Barranquilla, 30 de noviembre 2022.

Ingeniero.
Diana Ma. Miguel Mantilla Parra
Contrato No CD-06-2022-1476
Secretaria de Planeación.
Alcaldía Distrital de Barranquilla. Ciudad.

Asunto: **DECIMO INFORME DE GESTIÓN.**

En cumplimiento del objeto del contrato de prestación de servicios No. CD-06- 2022-1476,que usted supervisa, me permito rendirle informe de las actividades realizadas en la **PRESTACION DE SERVICIOS PROFESIONALES PARA APOYAR A LA SECRETARIA DISTRITAL DE PLANEACION EN EL DESARROLLO DE LOS PROGRAMAS Y PROYECTOS RELACIONADOS CON GESTION (30) DE NOVIEMBRE DE DOS MIL VEINTIDOS (2022).**

1. **Proyecto Titulación**

Objetivos:

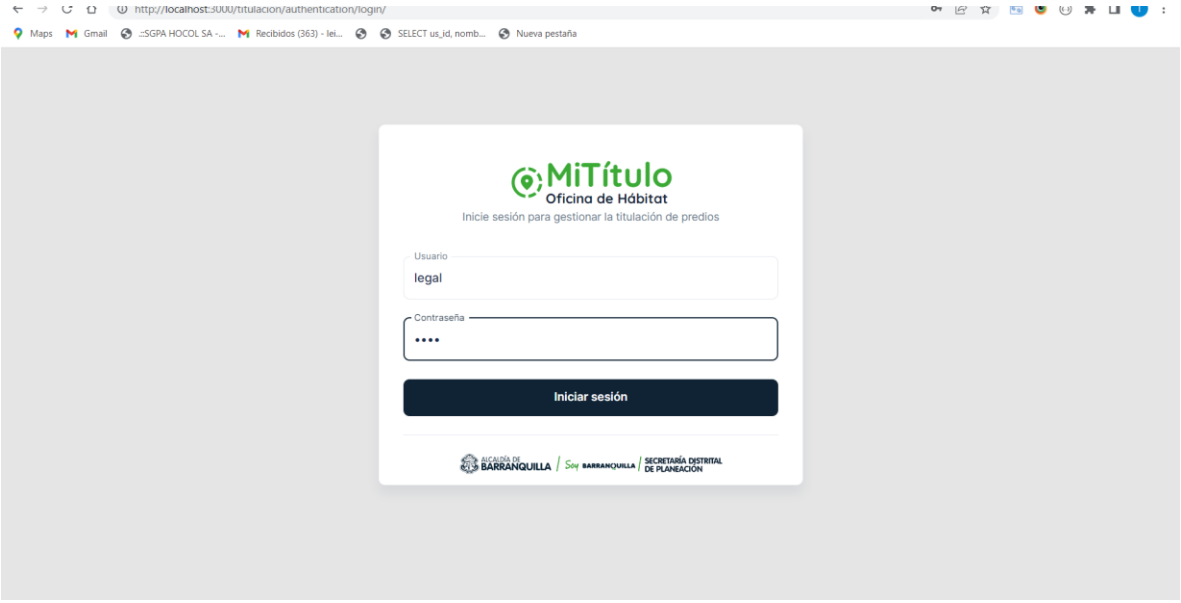
- Se desarrollo backend y frontend para el sub módulo de firmar de aprobación o devolución de la resolución de publicación de predios.

- Se inicio el desarrollo backend y frontend para el sub módulo de firmar de aprobación o devolución de la resolución individual de predios.

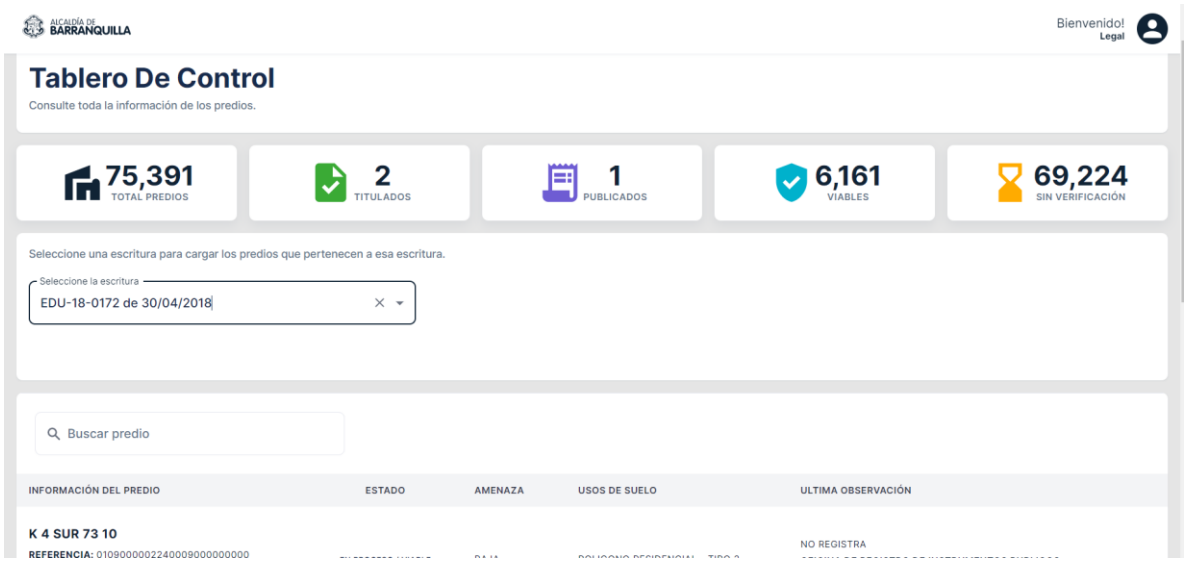
Evidencias funcionales:

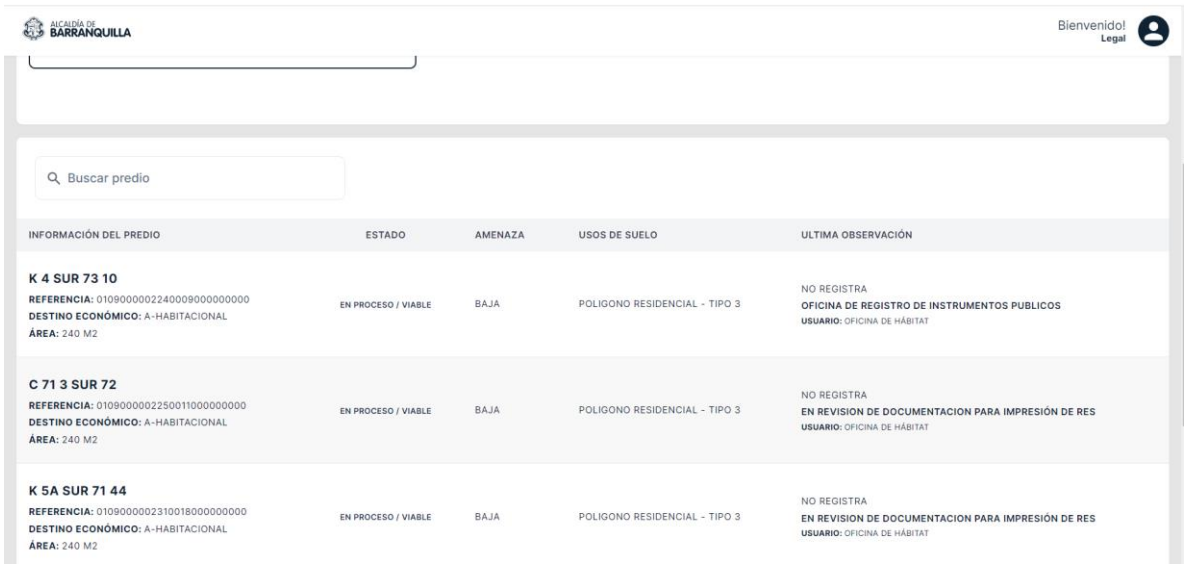
Proceso titulación.

Se ingresa a la aplicación de titulacion.

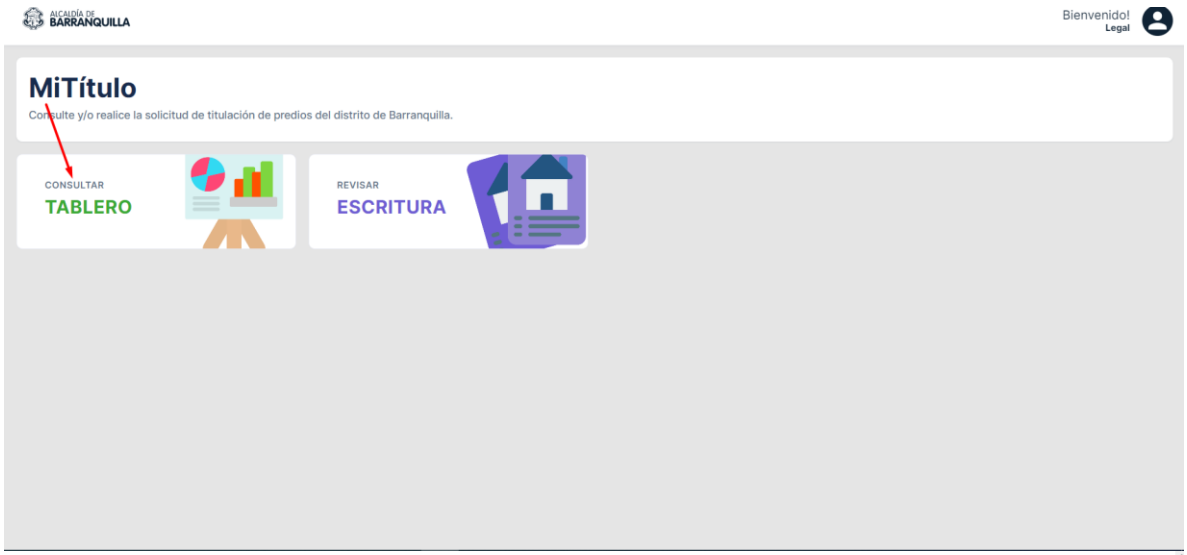


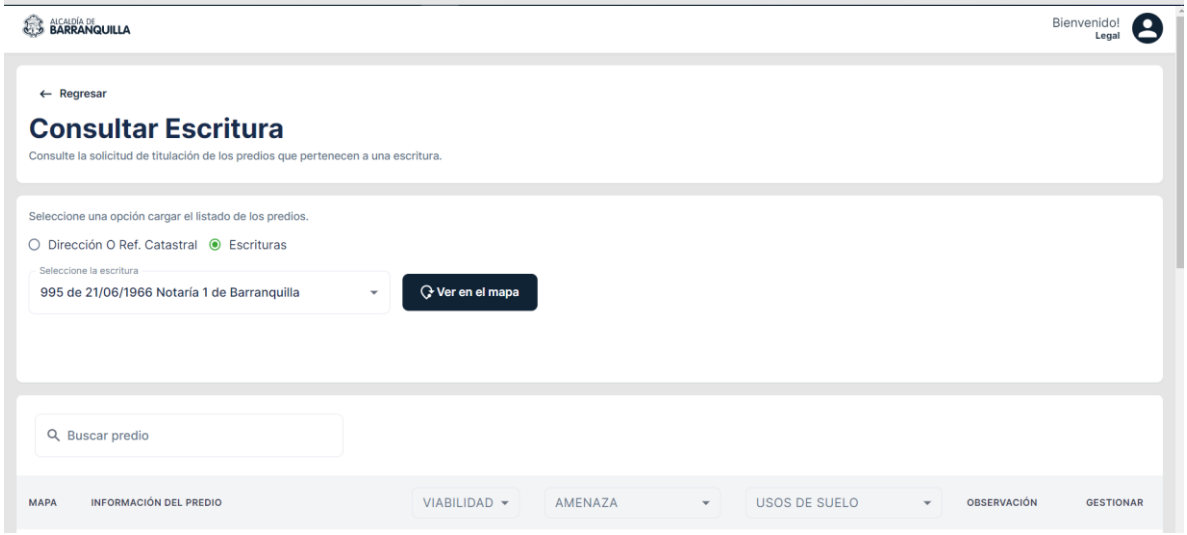
Tablero de control.



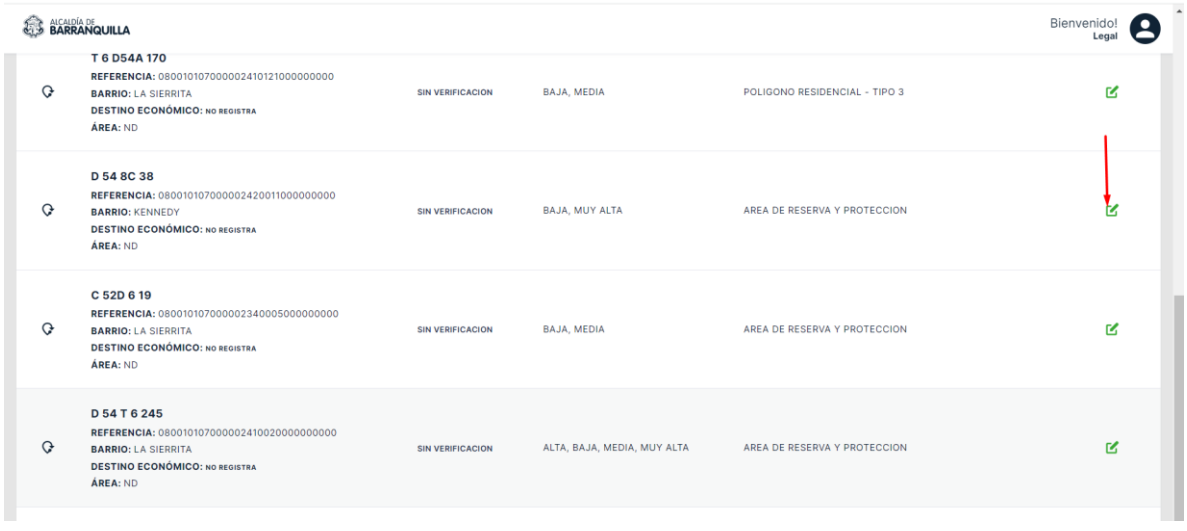


Consultar tablero.

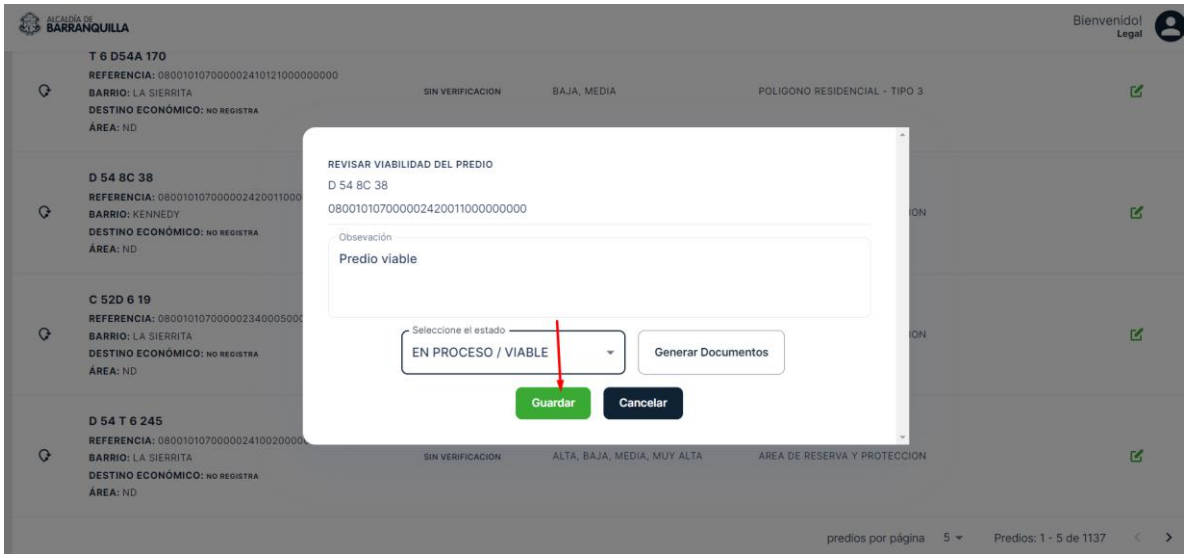




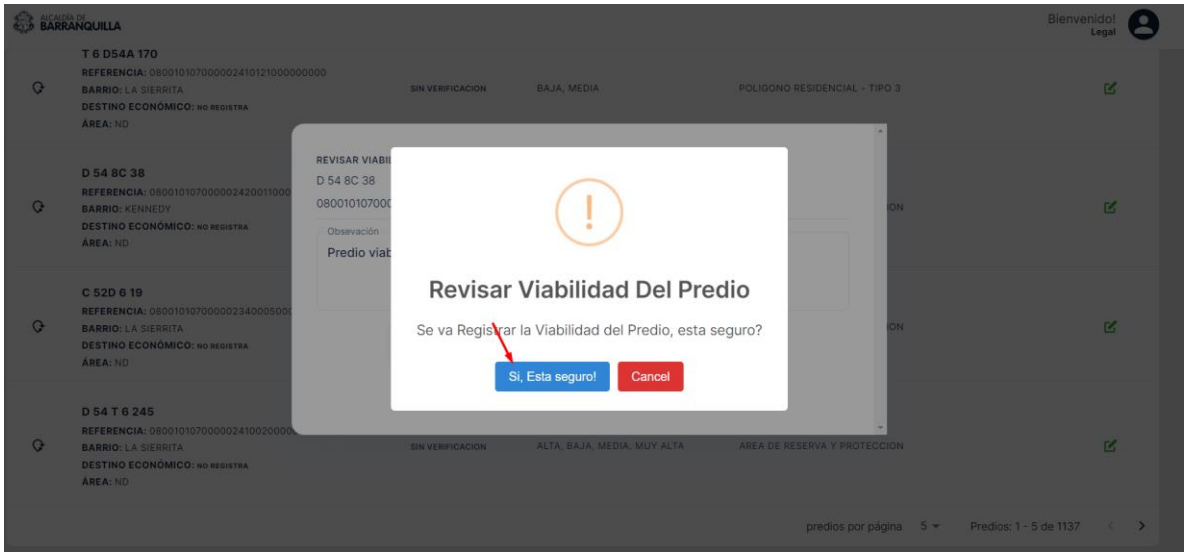
Se ingresa.



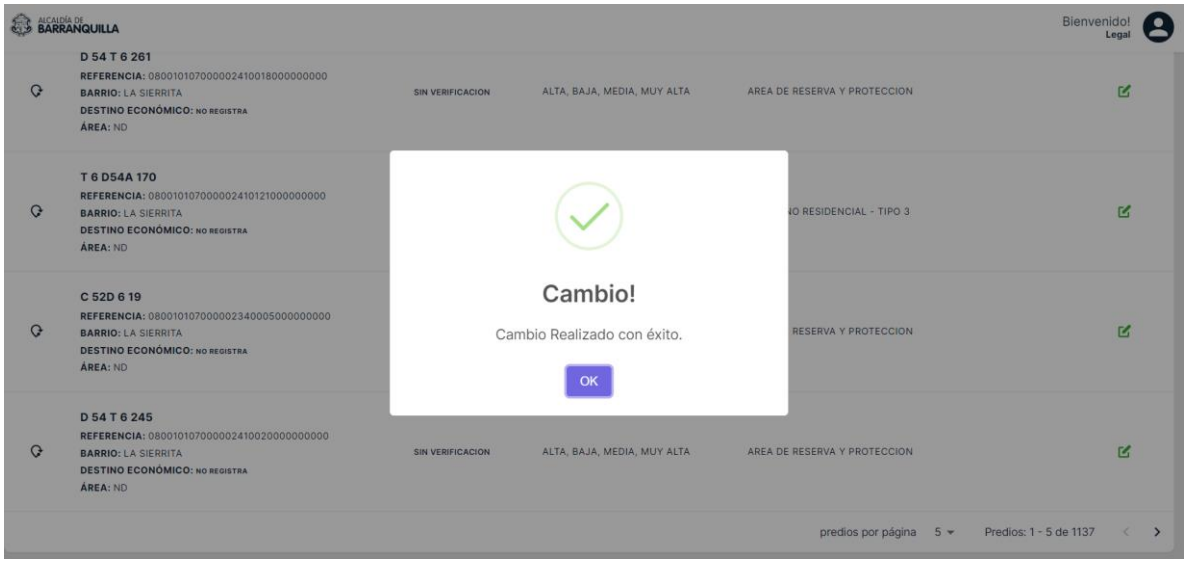
Guardar información.



Si estamos seguro de viabilidad del perdió.



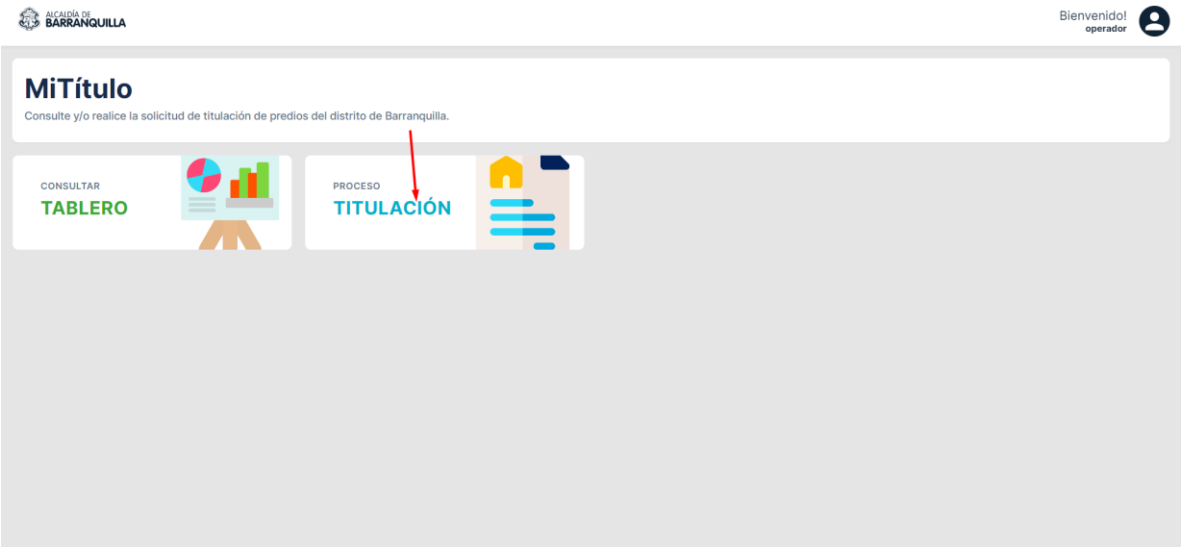
Cambio realizado



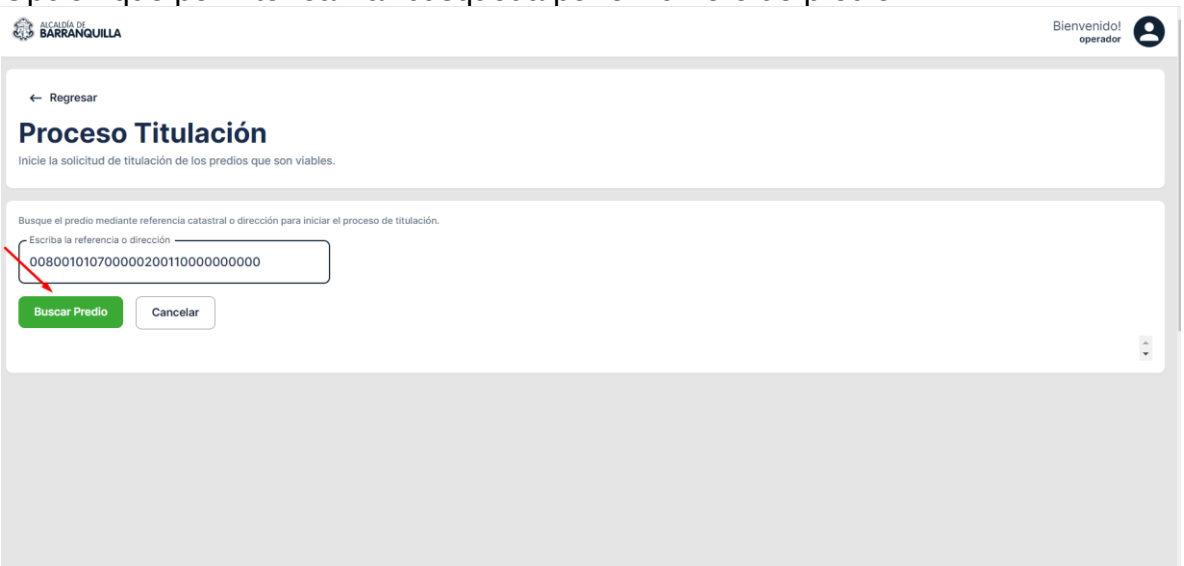
Se ingresa con el usuario de operador.



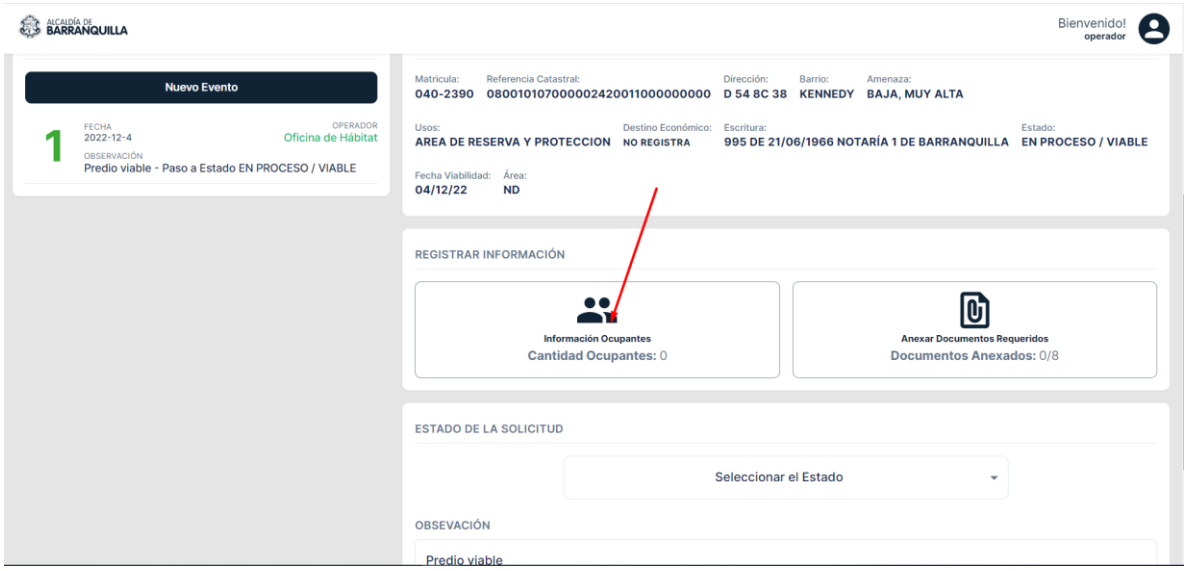
Proceso de titulación.



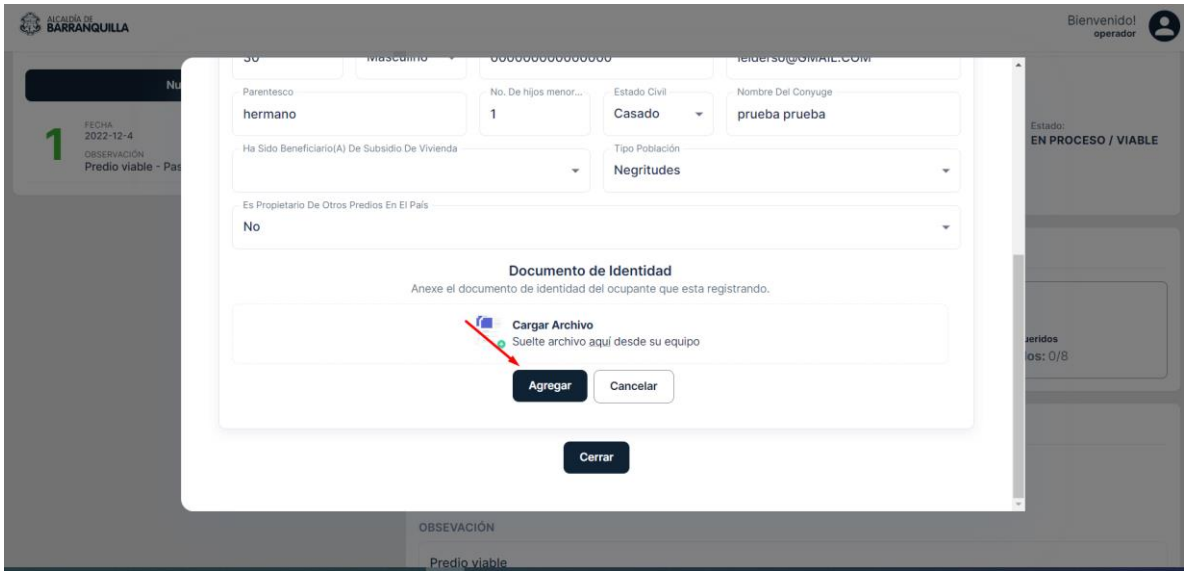
Opción que permite realizar búsqueda por el número de predio.



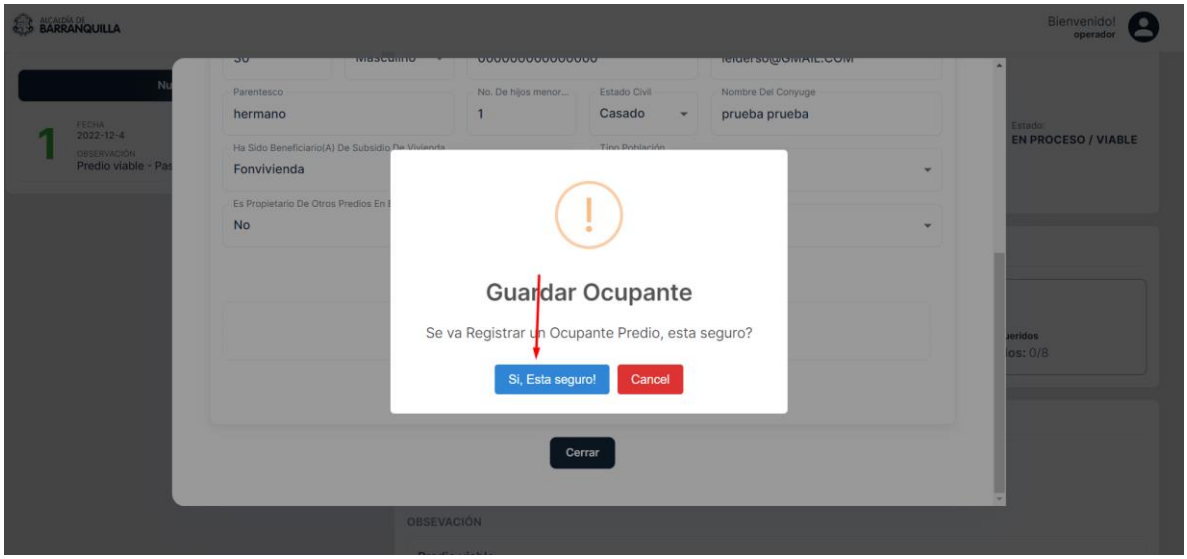
Diligenciar información del ocupante.



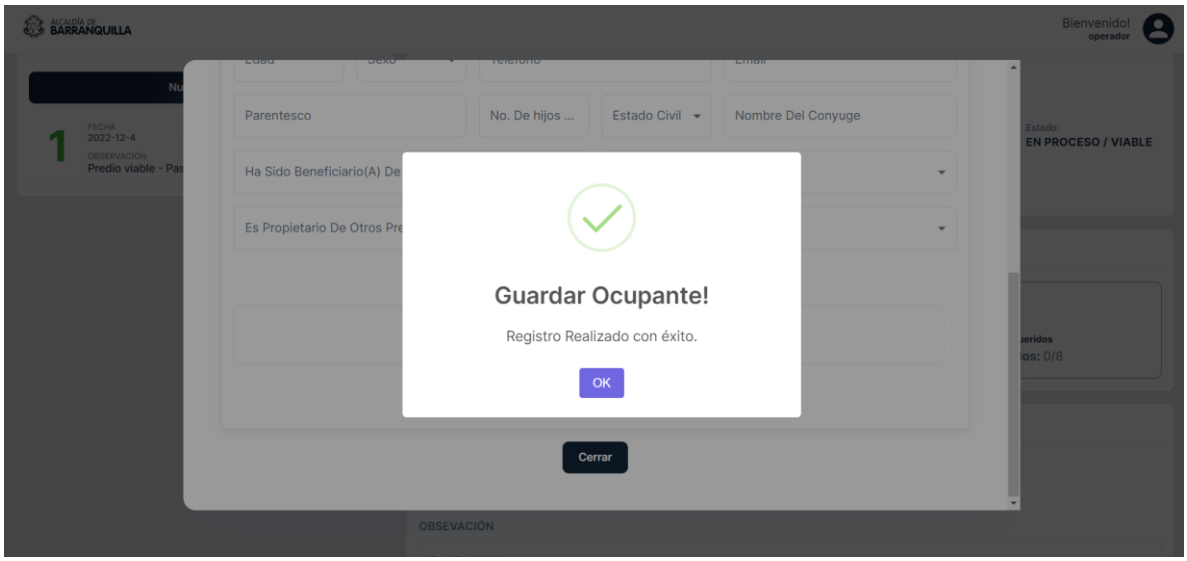
Botón agregar la información diligenciada.



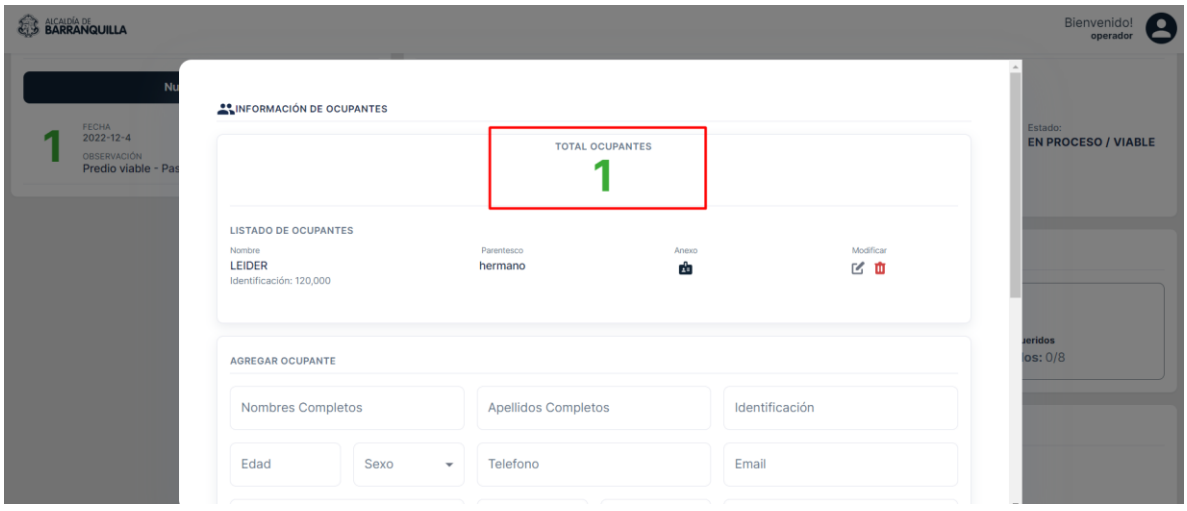
Botón que permite la confirmación si está seguro de guardar.



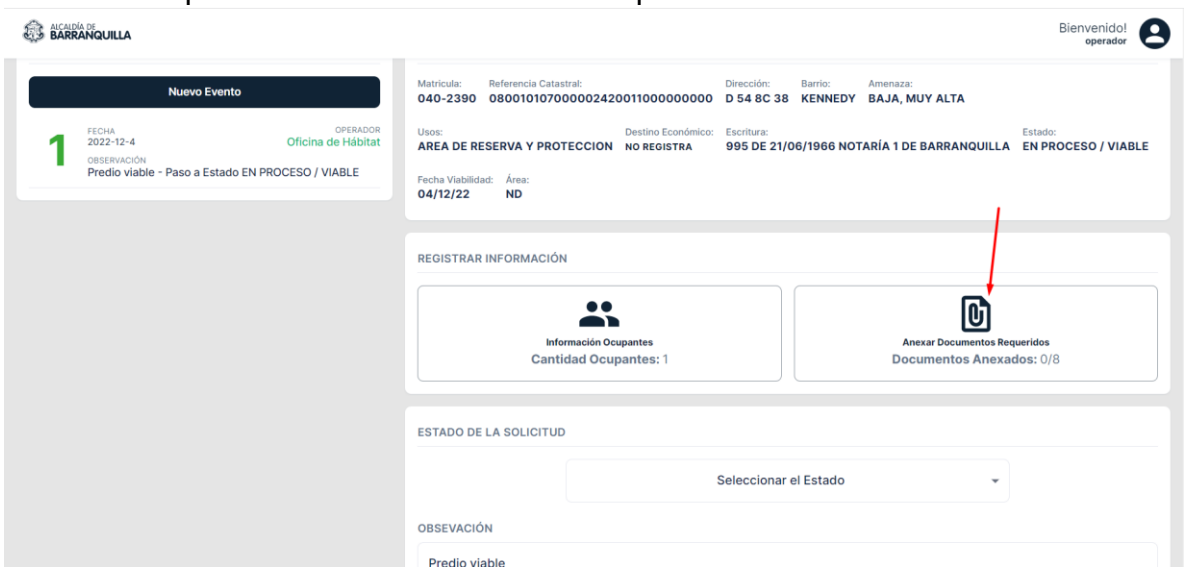
Se evidencia que la información se guardó de forma correcta.



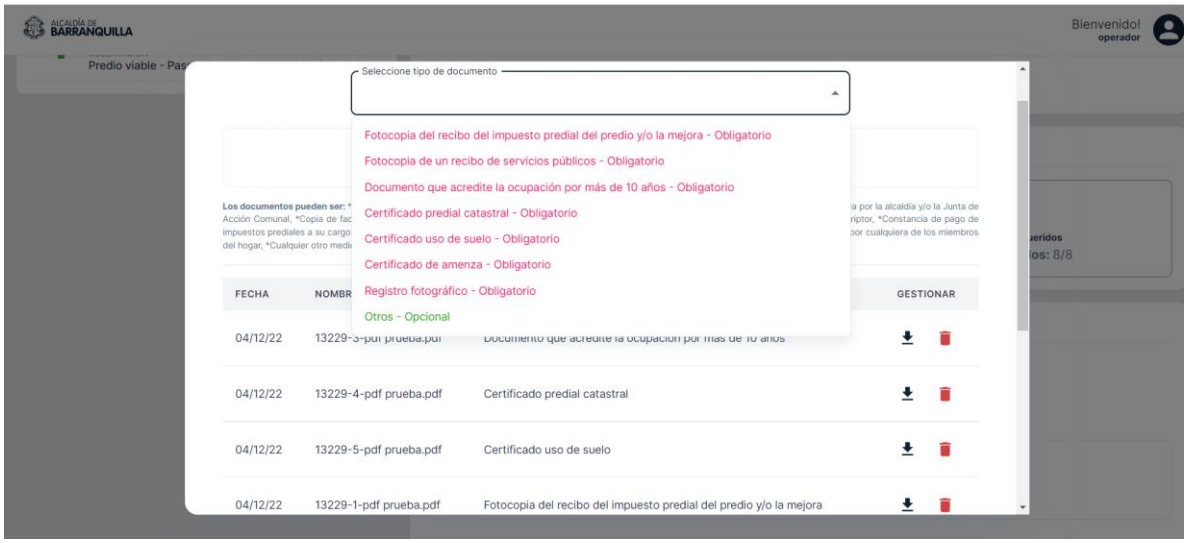
Se visualiza la información del ocupante.



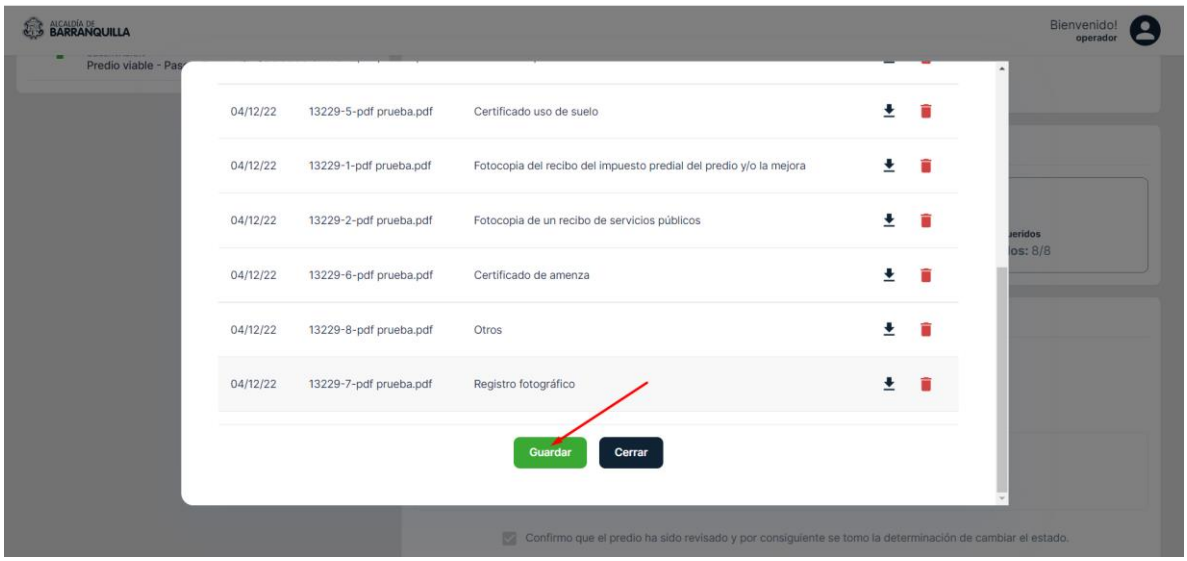
Icono indica para anexar los documentos requeridos.



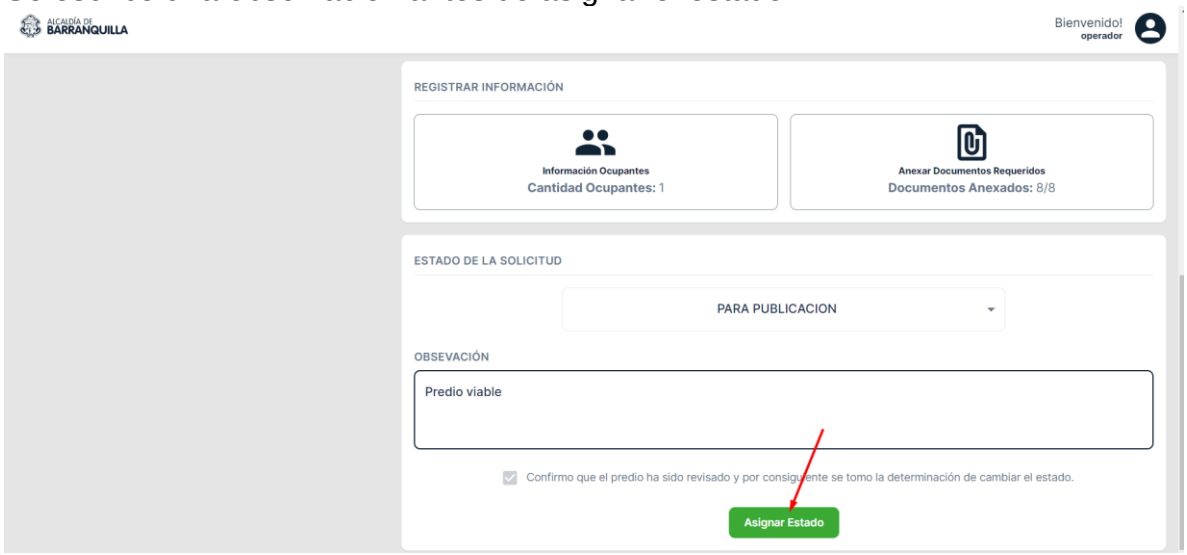
Se visualiza la lista de documentos que se deben anexar.



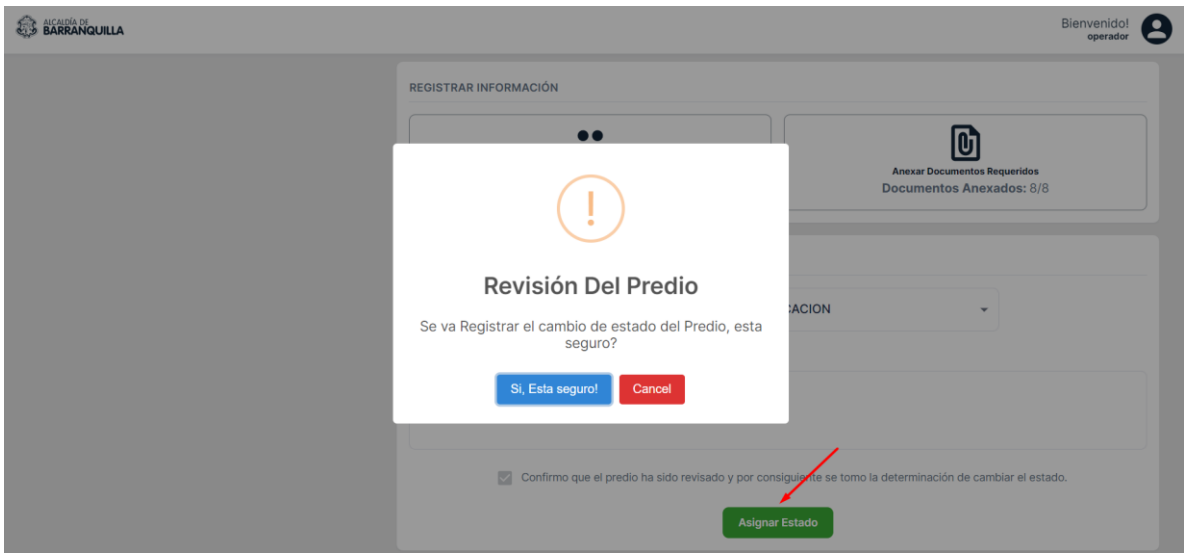
Botón es para guardar los documentos anexados.



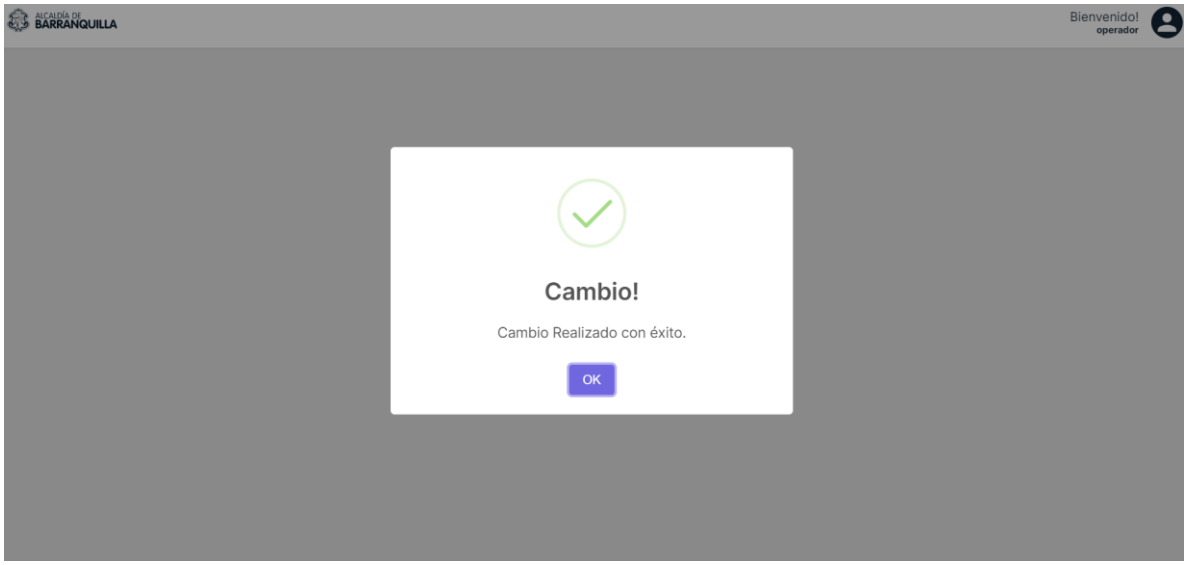
Se escribe una observación antes de asignar el estado.



Confirmar si está seguro de asignar el estado del predio.



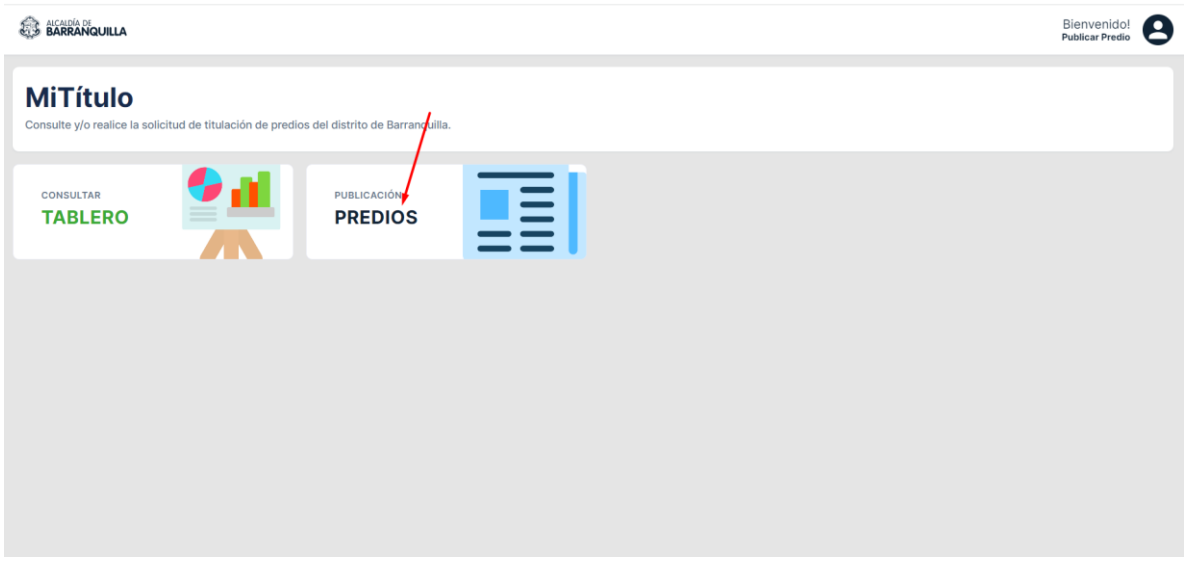
Cambio se realizó con éxito.



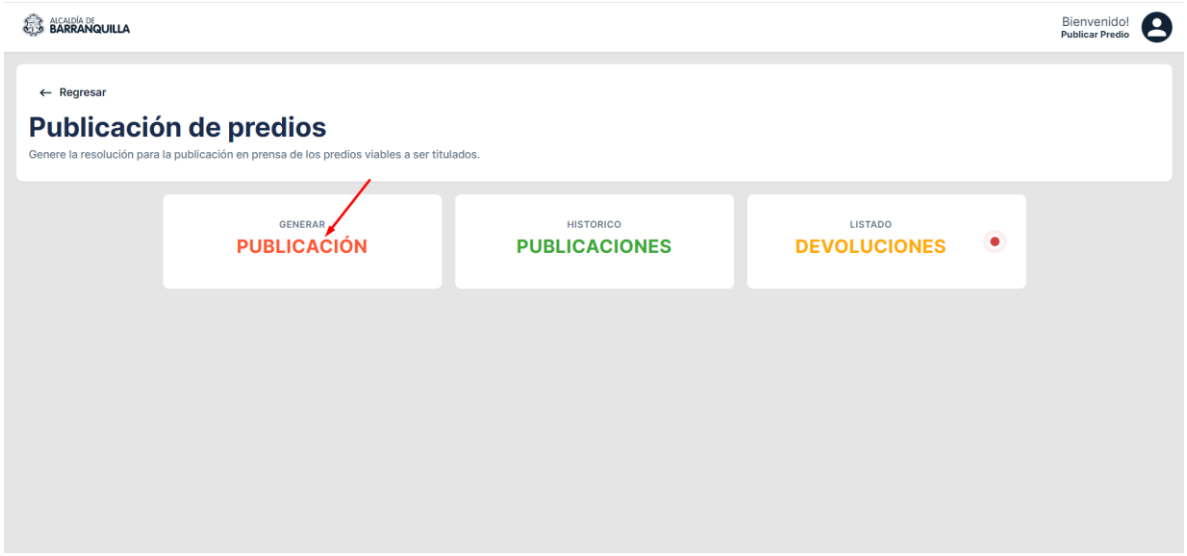
Se ingresa con el usuario publicar.



Publicación de predio.



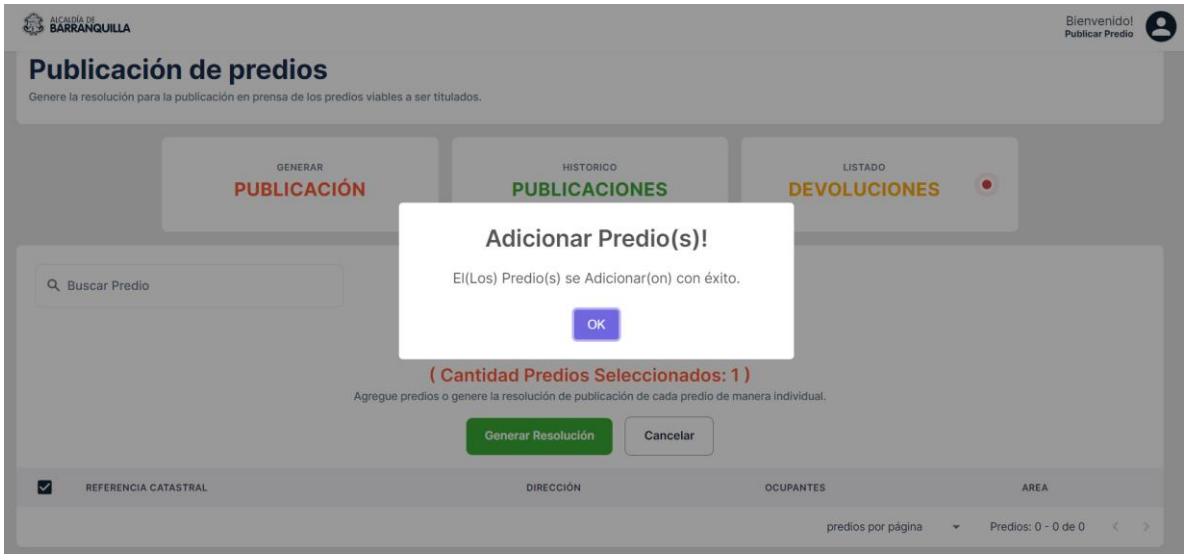
Generar publicación.



Opción permite agregar los predios seleccionado.



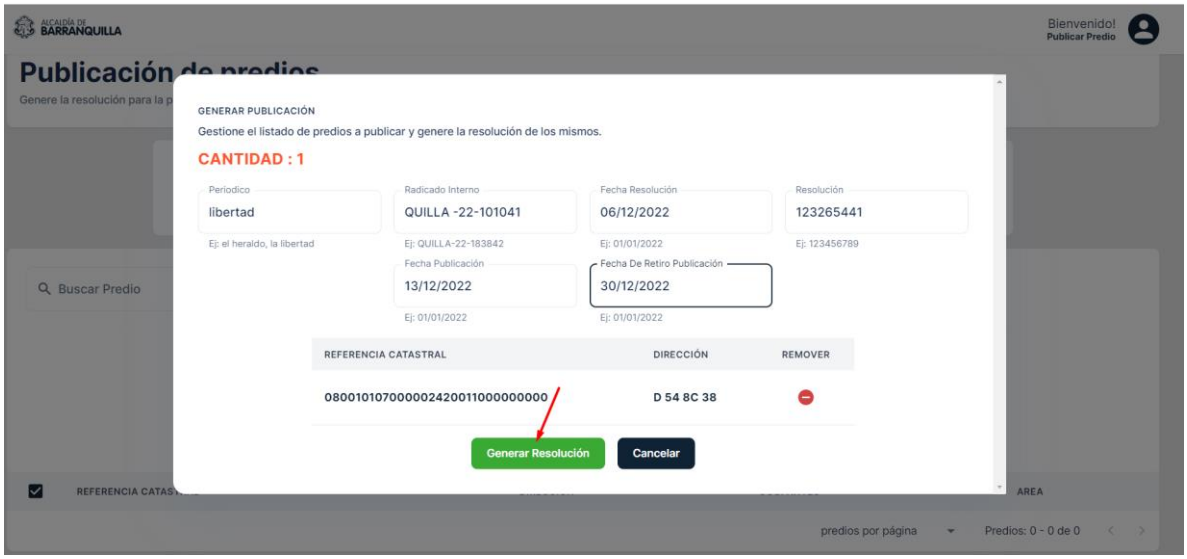
Los predios se adicionaron con éxitos.



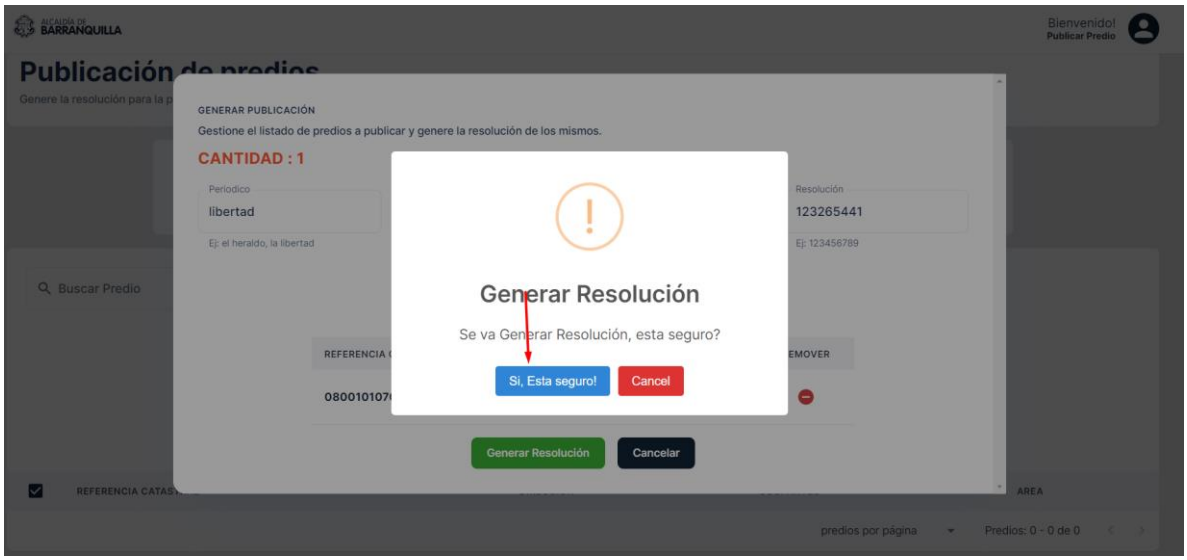
Genera resolución.



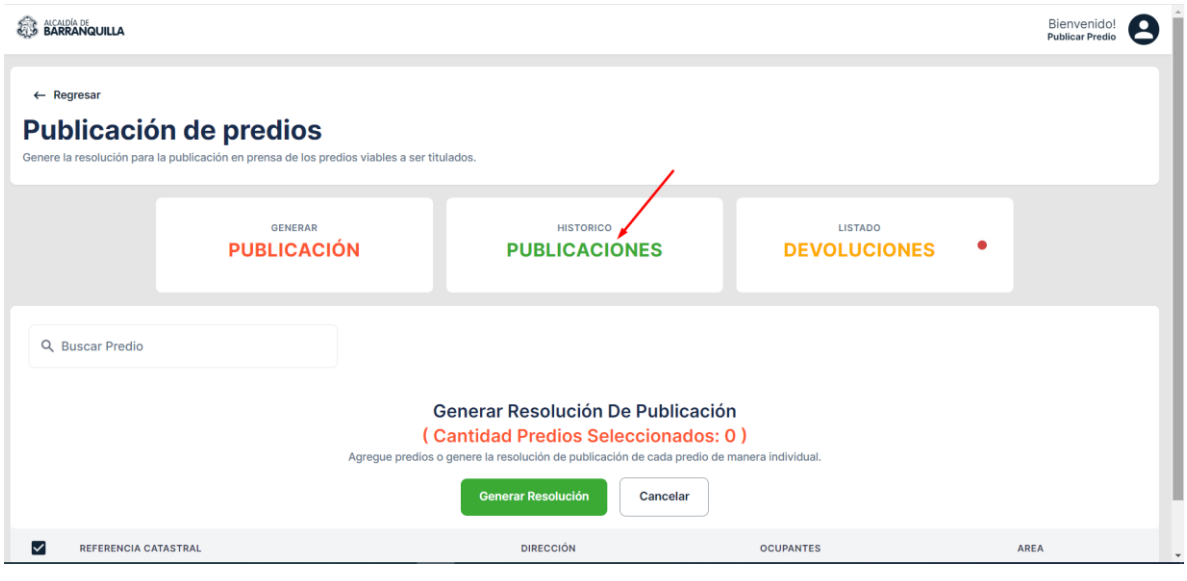
Generar el listado de predios a publicar y generar la resolución de los mismos.



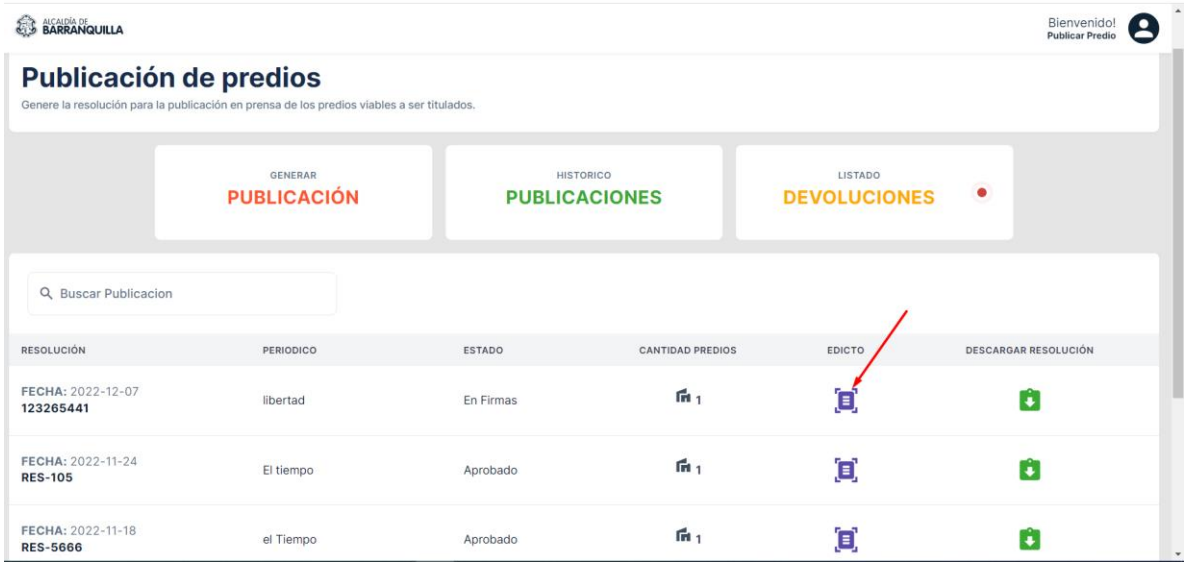
Confirmar si está seguro, generar la resolución.



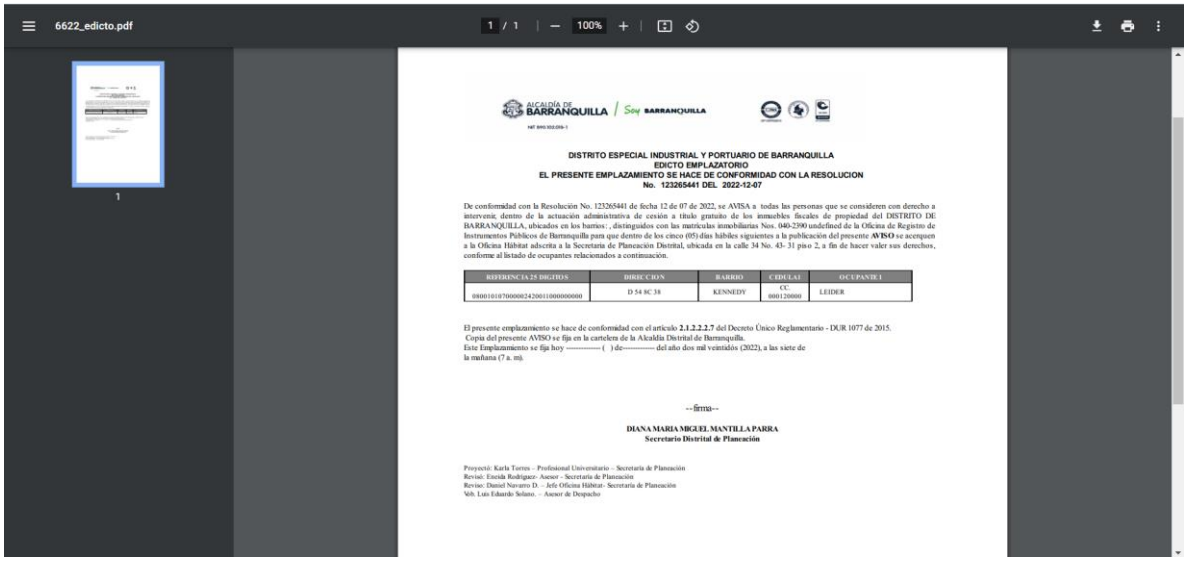
Histórico de publicación.



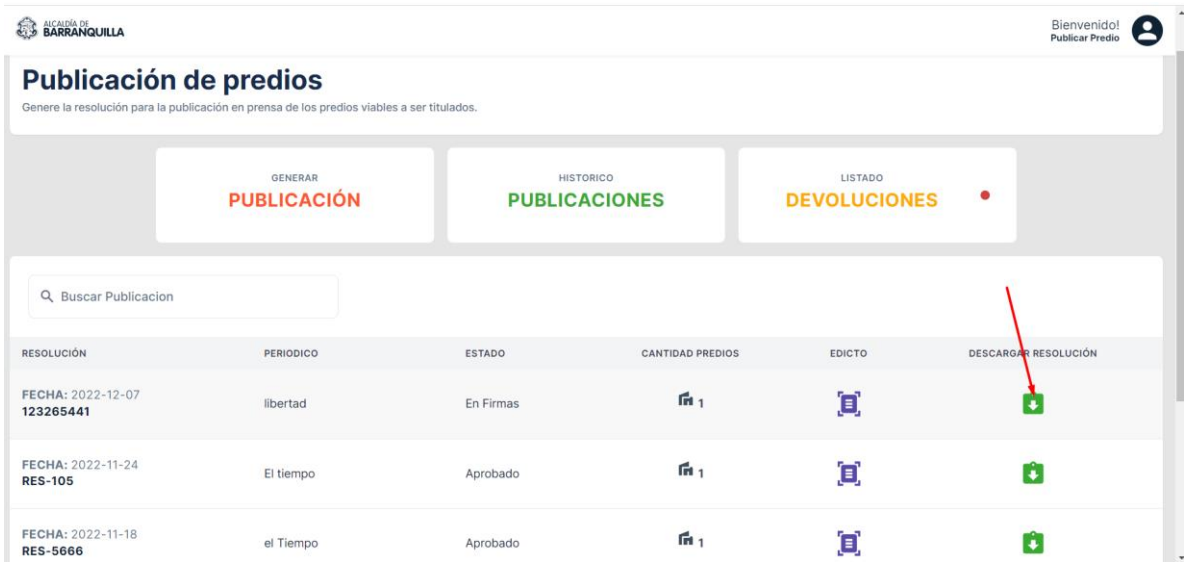
Icono de descarga.



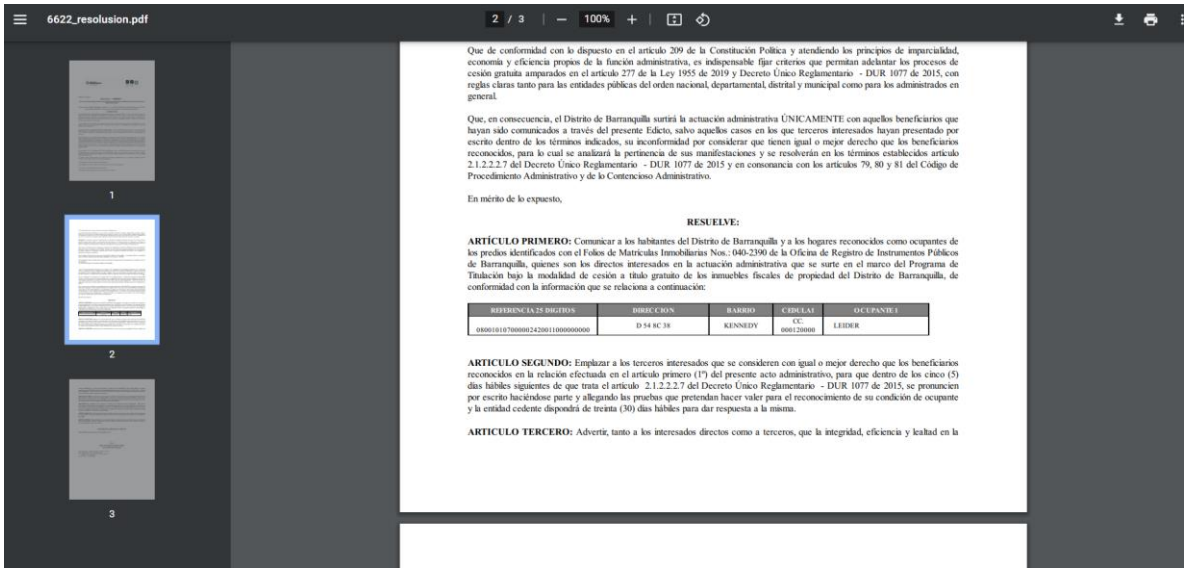
Se visualiza la descarga del PDF, con su respectiva información.



Descarga de resolución.



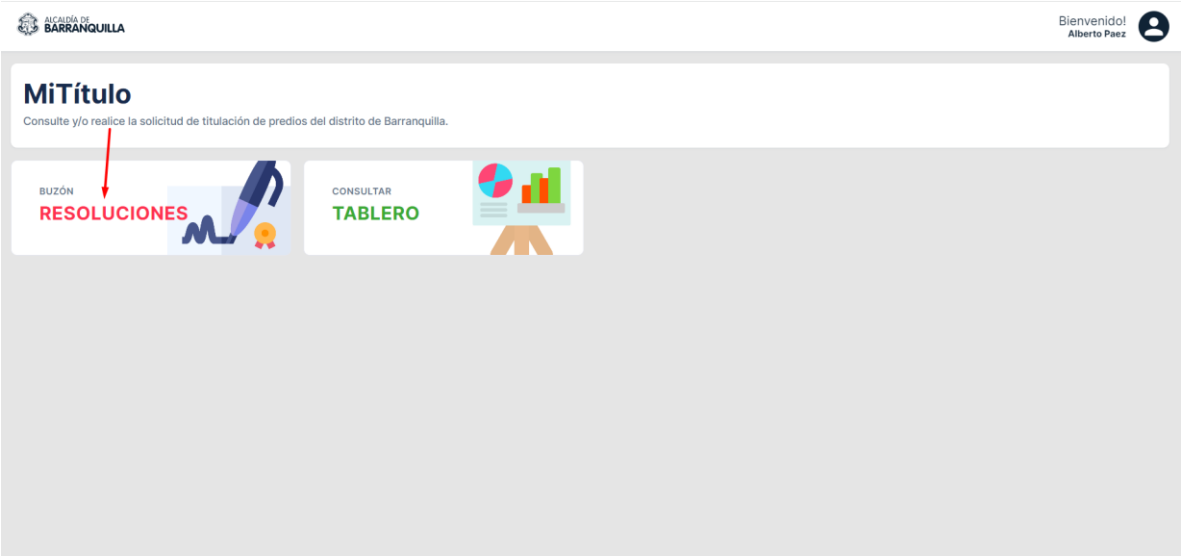
Se visualiza la descarga del PDF de la resolución.



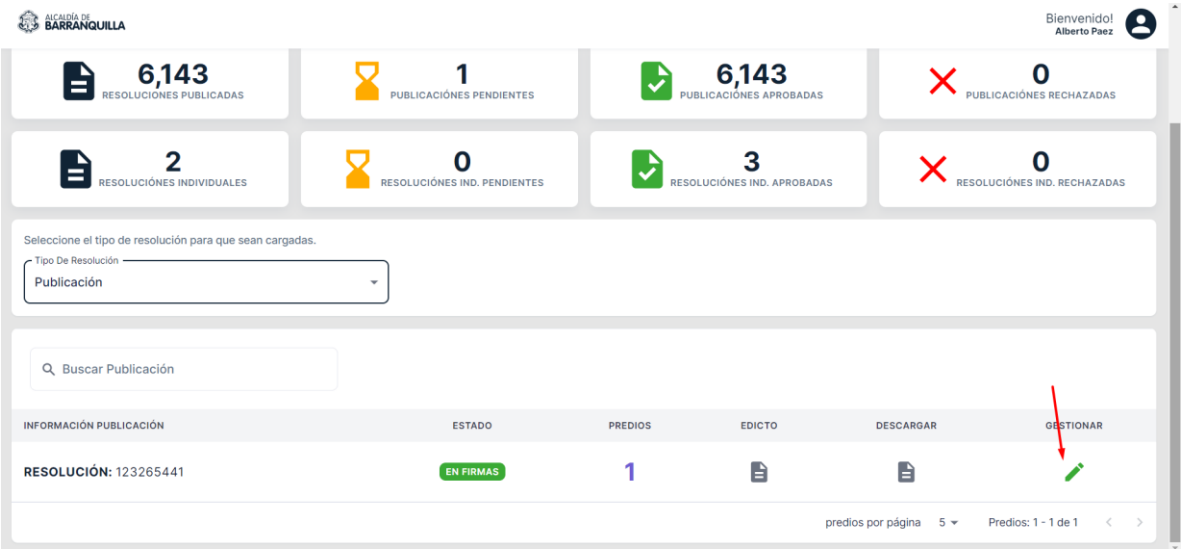
Se ingresa con el usuario Alberto Páez.



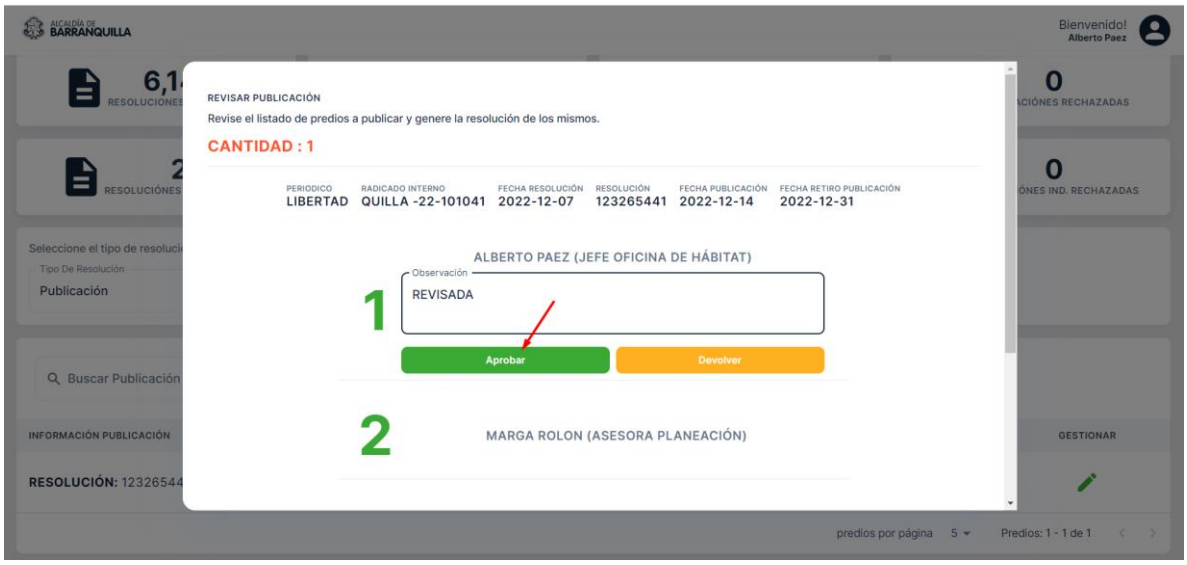
Buzón de resolución.



Se ingresa al icono gestionar.



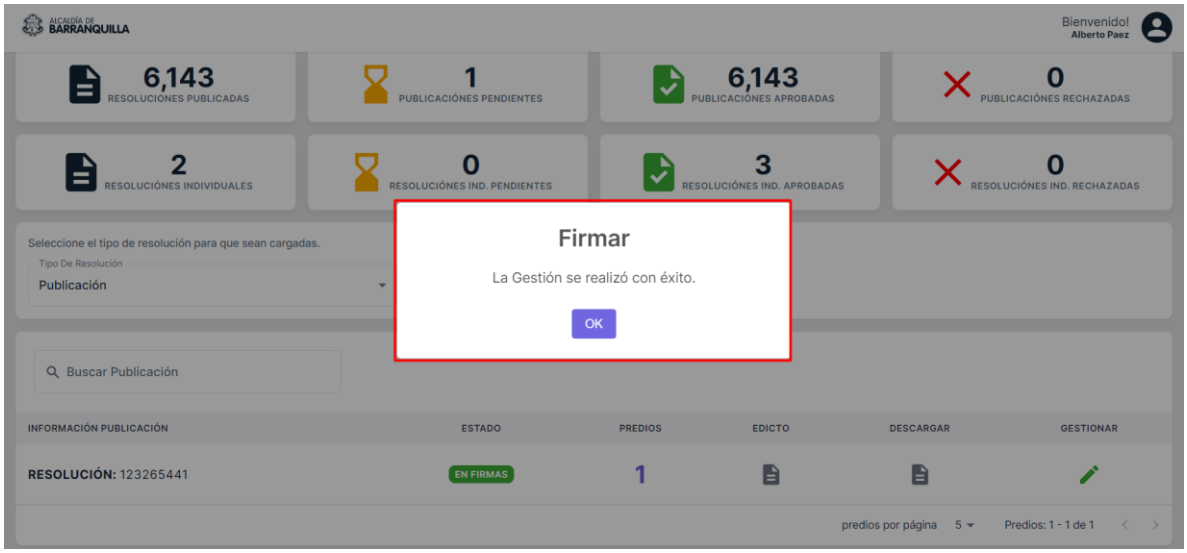
Se visualiza el primer nivel de aprobador.



Confirmar si está seguro de firmar resolución.



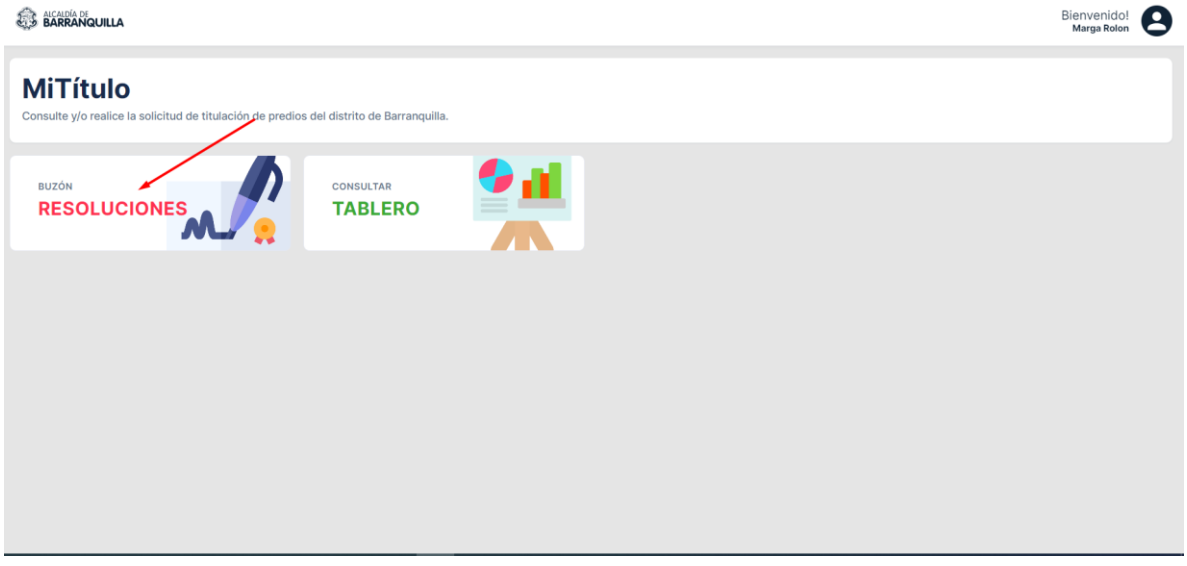
Se visualiza firma realizada.



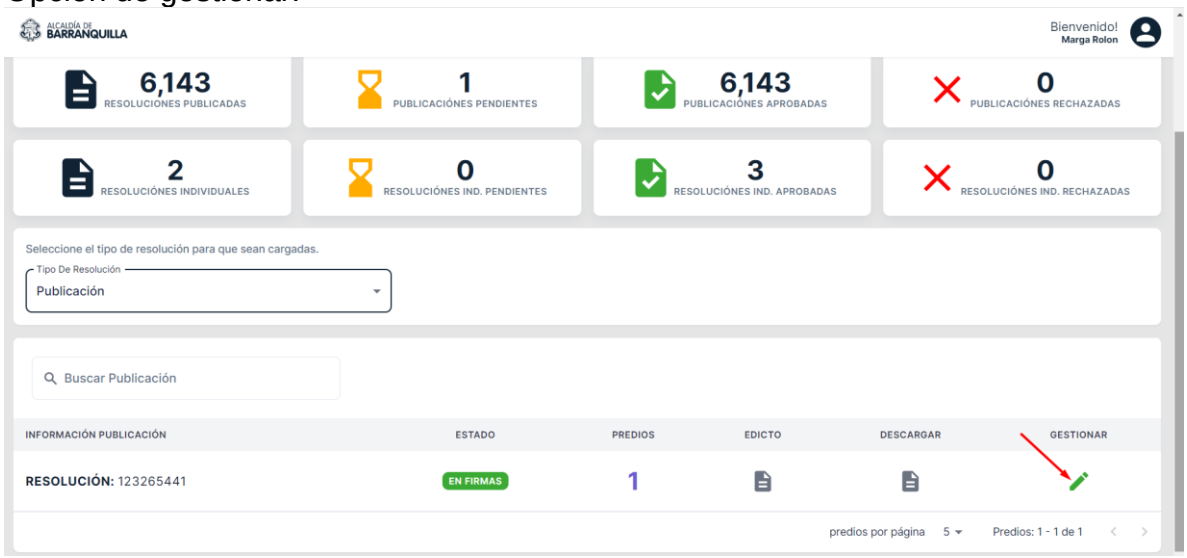
Se ingresa con el usuario Marga Rolón.



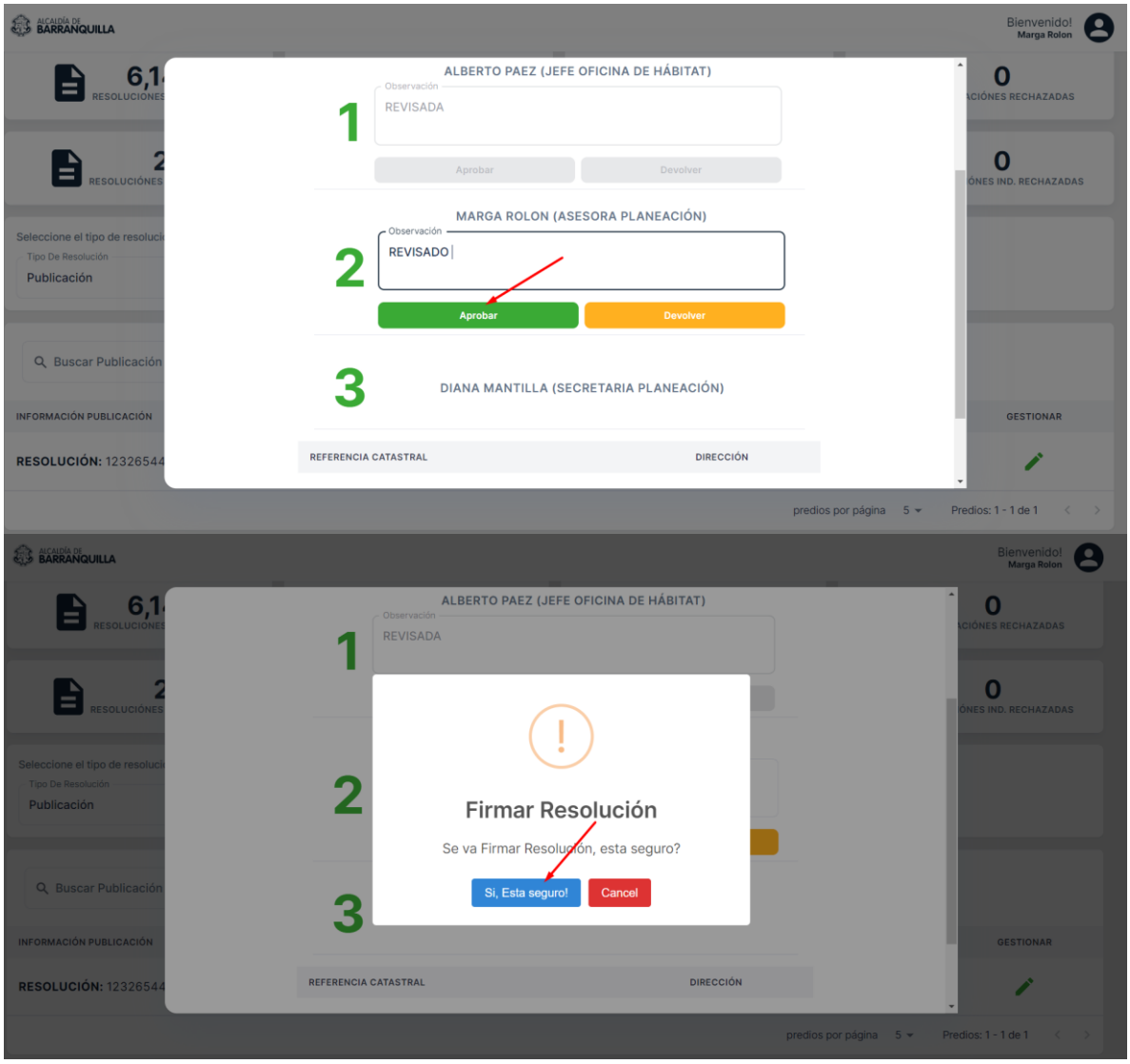
Buzón de resolución.



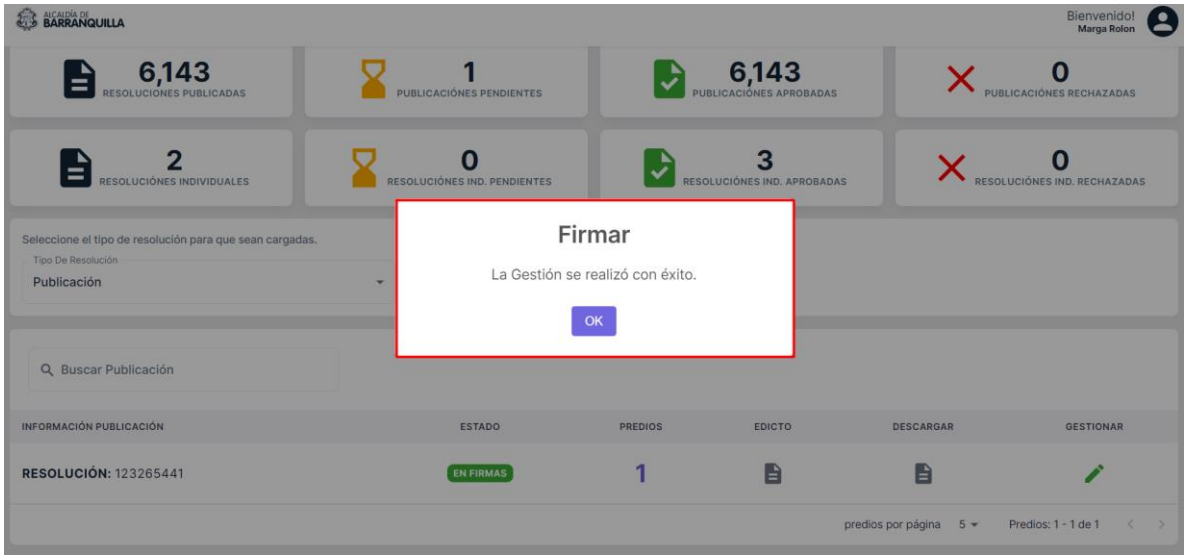
Opción de gestionar.



Se escribe una observación y click en el botón aprobar.



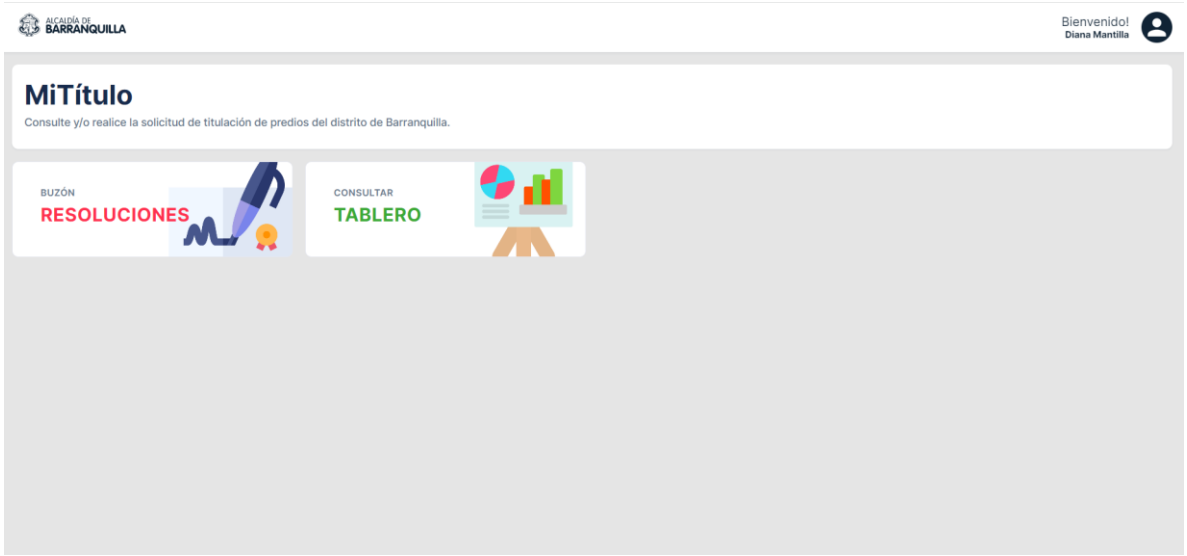
Gestión se realizó con éxito.



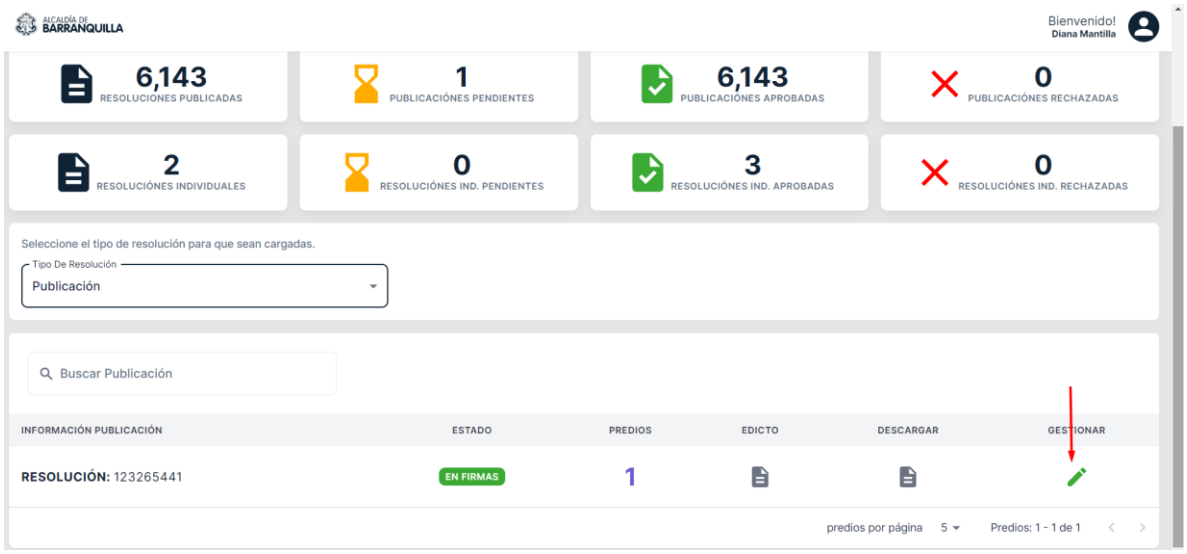
Se ingresa con el usuario diana mantilla.



Buzón de resolución.



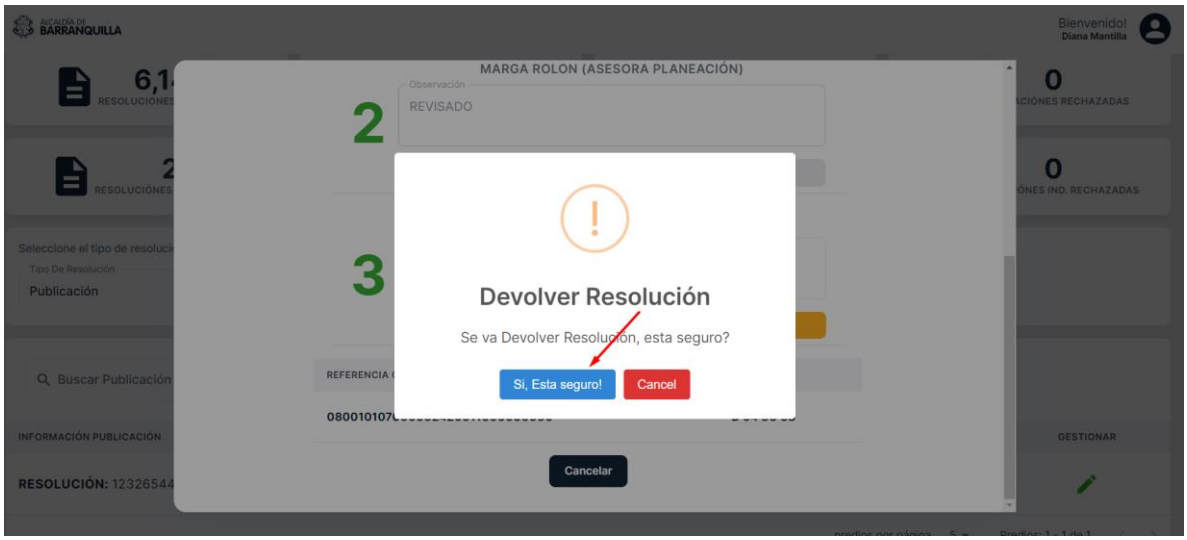
Gestionar.



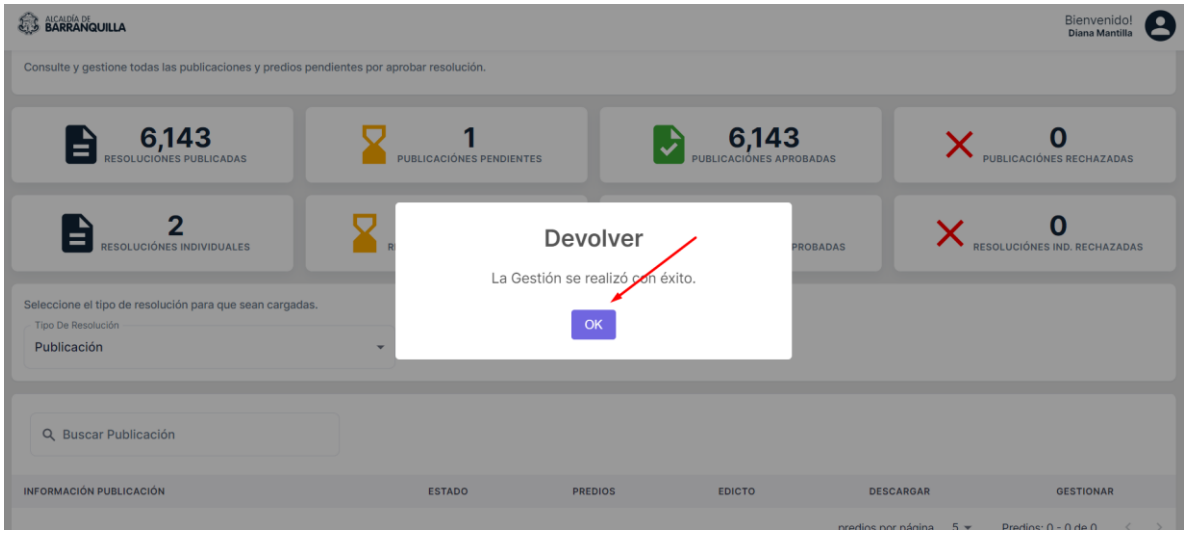
En este nivel, el usuario va a realizar una devolución.



Si está seguro de realizar la devolución.



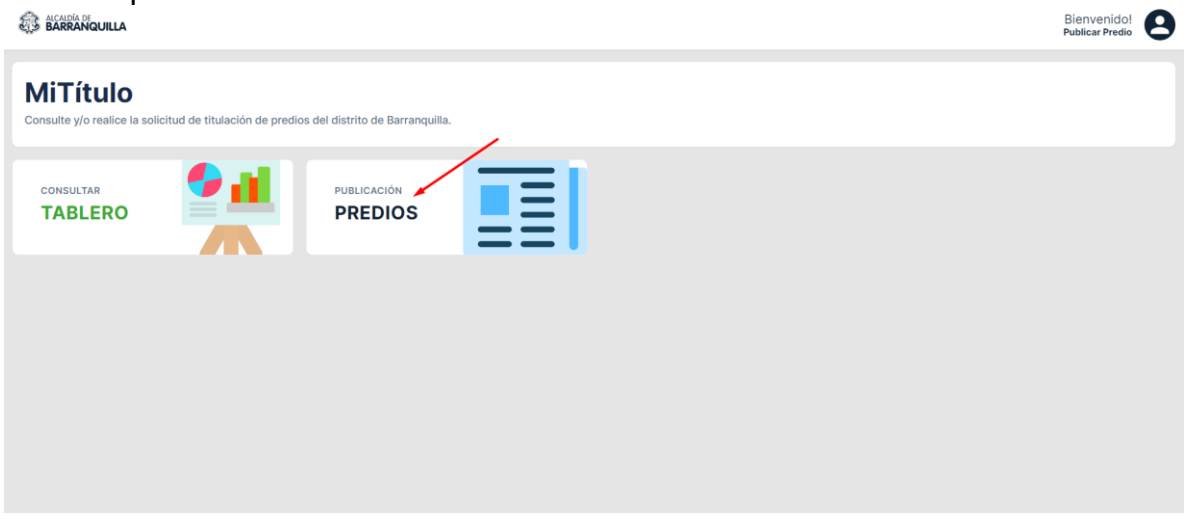
Gestión se realizó con éxito.



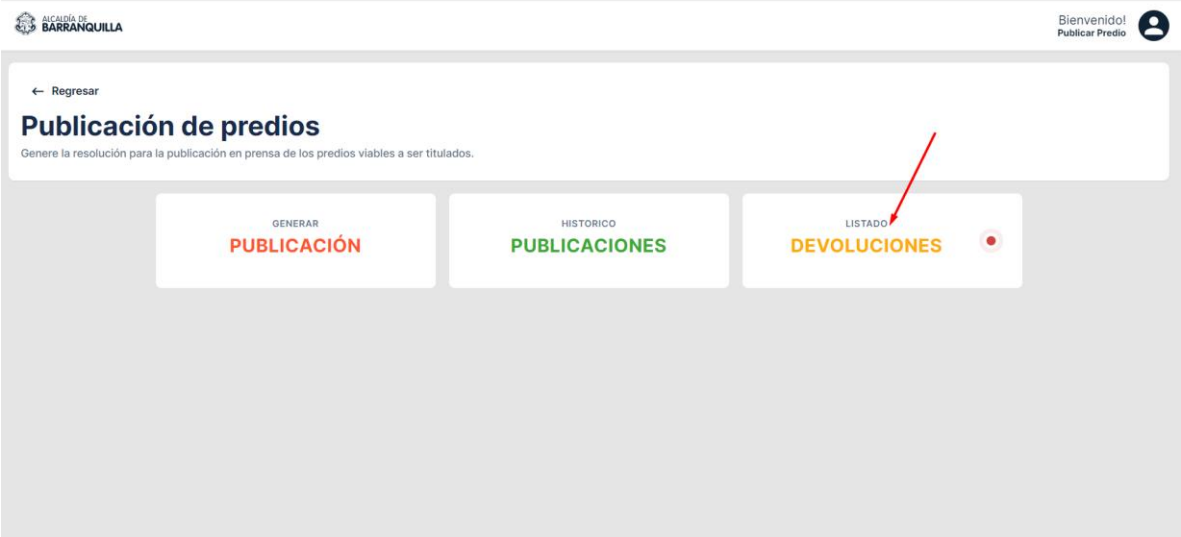
Se ingresa con el usuario publicar.



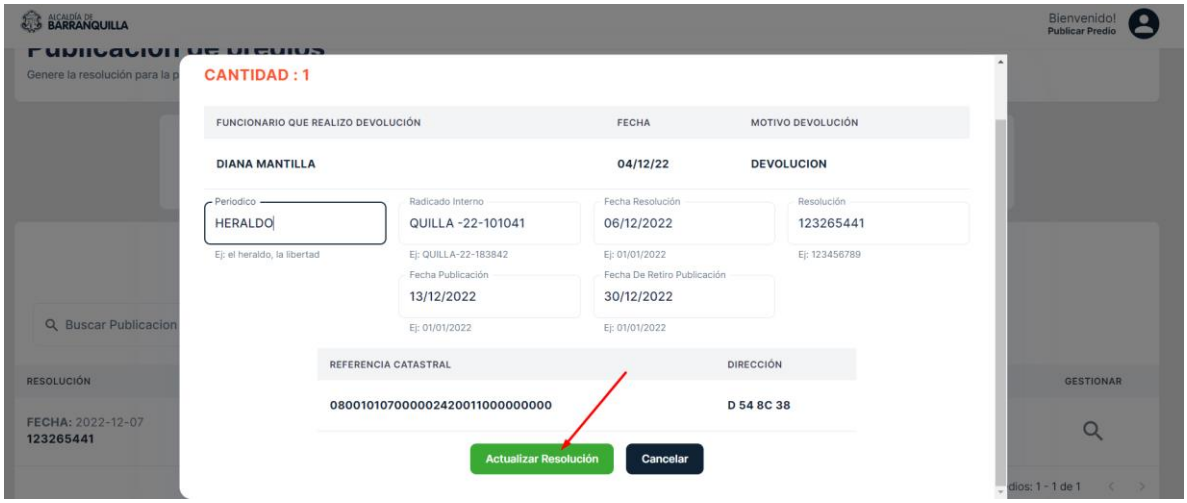
Publicar predio.



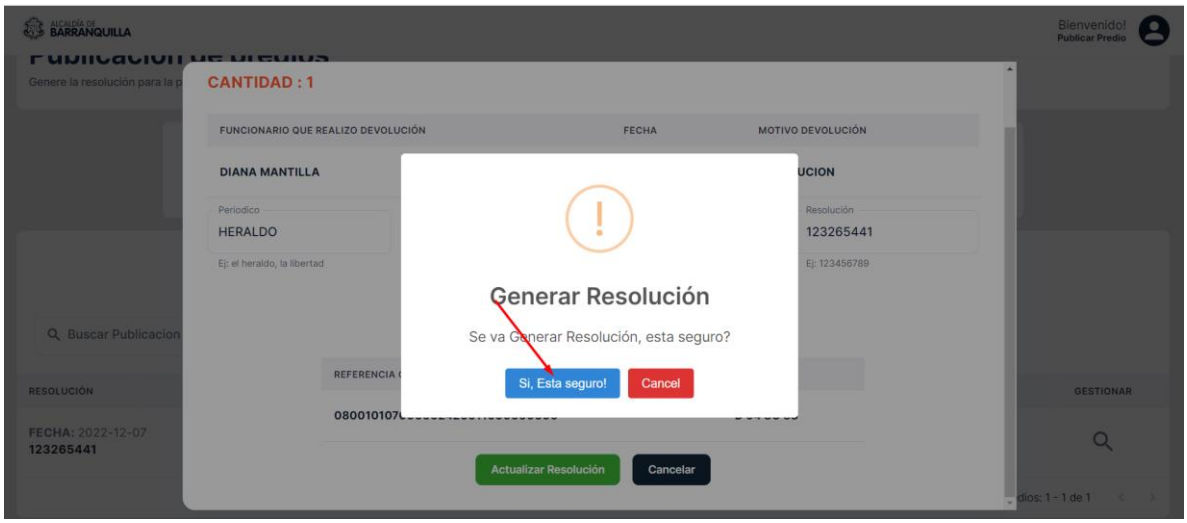
Listado de devolución.



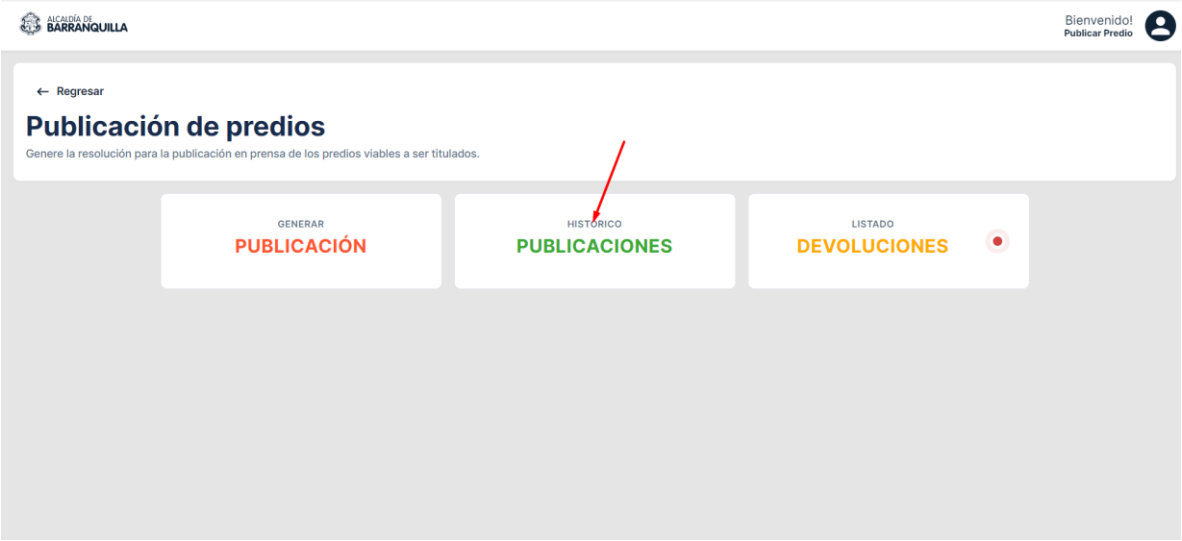
Se edita el nombre del periodo.



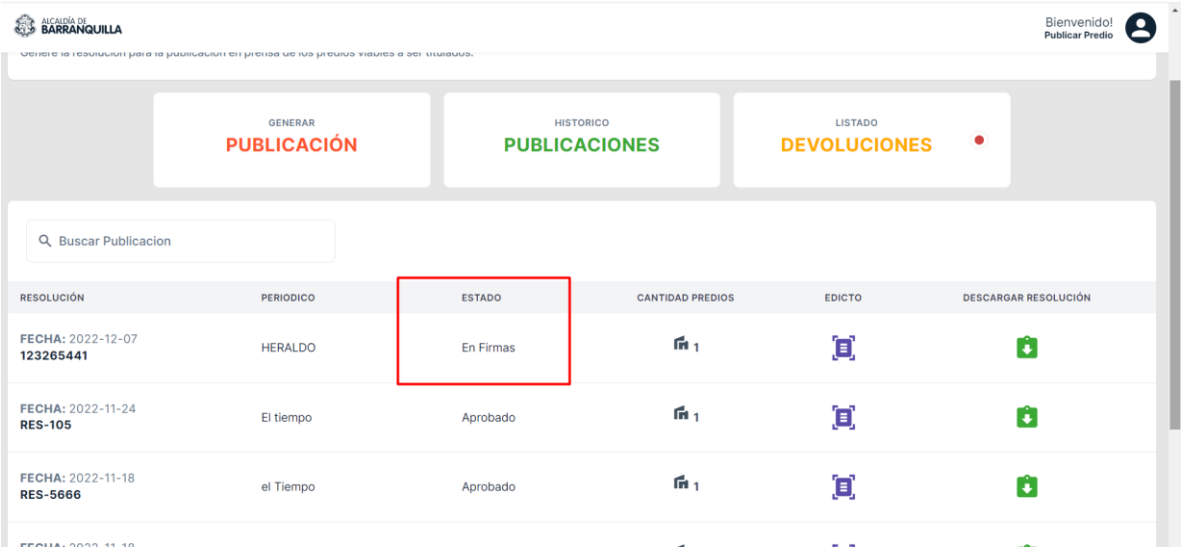
Generar resolución.



Histórico de publicación.



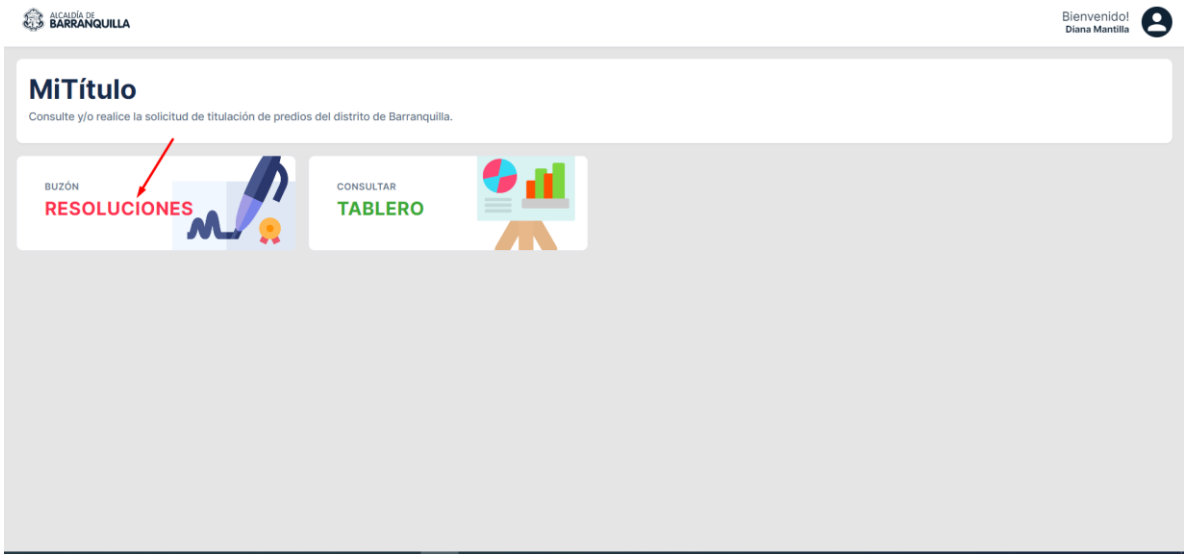
Se visualiza que se encuentra en estado firmado.



Se ingresa con el usuario diana mantilla.



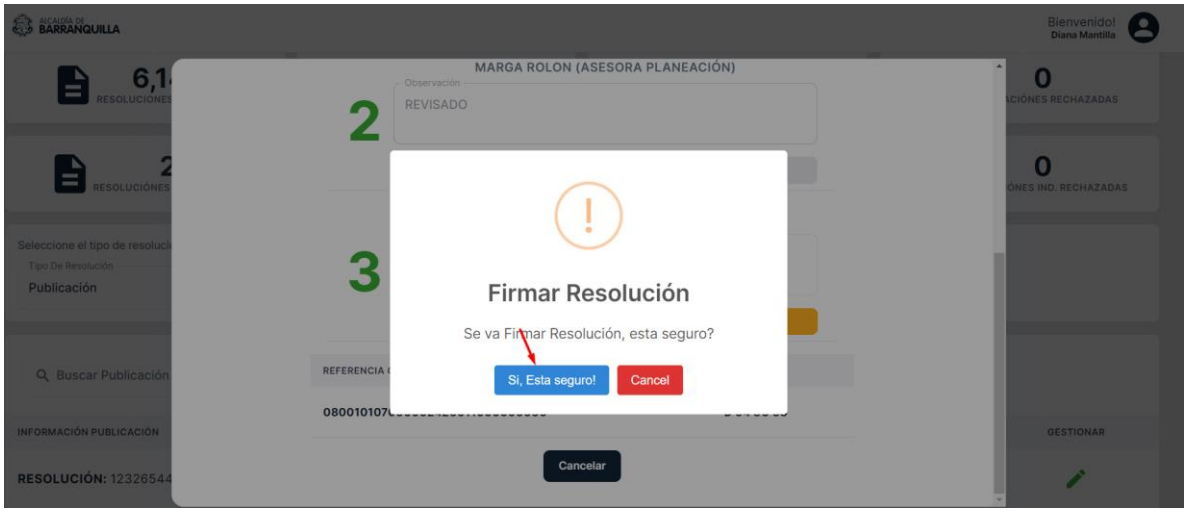
Buzón de resolución.



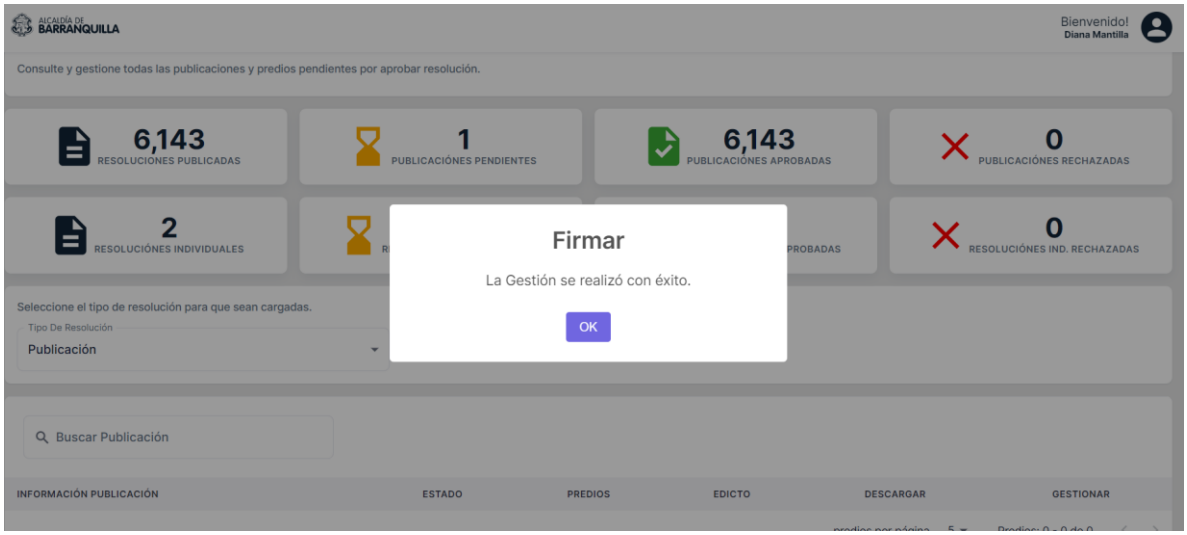
Se escribe una observación y aprobar.



Confirmación, si está seguro de firmar resolución.



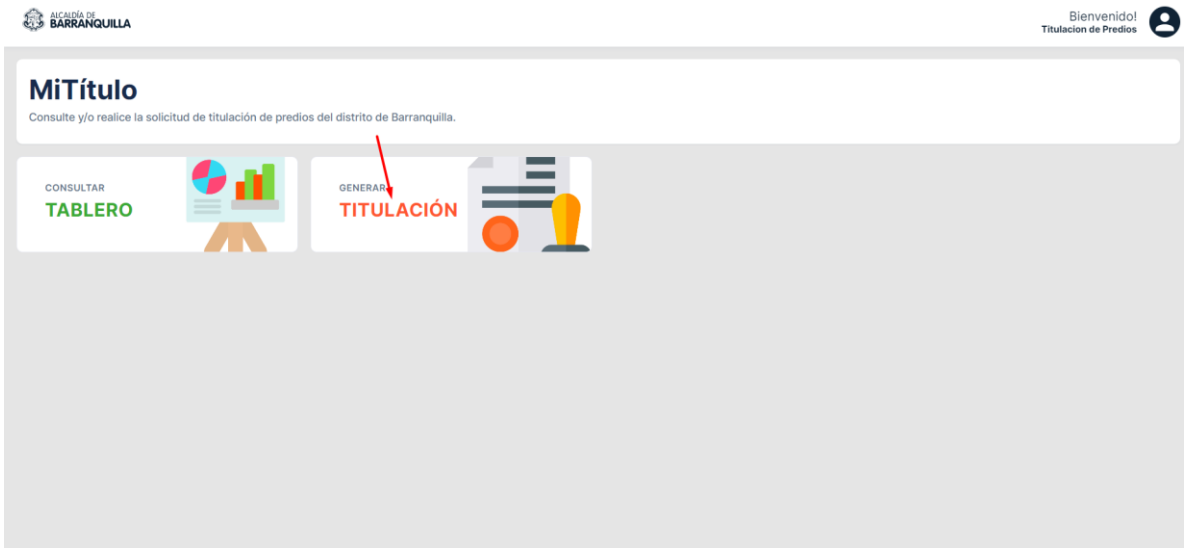
Firma realizada.

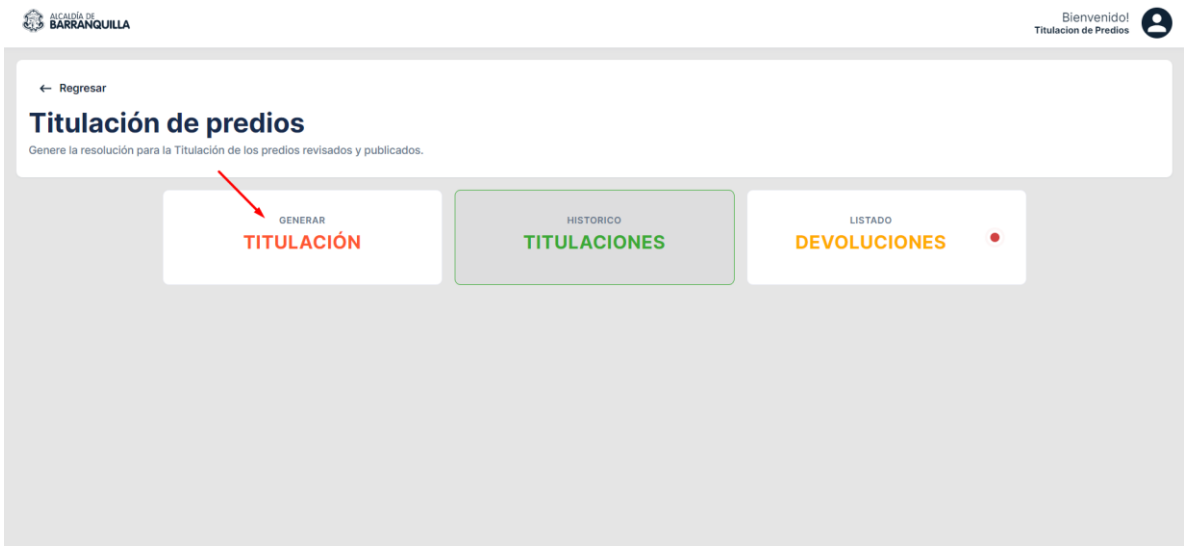


Se ingresa con el usuario de titulación.



Generar titulación.





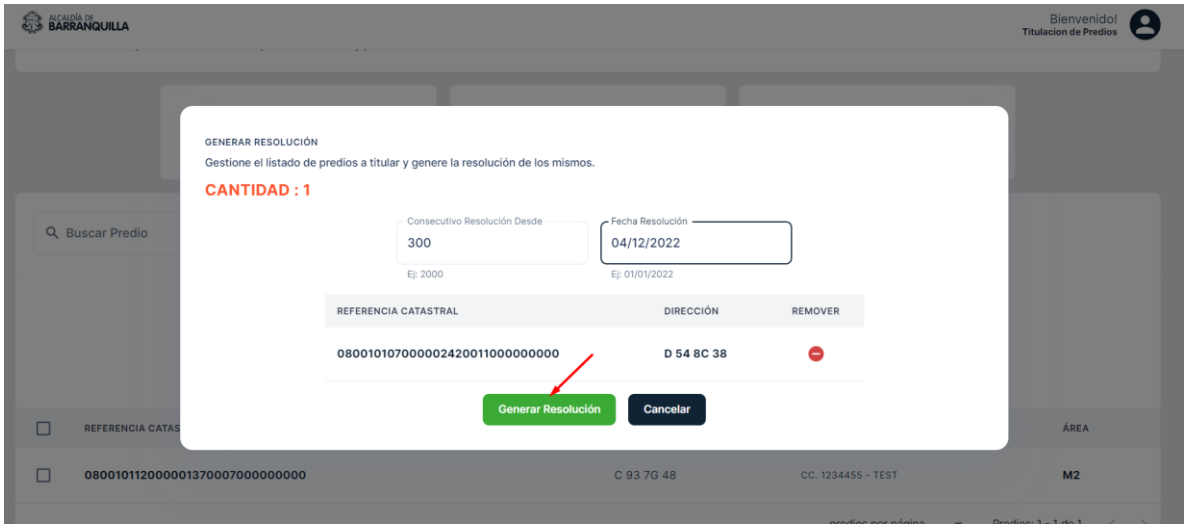
Agregar listado.



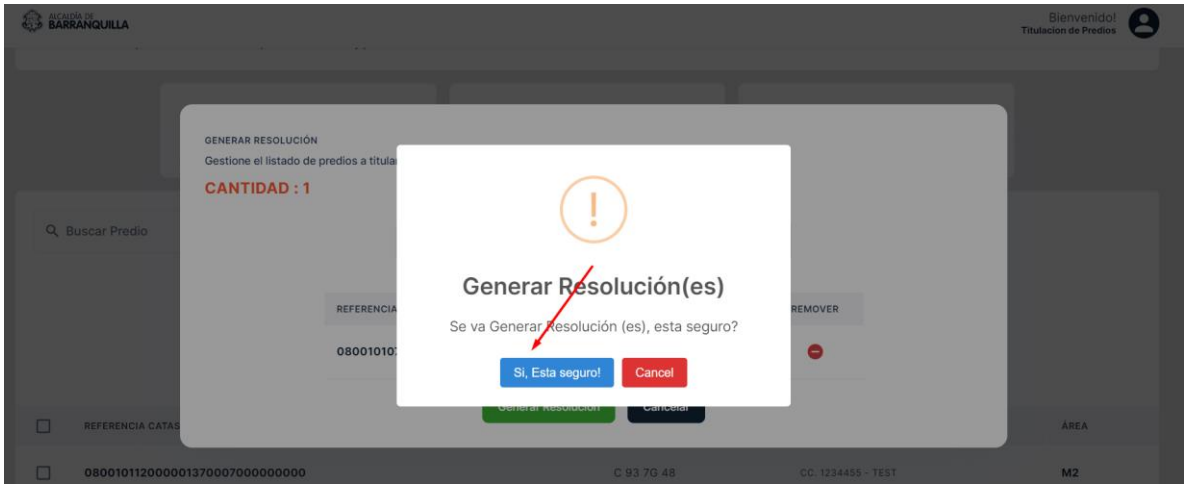
Adicionar predio.



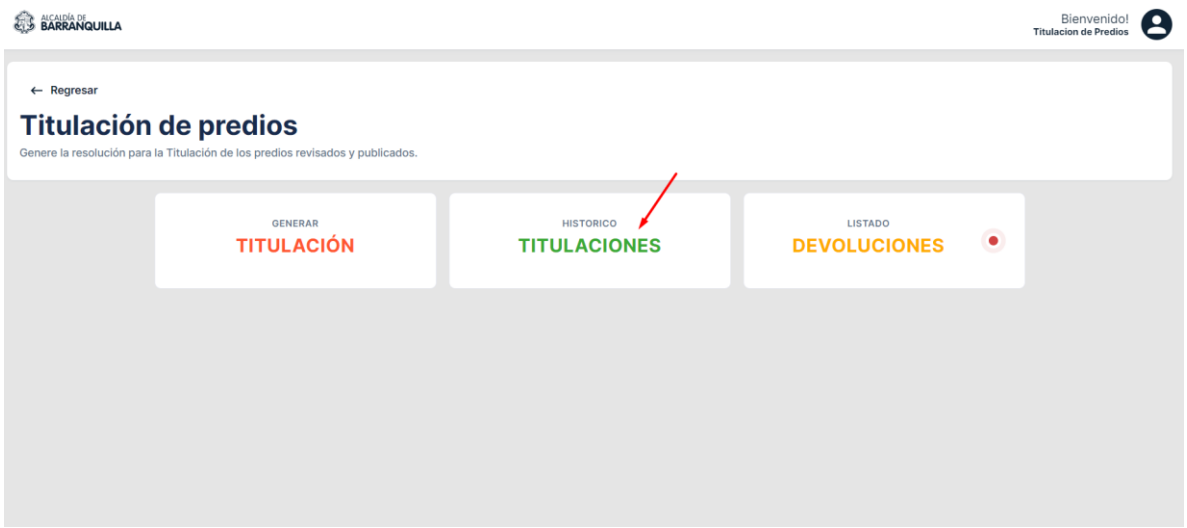
Generar resolución.



Confirmación si está seguro de generar predio.



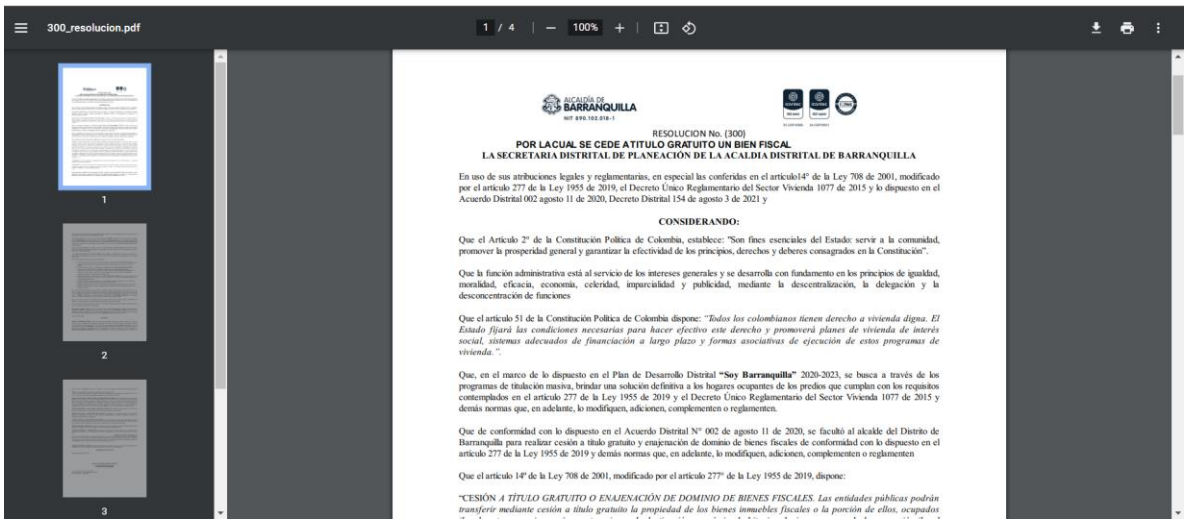
Histórico de titulación.



Descarga.



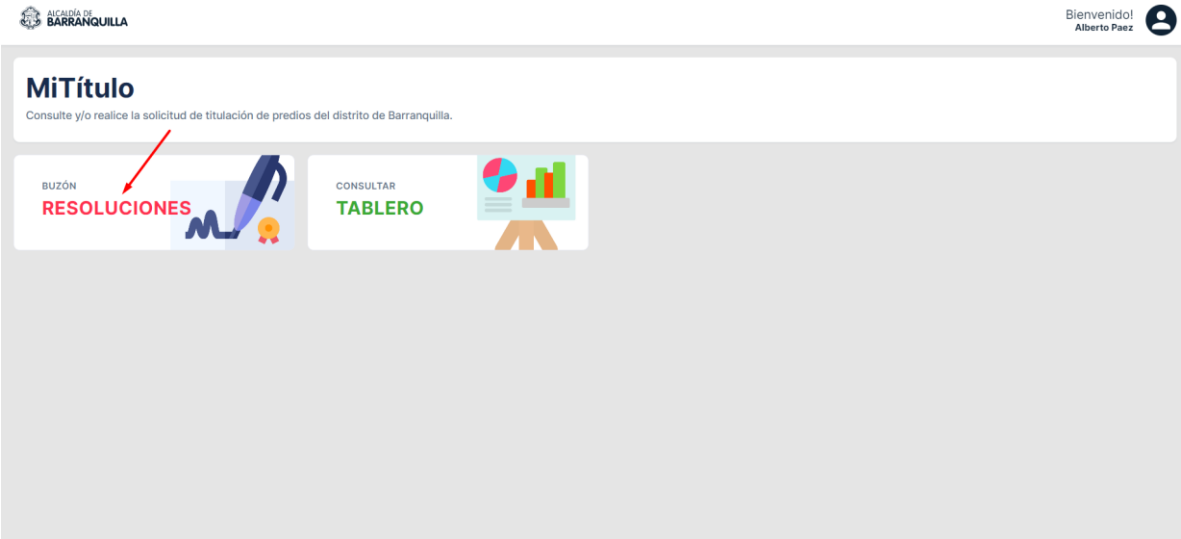
Se visualiza descarga del PDF.



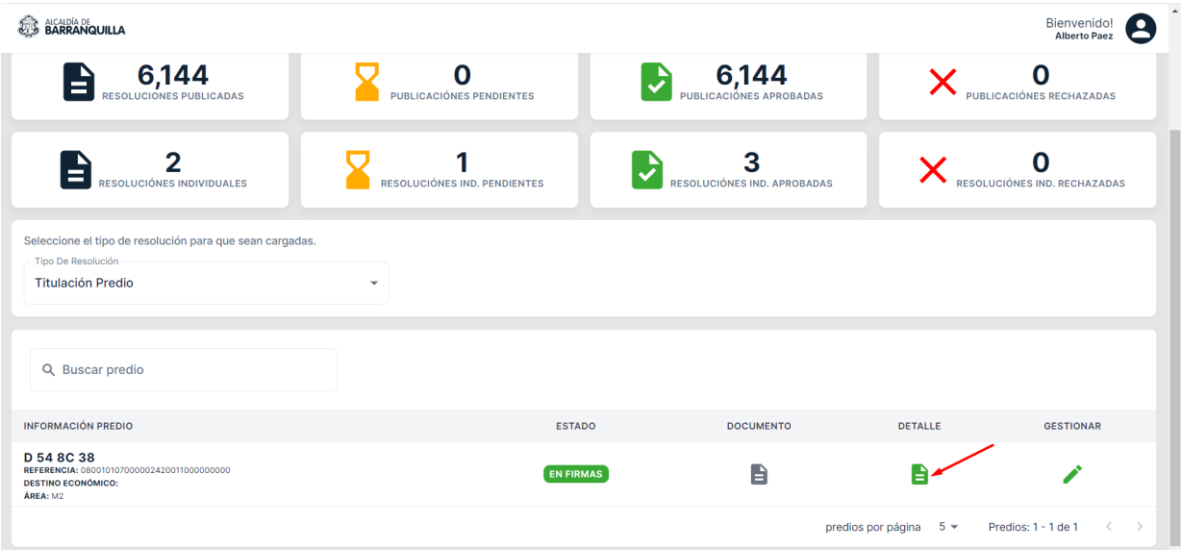
Se ingresa con el usuario Alberto Páez.

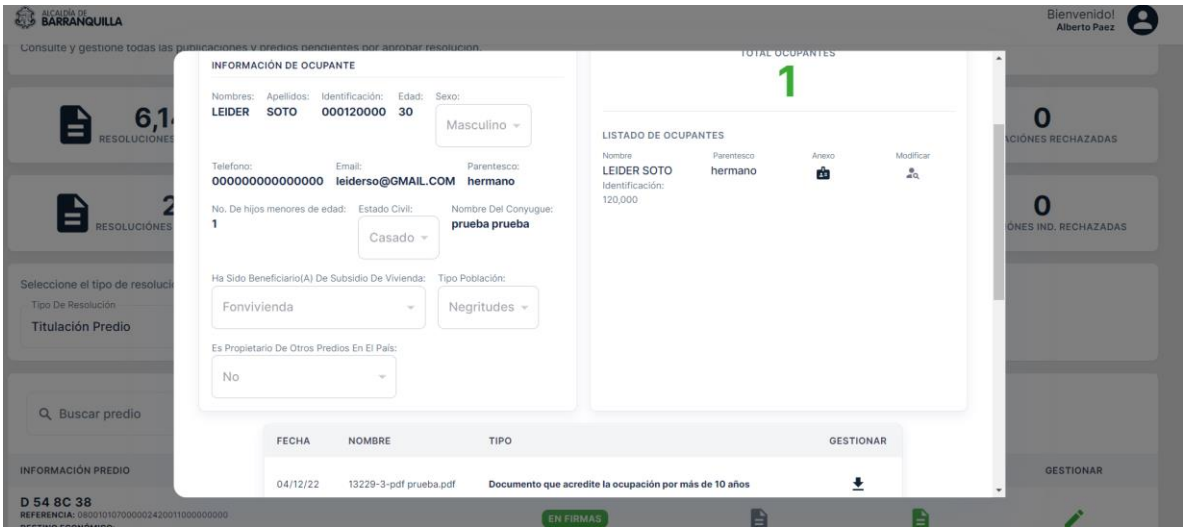


Buzón de resolución.

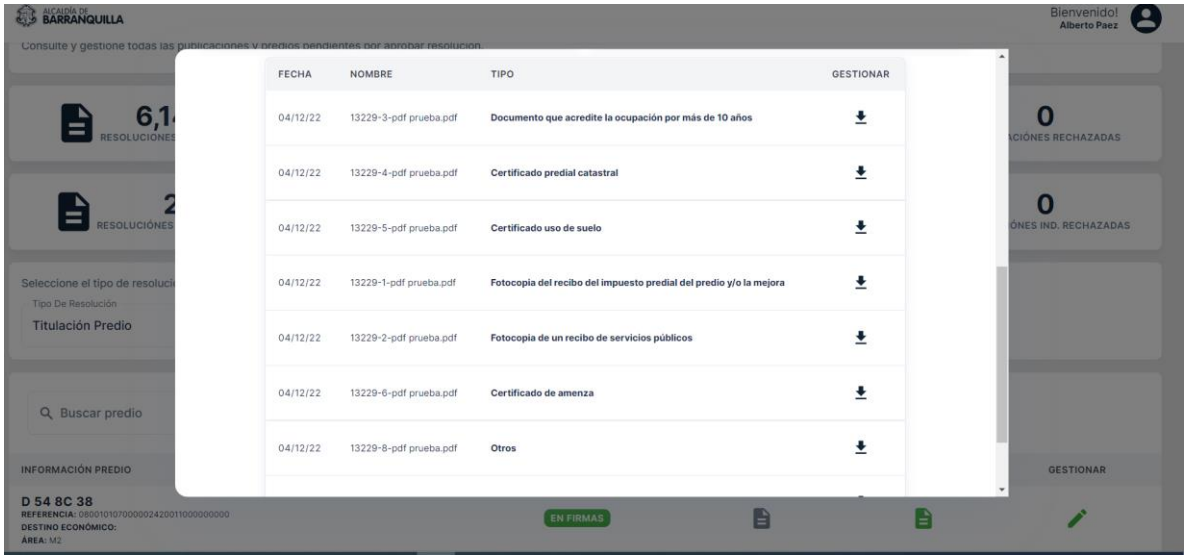


Se visualiza el detalle.

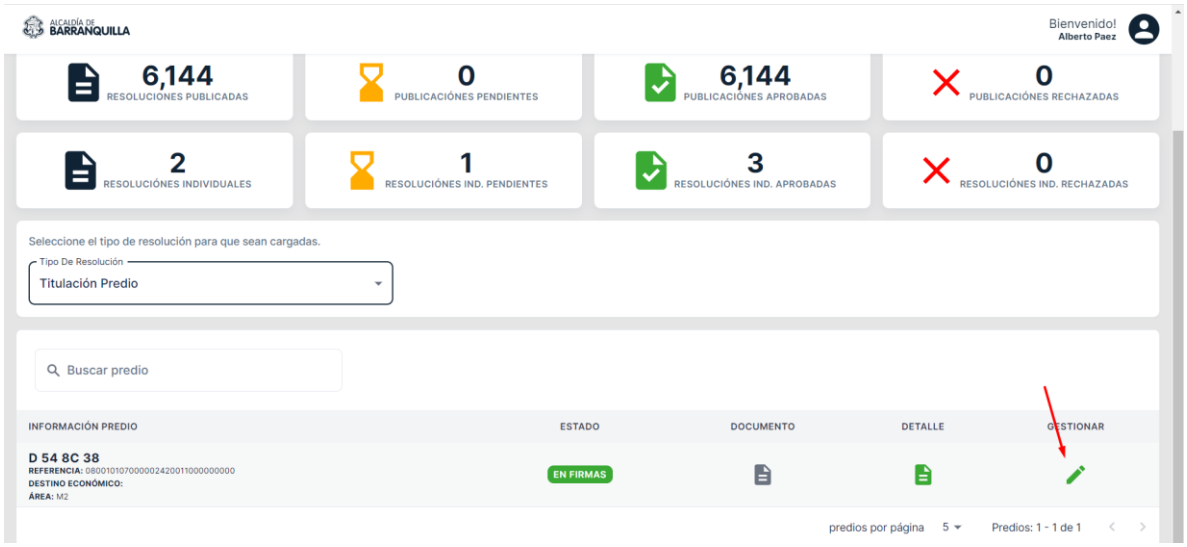




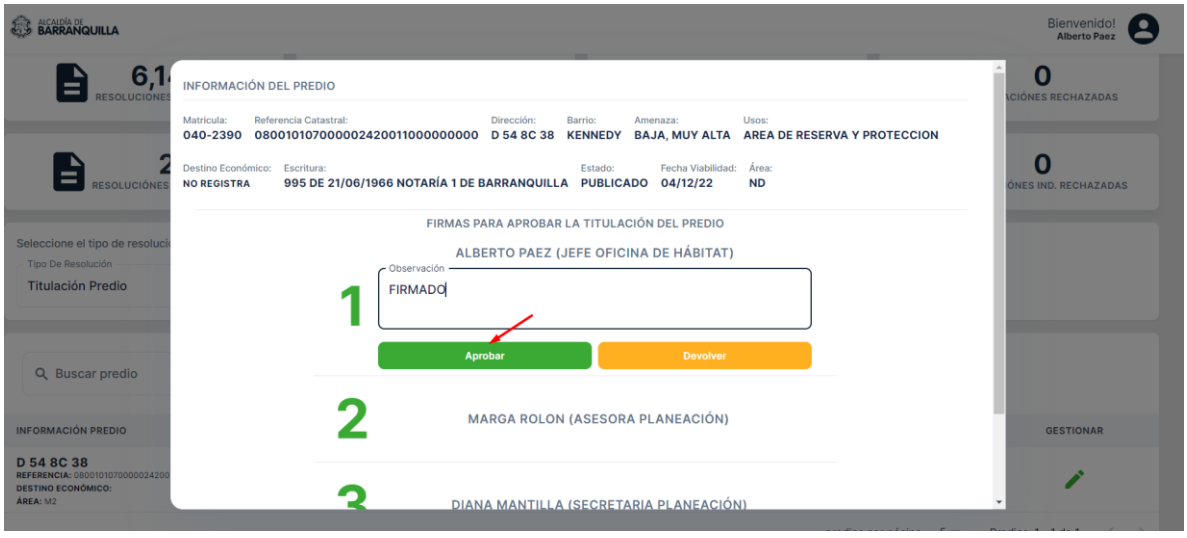
Se visualiza el listado de documentos que se gestionó.



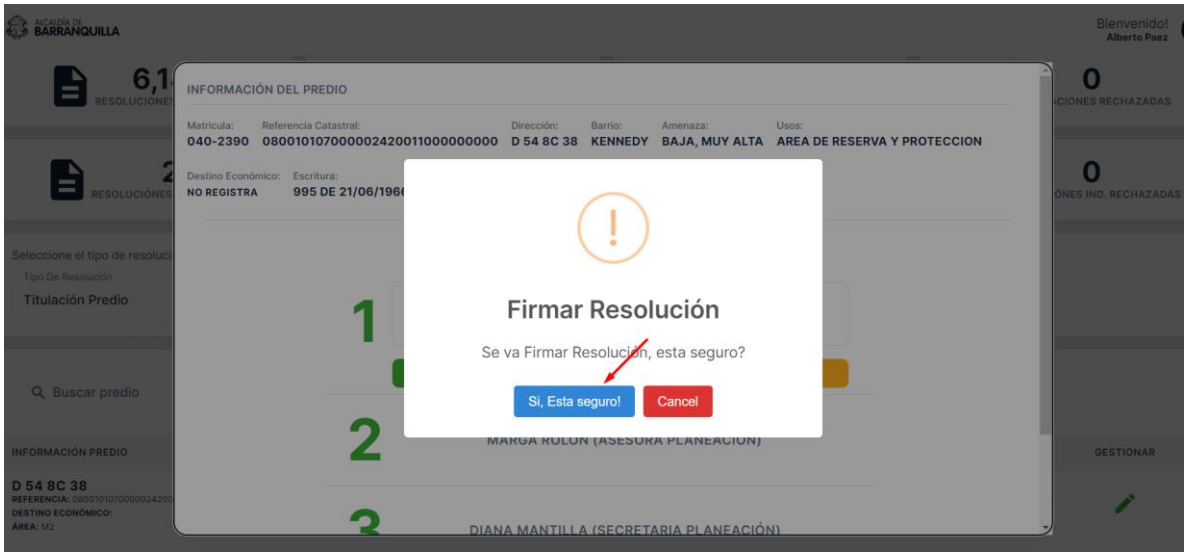
Gestionar.



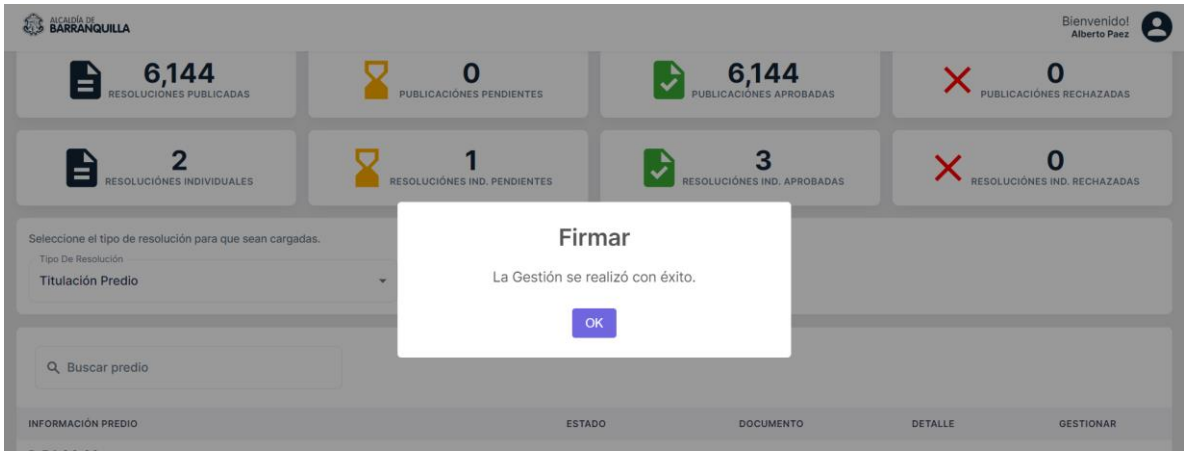
Click en el botón aprobar.



Firmar resolución.



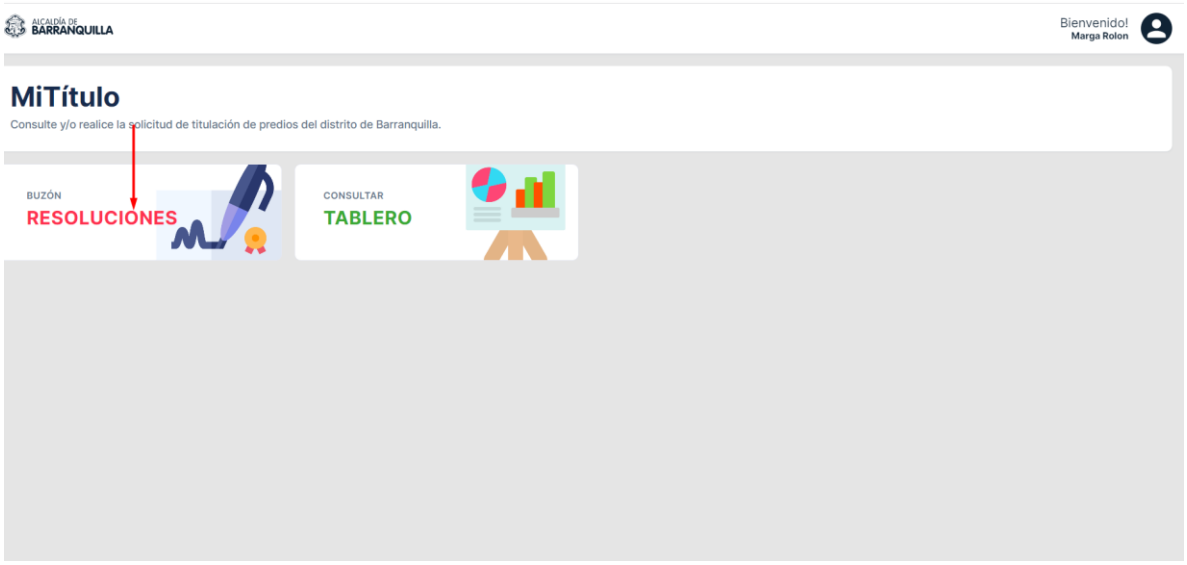
Gestión se realizó con éxito.



Se ingresa con el usuario Marga Rolón.



Buzón de resolución.



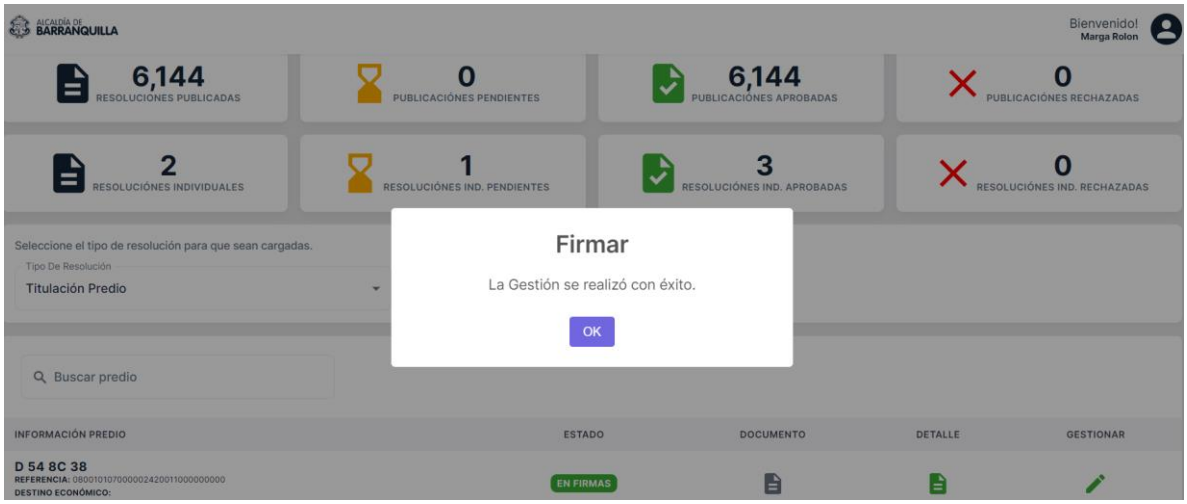
Click en aprobar.



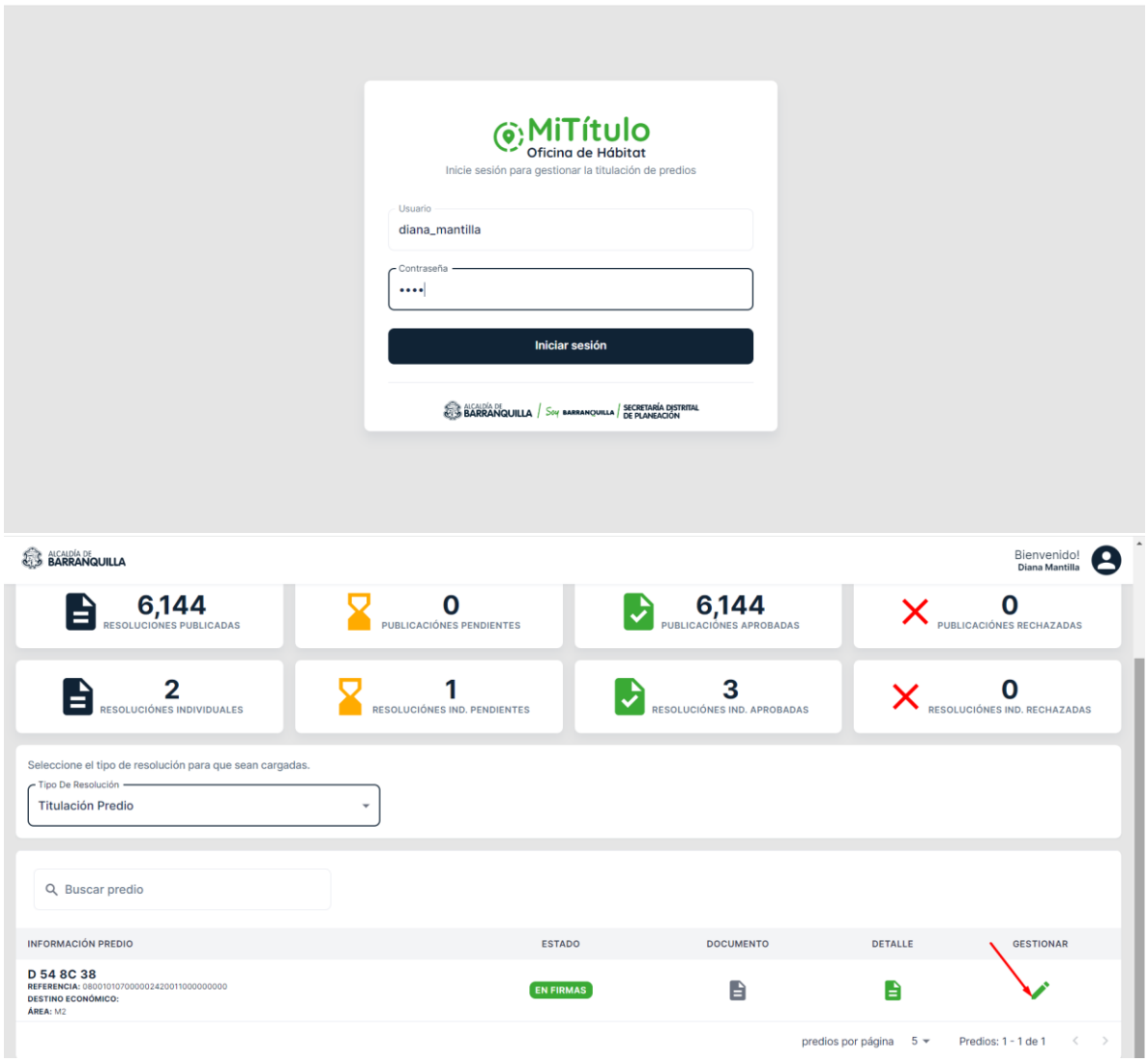
Confirmar si está seguro de firmar la resolución.



Se evidencia que la firma se realizó con éxito.



Se ingresa con el usuario diana mantilla.



Se escribe una observación y click en aprobar.



Firma de resolución.



Evidencias técnicas.

src/components/dashboard/buzon/buzon-inicio.js

```
@@ -6/8,6 +6/8,7 @@ export const BuzonResoluciones = () => {
            </Grid>
            <Grid
              item
+             display='none'
            >
              <Button
                variant="contained"
```

src/components/dashboard/predio/predio-lista-proceso.js

```
@@ -213,7 +213,8 @@ const GenerarResolcuion = async () => {
            'RES_FECHA':resolDate,
            'RES_FECHA_FIJADO':startDate,
            'RES_FECHA_DESFIJADO':endDate,
-           'LISTA_REF':selectedCustomers
+           'LISTA_REF':selectedCustomers,
+           'RES_ACTUALIZAR':0
          }
```

src/components/dashboard/predio/predio-lista-validacion.js

```
                    item
-                   sm={4}
+                   sm={6}
                    xs={12}
                  >
                    <Button size="small" variant="contained" color="success" fullWidth disabled >Aprobar</Button>
                  </Grid>
                  <Grid
                    item
-                   sm={4}
+                   sm={6}
                    xs={12}
                  >
                    <Button size="small" variant="contained" color="warning" fullWidth  disabled>Devolver</Button>
                  </Grid>
-                 <Grid
-                   item
-                   sm={4}
-                   xs={12}
-                 >
-                   <Button size="small" variant="contained" color="error" fullWidth  disabled>Rechazar</Button>
-                 </Grid>
+
               </Grid>
                 }

@@ -1728,25 +1722,19 @@ const ListarCategoriaRelacion = async (idprediosel) => {

                  <Grid
                    item
-                   sm={4}
+                   sm={6}
```

```
                    >
                      <Button size="small" variant="contained" color="success" fullWidth onClick={() => firmar(1)}
  >Aprobar</Button>
                    </Grid>
                    <Grid
                      item
-                     sm={4}
+                     sm={6}
                      xs={12}
                    >
-                     <Button size="small" variant="contained" color="warning" fullWidth  onClick={() => firmar(2)}
  >Devolver</Button>
-                   </Grid>
-                   <Grid
-                     item
-                     sm={4}
-                     xs={12}
-                   >
-                     <Button size="small" variant="contained" color="error" fullWidth onClick={() => firmar(3)}
  >Rechazar</Button>
+                     <Button size="small" variant="contained" color="warning" fullWidth  onClick={() => firmar(3)}
  >Devolver</Button>
                    </Grid>
+
                 </Grid>
                    }
               </Grid>
```

src/components/dashboard/predio/predio-lista.js

```
  @@ -426,9 +426,7 @@ const abrirMapa = async (ref, dir) => {
                    <IconButton component="a" onClick={() => abrirMapa(customer.PRE_REFERENCIA,
  customer.PRE_DIRECCION)} color="primary">
                      <ModeOfTravelIcon fontSize="small" />
                    </IconButton>
-                   <IconButton component="a" onClick={() => setModalId(2)} color="primary">
-                     <ModeOfTravelIcon fontSize="small" />
-                   </IconButton>
+
                 </TableCell>
                 <TableCell>
                   <Grid
```

src/components/dashboard/publicacion/publicacion-devoluciones.js

```
@@ -141,6 +141,7 @@ export const PublicacionDevoluciones = () => {

  const [pageCantidad, setPageCantidad] = useState(1);

+
   const getCustomers = useCallback(async () => {
     let data ={
       'ESC_NUM_ESCRI':'ESCRITURA',
@@ -175,7 +176,7 @@ export const PublicacionDevoluciones = () => {

     try {
       axios
-      .get(`${process.env.NEXT_PUBLIC_AUTH0_DOMAIN_SERVER}mit_predio/lista-resoluciones/0`, {}, config)
+      .get(`${process.env.NEXT_PUBLIC_AUTH0_DOMAIN_SERVER}mit_predio/lista-resoluciones/3`, {}, config)
       .then(({ data: res }) => {
         setResoluciones(res);

@@ -195,6 +196,7 @@ export const PublicacionDevoluciones = () => {

   }, []);

+

   useEffect(() => {
       getCustomers();
```

src/components/dashboard/publicacion/publicacion-lista-devoluciones.js

```
@@ -41,6 +41,7 @@ import { Scrollbar } from '../../scrollbar';
 import { MobileDatePicker } from '@mui/lab';
 import { FileDropzone } from '../../file-dropzone';
 import Swal from 'sweetalert2'
+import AssignmentReturnedIcon from '@mui/icons-material/AssignmentReturned';

 const config = {
   headers: { Authorization: `Bearer 123` }
@@ -109,7 +110,14 @@ export const ListaPublicacionDevoluciones = (props) => {
   const [periodico, setPeriodico] = useState();
   const [radicadoInterno, setRadicadoInterno] = useState();
   const [resolucion, setResolucion] = useState();
+  const [cantidad, setCantidad]  = useState(0);

+  const [listarSelectedCustomers, setListarSelectedCustomers] = useState([]);
+
+  const [listarMotivosDev, setlistarMotivosDev] = useState([]);
+
+
+  const [resolucionGenerada, setResolucionGenerada]  = useState(0);

   const handleResolucionChange = (event) => {
     event.preventDefault();
@@ -176,13 +184,57 @@ const handleDrop  = async ([file]) => {

 };

+const ListarMotivoDevolucion = async (RES_ID) => {
+ try {
+   axios
+   .get(`${process.env.NEXT_PUBLIC_AUTH0_DOMAIN_SERVER}mit_resolucion/get-motivos-dev-publica/${RES_ID}`, {}, config)
+   .then(({ data: res }) => {
+      setlistarMotivosDev(res);
```

```
+    })
+    .catch(error => {
+      console.log(error);
+    });
+
+  } catch (err) {
+    console.error(err);
+  }
+ }
+ const ListarSeleccionadosResolucionInd = async (RES_ID) => {
+   setResolucionGenerada(RES_ID);
+  try {
+    axios
+    .get(`${process.env.NEXT_PUBLIC_AUTH0_DOMAIN_SERVER}mit_resolucion/get-resolucion-unico/${RES_ID}`, {}, config)
+    .then(({ data: res }) => {
+       setCantidad(res.length);
+       ListarMotivoDevolucion(RES_ID);
+       setListarSelectedCustomers(res);
+       setPeriodico(res[0].RES_PERIODICO);
+       setRadicadoInterno(res[0].RES_NUM_RADICADO);
+       setResolucion(res[0].RES_NUM_RSOLUCION);
+       setResolDate(res[0].RES_FECHA);
+       setStartDate(res[0].RES_FECHA_FIJADO);
+       setEndDate(res[0].RES_FECHA_DESFIJADO);
+
+       setModalId(1);
+
+    })
+    .catch(error => {
+      console.log(error);
+    });
+
+ } catch (err) {
 && resolucion &&  resolucion!=''  && resolucion!="undefined" )
 {
       Swal.fire({
        title: 'Generar Resolución',
@@ -207,7 +259,8 @@ const GenerarResolcuion = async () => {
             'RES_FECHA_FIJADO':startDate,
             'RES_FECHA_DESFIJADO':endDate,
             'RES_EDICTO':nombre_archivo,
-            'LISTA_REF':selectedCustomers
+            'LISTA_REF':selectedCustomers,
+            'RES_ACTUALIZAR':resolucionGenerada
           }

           if(nombre_archivo != ""){
@@ -244,7 +297,7 @@ const GenerarResolcuion = async () => {
           type: "success"
         });
         setSelectedCustomers([]);
-        window.location = `${process.env.NEXT_PUBLIC_AUTH0_DOMAIN_SERVER}mit_predio/descarga-
anexo/`+res.data+`_resolusion.pdf`;
        window.location.href = `${process.env.NEXT_PUBLIC_URL_SITIO}`+'publicacion/consultar';

          } )

@@ -303,6 +356,9 @@ const GenerarResolcuion = async () => {
             <TableCell align="center">
               Edicto
             </TableCell>
+            <TableCell align="center">
+              Descargar Resolución
+            </TableCell>
             <TableCell align="center">
```

```
+                                >
+                                  {customer.USU_NOMBRE}
+                                </Typography>
+
+                              </Grid>
+                            </TableCell>
+                            <TableCell>
+                              <Grid
+                                container
+                                direction="column"
+                              >
+
+                                <Typography
+                                  color="primary"
+                                  fontWeight="bold"
+                                  variant="overline"
+                                  fontSize={14}
+                                >
+                                  {customer.APR_FECHA_REGISTRO}
+                                </Typography>
+
+                              </Grid>
+                              </TableCell>
+                            <TableCell>
+                              <Grid
+                                container
+                                direction="column"
+                              >
+
+                                <Typography
+                                  color="primary"
+                                  fontWeight="bold"
+                                  variant="overline"
@@ -451,6 +612,7 @@ const GenerarResolcuion = async () => {
                fullWidth
                value={periodico}
                onChange={handlePeriodicoChange}
+
              />
            </Grid>
            <Grid
@@ -560,60 +722,65 @@ const GenerarResolcuion = async () => {
                      <TableCell>
                        Dirección
                      </TableCell>
-                     <TableCell>
-                       Remover
+                     <TableCell align="center">
+                       Remover
                      </TableCell>
+
                    </TableRow>
                  </TableHead>
+
+
                  <TableBody>
-                   {listarSelectedCustomers.map((customer) => {
-                     const isCustomerSelected = selectedCustomers.includes(customer.PRE_ID);
-                     return (
-                       <TableRow
-                         hover
-                         key={customer.PRE_ID}
-                         selected={isCustomerSelected}
-                       >
-                         <TableCell>
-                           {customer.USU_NOMBRE}
-                         </TableCell>
-                         <TableCell>
-                           {customer.APR_FECHA_REGISTRO}
-                         </TableCell>
-                         <TableCell>
-                           {customer.PRE_DIRECCION}
-                         </TableCell>
-                         <TableCell>
-                           <Button
-                             color="primary"
-                             onClick={() => handleRemoveCustomer(customer.PRE_ID)}
-                             variant="contained"
-                           >
-                             Remover
-                           </Button>
-                         </TableCell>
-                       </TableRow>
-                     );
-                   })}
+                   <TableRow
+                     hover
+                   >
+                     <TableCell>
+                       <Grid
+                         container
+                         direction="column"
+                       >
+                   {listarSelectedCustomers.map((customer) => {
+                     const isCustomerSelected = selectedCustomers.includes(customer.PRE_ID);
+                     return (
+                       <TableRow
+                         hover
```

```
+                          >

-                          <Typography
-                            color="primary"
-                            fontWeight="bold"
-                            variant="overline"
-                            fontSize={14}
-                          >
-                            REFERENCIA
-                          </Typography>
+                            <Typography
+                              color="primary"
+                              fontWeight="bold"
+                              variant="overline"
+                              fontSize={14}
+                            >
+                              {customer.PRE_REFERENCIA}
+                            </Typography>

-                        </Grid>
-                      </TableCell>
-                      <TableCell>
-                        <Grid
-                          container
-                          direction="column"
-                        >
-                          <Typography
-                            color="primary"
-                            fontWeight="bold"
-                            variant="overline"
-                            fontSize={14}
-                          >
-                            DIRECCION
+                              direction="column"
+                            >
+
+                              <Typography
+                                color="primary"
+                                fontWeight="bold"
+                                variant="overline"
+                                fontSize={14}
+                              >
+                                {customer.PRE_DIRECCION}
+                              </Typography>
+
+                            </Grid>
+                          </TableCell>
+                        </TableRow>
+                      );
+                    })}
                    </TableBody>
+
+
                  </Table>
                </Scrollbar>
              </Grid>
              <Grid item>
-               <Button variant="contained" color="secondary">Generar Resolución</Button>
+               <Button variant="contained" color="secondary"  onClick={() => GenerarResolcuion()}>Actualizar
  Resolución</Button>
              </Grid>
              <Grid item>
                <Button variant="contained" onClick={() => setModalId(0)}>Cancelar</Button>
```

leider soto

LEIDER ANDRES SOTO BETIN
CC: 78.300.913 de Montelíbano