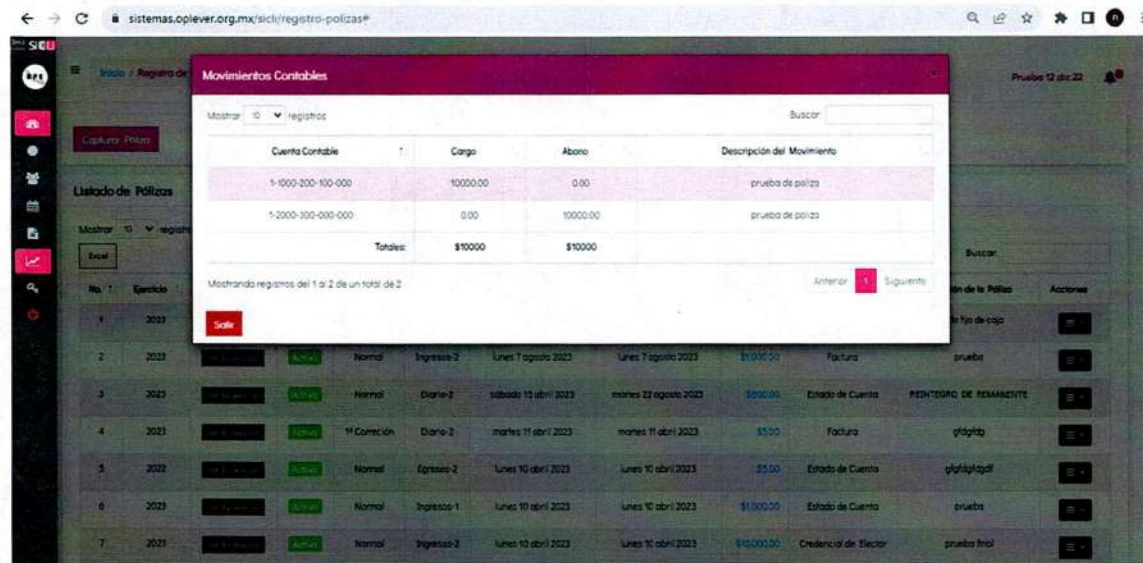
<b>PROGRAMA ANUAL DE TRABAJO</b>				
				<b>FORMATO: PAT</b>
				<b>EJERCICIO: 2023</b>
<b>NOMBRE DE LA ASOCIACIÓN POLÍTICA ESTATAL</b>				
Prueba 12 dic 22				
<b>NOMBRE DEL PROGRAMA</b>				
democracia en la pandemia				
<b>MISIÓN</b>				
Prueba UTSI				
<b>VISIÓN</b>				
Prueba				
<b>OBJETIVOS</b>				
probar la funcionalidad				
<b>METAS</b>				
ninguna				
LÍNEAS DE ACCIÓN	PERIODO		MONTO	PRESUPUESTO
	Inicio	Fin		
actividades relacionadas a capa	13/07/2023	31/07/2023	\$5,458.88	Educación y Capacitación
actividades relacionadas a capa	13/07/2023	31/07/2023	\$5,458.88	Educación y Capacitación
actividades relacionadas a capa	13/07/2023	31/07/2023	\$5,458.88	Educación y Capacitación
probar funcionalidad dos	01/10/2023	06/10/2023	\$5,800.00	Gastos por Tareas Editoriales
probar funcionalidad dos	01/10/2023	06/10/2023	\$5,800.00	Gastos por Tareas Editoriales
probar funcionalidad dos	01/10/2023	06/10/2023	\$5,800.00	Gastos por Tareas Editoriales
probar	01/10/2023	06/10/2023	\$5,800.00	Gastos por Tareas



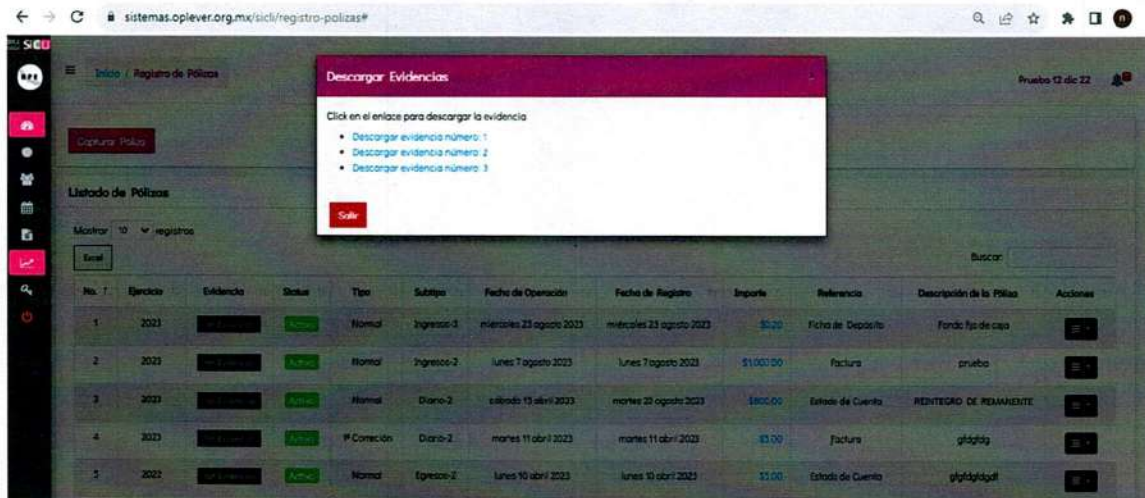
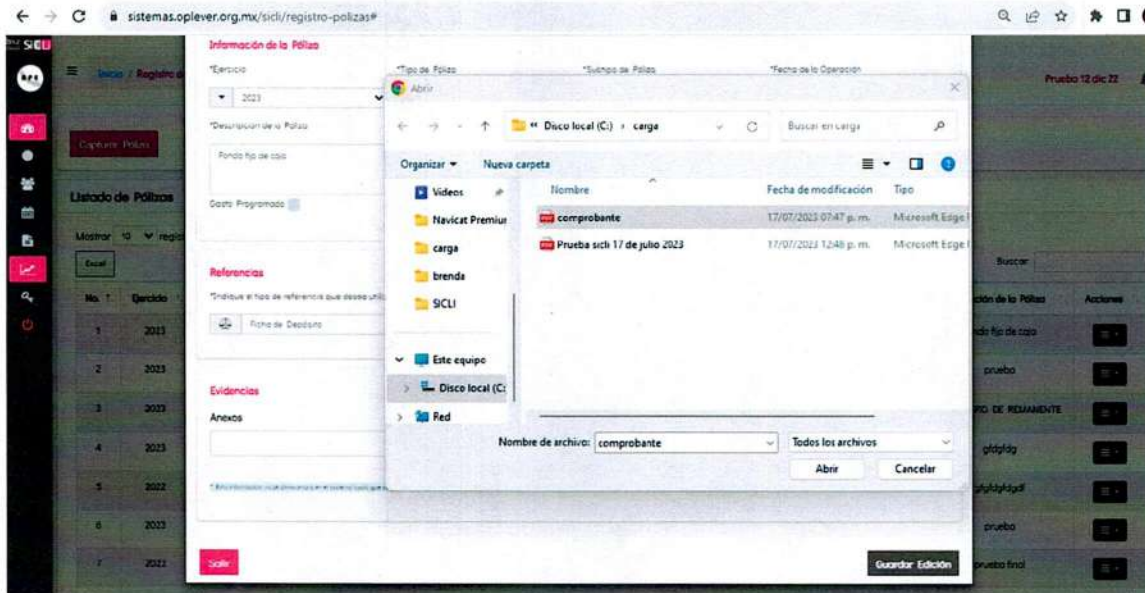
Captura de pólizas



Registro de movimientos contables



Adjuntar y visualizar evidencias




## Formato IF APE

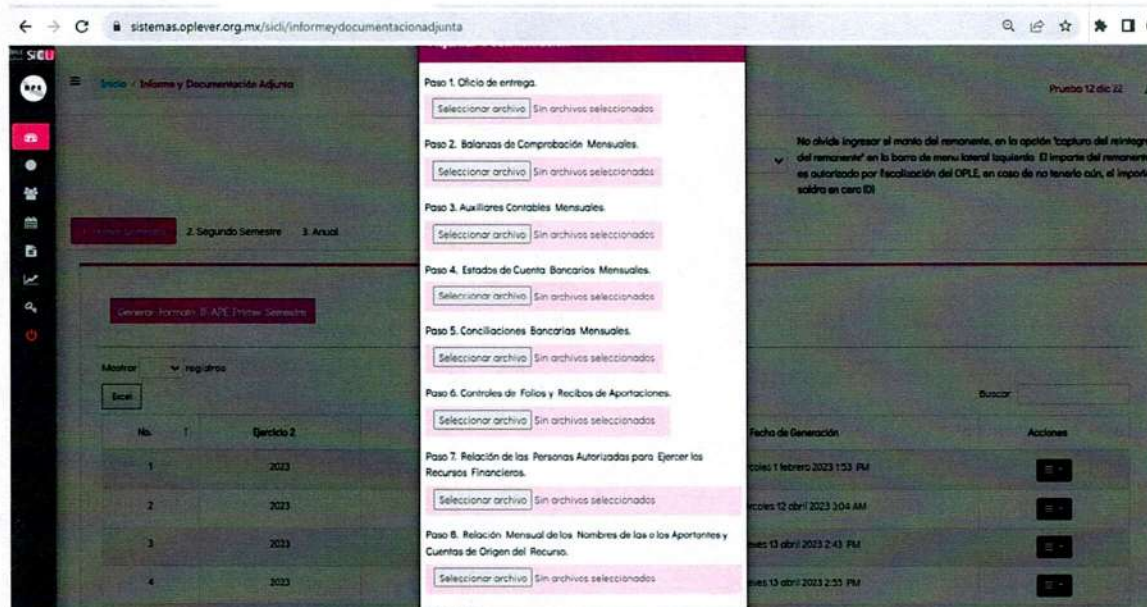
### Generación



The screenshot shows the 'Generación' screen for the 'Formato IF APE Primer Semestre'. It includes a sidebar with navigation options, a top navigation bar, and a main content area. The main content area features a 'Selección del año' dropdown set to '2023', a 'Mostrar' dropdown set to 'registros', and a 'Buscar' input field. Below these elements is a table with the following data:

No.	Ejercicio 2	Nombre del Reporte	Fecha de Generación	Acciones
1	2023	IF APE Primer Semestre	miércoles 1 febrero 2023 1:53 PM	[Icon]
2	2023	IF APE Primer Semestre	miércoles 12 abril 2023 3:04 AM	[Icon]
3	2023	IF APE Primer Semestre	jueves 13 abril 2023 2:43 PM	[Icon]

### Carga de evidencias



The screenshot shows the 'Carga de evidencias' screen, which is a multi-step process for uploading evidence. The steps are:

- Paso 1. Oficio de entrega. (Seleccionar archivo) Sin archivos seleccionados
- Paso 2. Balanzas de Comprobación Mensuales. (Seleccionar archivo) Sin archivos seleccionados
- Paso 3. Auxilios Contables Mensuales. (Seleccionar archivo) Sin archivos seleccionados
- Paso 4. Estados de Cuenta Bancarios Mensuales. (Seleccionar archivo) Sin archivos seleccionados
- Paso 5. Conciliaciones Bancarias Mensuales. (Seleccionar archivo) Sin archivos seleccionados
- Paso 6. Contróles de Folios y Recibos de Aportaciones. (Seleccionar archivo) Sin archivos seleccionados
- Paso 7. Relación de las Personas Autorizadas para Ejercer los Recursos Financieros. (Seleccionar archivo) Sin archivos seleccionados
- Paso 8. Relación Mensual de los Nombres de los Aportantes y Cuentas de Origen del Recurso. (Seleccionar archivo) Sin archivos seleccionados

The interface also includes a sidebar, a top navigation bar, and a main content area with a 'Mostrar' dropdown set to 'registros' and a 'Buscar' input field. A table with the following data is visible in the background:

No.	Ejercicio 2	Nombre del Reporte	Fecha de Generación	Acciones
1	2023	IF APE Primer Semestre	miércoles 1 febrero 2023 1:53 PM	[Icon]
2	2023	IF APE Primer Semestre	miércoles 12 abril 2023 3:04 AM	[Icon]
3	2023	IF APE Primer Semestre	miércoles 13 abril 2023 2:43 PM	[Icon]
4	2023	IF APE Primer Semestre	jueves 13 abril 2023 2:55 PM	[Icon]

Como se puede observar en los ejemplos son procesos que van llevando a diferentes niveles y a cargas de documentación, lo cual implica revisar tanto el flujo de datos correcto del proceso, como la visualización para los dos perfiles (UF y



APE), carga y visualización de evidencias, esto va haciendo compleja la programación y genera mayor uso de líneas de código.

También es importante mencionar que para la generación de un estado financiero se deben verificar el cumplimiento de muchas condiciones y formulas, para esta actividad se usan los controladores y en ellos está contenido el código para estos productos, lo cual también hace complejo el análisis y la verificación correcta de ellos en el plazo previsto.

A continuación, se presente un fragmento del controlador para generar los estados financieros, el cual corresponde específicamente a la balanza de comprobación, que contiene un total de 212 líneas, de un total de más de cuatro mil que contiene el controlador:

```
public function balanza_comprobacion(Request $request)
{
    $breadcrumb = "Balanza de Comprobación";
    $id_area = Auth::user()->id_area;
    $ape = Ape::find($id_area);
    $tipo_reporte = "balanza_comprobacion";
    $sufijo = $ape->sufijo;
    $dataHoraDos = date('Y-m-d-H-i-s');
    // $tipo = Auth::user()->tipo;
    $user = auth()->user();
    $tipo = $user->tipo;
    $id = $user->id;
    $fecha_uno = Carbon::parse($request->get('dateIniBalCom'))->startOfDay();
    $fecha_dos = Carbon::parse($request->get('dateFinLibroBalCom'))->endOfDay();
    // dd($fecha_uno,$fecha_dos);
    $dataHora = date('Y-m-d H:i:s');
    $foliosito = mt_rand();
    $encrypted_txt = $this->encrypt_decrypt('encrypt', $foliosito);
    $decrypted_txt = $this->encrypt_decrypt('decrypt', $encrypted_txt);
    $rest1 = substr($encrypted_txt, 0, 7);
    $rest2 = substr($encrypted_txt, 7, 5);
    $rest3 = substr($encrypted_txt, 12, 8);
    $rest4 = substr($encrypted_txt, 20, 6);
    $rest5 = substr($encrypted_txt, 26, 6);
    // Se obtiene los catalogos con sus subcuentas y movimientos de cada cuenta
    $cuentas = Catalogo::with(['movimientos' => function ($q) use ($id_area, $fecha_uno, $fecha_dos) {
        $q->where('id_ape', '=', $id_area);
        $q->where('status', '=', 'Activo');
        $q->whereDate('fecha_operacion', '>=', $fecha_uno);
        $q->whereDate('fecha_operacion', '<=', $fecha_dos);
    }, 'movimientos.poliza', 'subcuentas' => function ($q) use ($id_area) {
        $q->where('id_ape', '=', $id_area);
```

```
    }, 'subcuentas.movimientos' => function ($q) use ($id_area,
$fecha_uno, $fecha_dos) {
    $q->where('id_ape', '=', $id_area);
    $q->where('status', '=', 'Activo');
    $q->whereDate('fecha_operacion', '>=', $fecha_uno);
    $q->whereDate('fecha_operacion', '<=', $fecha_dos);
    })->orderBy('clave', 'ASC')->get();
    //dd($cuentas[4]);
    $mesUno = $fecha_uno->month;
    $mesDos = $fecha_dos->month;
    $anioUno = $fecha_uno->year;
    $anioDos = $fecha_dos->year;
    $sumaSaldosIniciales = 0;
    $sumaCargos = 0;
    $sumaAbono = 0;
    $sumaTotal = 0;
    foreach ($cuentas as $cuenta) {
        // Saldos Iniciales
        if ($mesUno === 1) {
            $saldo_inicial_cargo = $cuenta->movimientos->reject(function
($query) {
                return !$query->inicial();
            })->sum('cargo');
            $saldo_inicial_abono = $cuenta->movimientos->reject(function
($query) {
                return !$query->inicial();
            })->sum('abono');
            $clave = explode('-', $cuenta->clave);
            switch ($clave[0]) {
                case '1':
                case '5':
                    $saldo_inicial = $saldo_inicial_cargo - $saldo_inicial_abono;
                    break;
                case '2':
                case '3':
                case '4':
                    $saldo_inicial = $saldo_inicial_abono - $saldo_inicial_cargo;
                    break;
                default:
                    $saldo_inicial = $saldo_inicial_abono;
                    break;
            }
        } else {
            $saldo_inicial = $this->saldosInicialesAnteriores($id_area,
$fecha_uno, $fecha_dos, $cuenta);
        }
        $cuenta->saldo_inicial = $saldo_inicial;
        // Suma de los movimientos de cargos de una cuenta
        $total_cargo = $cuenta->movimientos->reject(function ($query) {
            return $query->inicial();
        })->sum('cargo');
        $sumaCargos += $total_cargo;
        $cuenta->total_cargo = $total_cargo;
    }
}
```



```
// Suma de los moviemotos de abono de una cuenta
$total_abono = $cuenta->movimientos->reject(function ($query) {
    return $query->inicial();
})->sum('abono');
if (in_array($cuenta->id, [16, 18, 20, 22, 24])) {
    // dd($cuenta);
    $sumaAbono += 0;
    $cuenta->total_abono = 0;
} else {
    $sumaAbono += $total_abono;
    $cuenta->total_abono = $total_abono;
}
// Si tiene subcuentas se tendrá que sumar también sus cargos y
abonos
if (count($cuenta->subcuentas) != 0) {
    foreach ($cuenta->subcuentas as $subcuenta) {
        // Saldos Iniciales
        if ($mesUno == 1) {
            $saldo_inicial_sub_cargo = $subcuenta->movimientos-
>reject(function ($query) {
                return !$query->inicial();
            })->sum('cargo');
            $saldo_inicial_sub_abono = $subcuenta->movimientos-
>reject(function ($query) {
                return !$query->inicial();
            })->sum('abono');
            $clave = explode('-', $subcuenta->clave);
            switch ($clave[0]) {
                case '1':
                case '5':
                    $saldo_inicial_sub = $saldo_inicial_sub_cargo -
                    $saldo_inicial_sub_abono;
                    break;
                case '2':
                case '3':
                case '4':
                    $saldo_inicial_sub = $saldo_inicial_sub_abono -
                    $saldo_inicial_sub_cargo;
                    break;
                default:
                    $saldo_inicial_sub = $saldo_inicial_sub_abono;
                    break;
            }
        } else {
            $saldo_inicial_sub = $this-
>saldosInicialesAnterioresSubcuenta($id_area, $fecha_uno,
$fecha_dos, $subcuenta);
        }
        $subcuenta->saldo_inicial = $saldo_inicial_sub;
        // Suma de los movientos de cargos de una subcuenta
        $total_cargo = $subcuenta->movimientos->reject(function
($query) {
            return $query->inicial();
        });
    }
}
```




```
    })->sum('cargo');
    $subcuenta->total_cargo = $total_cargo;
    // Suma de los moviemotos de abono de una cuenta
    $total_abono = $subcuenta->movimientos->reject(function
($query) {
    return $query->inicial();
    })->sum('abono');
    $subcuenta->total_abono = $total_abono;
    $clave = explode('-', $subcuenta->clave);
    switch ($clave[0]) {
    case '1':
    case '5':
        $total = ($subcuenta->saldo_inicial + $subcuenta-
>total_cargo) - $subcuenta->total_abono;
        break;
    case '2':
    case '3':
    case '4':
        $total = ($subcuenta->saldo_inicial + $subcuenta-
>total_abono) - $subcuenta->total_cargo;
        break;
    default:
        $total = $subcuenta->total_abono;
        break;
    }
    $subcuenta->total = $total;
    }
    }
    $clave = explode('-', $cuenta->clave);
    switch ($clave[0]) {
    case '1':
    case '5':
        $total = ($cuenta->saldo_inicial + $cuenta->total_cargo) -
$cuenta->total_abono;
        $sumaSalDOSIniciales += $saldo_inicial;
        $sumaTotal += $total;
        break;
    case '2':
    case '3':
    case '4':
        $total = ($cuenta->saldo_inicial + $cuenta->total_abono) -
$cuenta->total_cargo;
        $sumaSalDOSIniciales -= $saldo_inicial;
        $sumaTotal -= $total;
        break;
    default:
        $total = $cuenta->total_abono;
        break;
    }
    $cuenta->total = $total;
    }
    }
    $cuentas = $this->sum_balance($cuentas);
    $cuentas = $this->depreciacionBalanza($cuentas);
```

```
$pdf = PDF::loadView('balanza', compact(
    'ape',
    'fecha_uno',
    'fecha_dos',
    'dataHora',
    'rest1',
    'rest2',
    'rest3',
    'rest4',
    'rest5',
    'cuentas',
    'sumaSaldosIniciales',
    'sumaCargos',
    'sumaAbono',
    'sumaTotal'
));
$filename = $sufijo . "_balanza_comprobacion_del_" .
$dataHoraDos . '.pdf';
$notifications = Notify::where('ape', '=', $id_area)->where('status', '=',
'0')->where('a', '!=', $id)->orderBy('created_at', 'DESC')->get();
$content = $pdf->output();
Storage::put('RegistroContable/Reportes/' . strtoupper($ape->sufijo)
. '/' . $filename, $content);
$ruta = 'reporte-balanza-comprobacion';
return view('vista_de_reporte', compact('tipo', 'filename',
'tipo_reporte', 'foliosito', 'dataHora', 'ruta', 'ape', 'breadcrumb',
'notifications'));
}
```

### c) Los requerimientos de infraestructura de procesamiento de datos y almacenamiento.

En cuanto al hardware cabe mencionar que está compuesta de la siguiente forma: el sistema esta alojado en una máquina virtual con espacio de 200 gigas, 24Gb memoria RAM y un procesador de 16 núcleos, actualmente se ocupa el 31% de espacio.

Para poder realizar la auditoria sería necesario una clonación de la máquina virtual para salvaguardar la integridad de los sistemas que se alojan y realizar pruebas o ejercicios de manera segura sin afectar los datos de los sistemas en producción, una vez realizada la clonación es necesario la interconexión de manera local para poder acceder al sistema y en dado caso instalar las herramientas de auditoría necesarias para realizar las tareas propuestas.



#### **d) situaciones generales adversas**

El tiempo necesario para realizar las actividades de revisión y/o auditoría es muy elevado, e implicaría destinar personal exclusivo para la realización de las pruebas de escritorio de todos los procesos, lo cual implicaría redirigir recursos humanos y materiales a dicha actividad durante al menos 2 meses

También sería necesario que el área usuaria verifique el correcto funcionamiento, siendo necesario, a efecto de dotar el procedimiento de mayor certeza, capturar un ORÁCULO (un ejercicio contable diseñado para la prueba) el cual debe tener perfectamente definidas las entradas y las salidas del sistema, para poder constatar ambos resultados, lo cual no es viable emitirlo en el presente informe, por el corto plazo disponible, además de la proximidad del Proceso Electoral 2023-2024 ya que los recursos humanos y técnicos están enfocados al desarrollo, pruebas e implementación de los Sistemas que serán utilizados en el Proceso Electoral Local Ordinario 2023-2024.

#### **Conclusión**

Por lo antes expuesto, esta Unidad Técnica de Servicios Informáticos concluye que, dada las condiciones descritas en el presente informe, no es posible realizar en el tiempo señalado en el Acuerdo OPLEV/CG095/2023 un ejercicio de verificación y/o auditoría al Sistema de Contabilidad en Línea, que permita a los integrantes del Consejo General determinar, bajo los principios de certeza, imparcialidad, independencia, legalidad, máxima publicidad y objetividad, su utilización para el segundo semestre del año 2023.

