**SISTEMA INTEGRAL BANCARIO**

**Manual Técnico**

CONTENIDO

[1. Introducción 3](#_Toc4397382)

[Objetivos 3](#_Toc4397383)

[2. Características 4](#_Toc4397384)

[Tipo 4](#_Toc4397385)

[Lenguaje 4](#_Toc4397386)

[Seguridad 4](#_Toc4397387)

[Bases de datos 4](#_Toc4397388)

[Módulos iniciales 4](#_Toc4397389)

[3. Aplicación WEB 5](#_Toc4397390)

[Estructura del proyecto 5](#_Toc4397391)

[Nómina de archivos 7](#_Toc4397392)

[Flujo de información 9](#_Toc4397393)

[4. Base de datos 11](#_Toc4397394)

[Modulo de Usuarios 12](#_Toc4397395)

[Modulo de Garantias 13](#_Toc4397396)

[Modulo de reportes 14](#_Toc4397397)

[5. Archivos de configuración 15](#_Toc4397398)

# **Introducción**

El Sistema Integral Bancario es una aplicación web escalable y flexible que va a permitir la integración de múltiples funcionalidades agrupadas para diferentes tipos de usuarios.

SIB permite a usuarios autenticados realizar las tareas que tenga asignadas según el tipo de usuario que posea. Esto permite, a partir de una base de datos centralizada, brindar múltiples vistas de un mismo sistema integrado. La división de módulos es una de las principales características de este sistema.

**Objetivos**

Entre los principales objetivos, se destacan los siguientes:

* + - Automatización de actividades.
    - Consulta de información actualizada.
    - Bajo tiempo de respuesta.
    - Flexibilidad y simplicidad.
    - Integración modular.

# Características

A continuación se detallan las características básicas del aplicativo:

## Tipo

**Aplicación web**. Accesible desde cualquier equipo del banco mediante un navegador web.

## Lenguaje

**PHP.** Además está diseñada utilizando HTML, CSS y JavaScript**.**

## Seguridad

**Mixta**. Se utiliza la autenticación de usuarios mediante active directory. Por lo cual, todo usuario que se encuentre dentro del dominio configurado, previa autenticación, podrá acceder al aplicativo.

## Bases de datos

**Sí**. Cuenta con una base de datos propia y conexiones externas según la necesidad de cada módulo.

## Módulos iniciales

El **Sistema Integral Bancario**, en su primera versión (**1.0.0**), cuenta con los siguientes módulos:

* Gestión de Usuarios.
* Gestión de Garantías.
* Gestión de Reportes.

# Aplicación WEB

SIB consta de una página web que se encuentra montada sobre IIS (Internet Information Services).

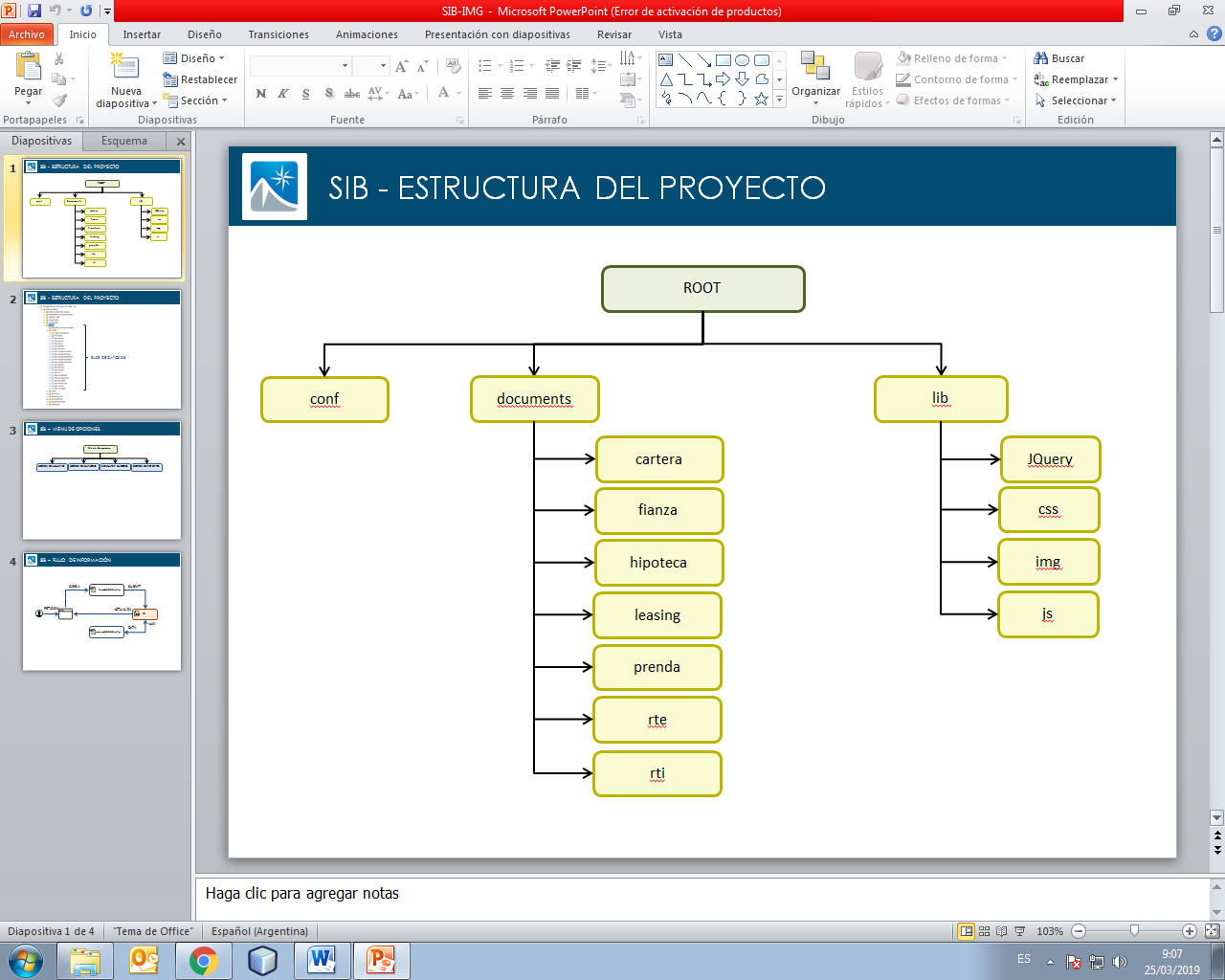
El servidor donde se encuentra la página web es **VM250IIS03** en desarrollo y el motor de base de datos SQL SERVER. La base de datos se encuentra en **192.168.250.144** en desarrollo bajo el nombre de “**BD\_SIB**”.

|  |  |  |
| --- | --- | --- |
|  | **Desarrollo** | **Producción** |
| SIB | 192.168.250.186 |  |
| BD SIB | 192.168.250.144 |  |

SIB está construido con seguridad integrada a través de Active Directory. Esto quiere decir que solo los usuarios autorizados tendrán acceso al sitio web. Además, cuenta con una capa de seguridad interna para filtrar a los usuarios según el rol que posean. Para ello, SIB administra usuarios, roles y permisos en forma interna e independiente de la seguridad integrada.

## Estructura del proyecto

La estructura del proyecto SIB desarrollado con NetBeans IDE 8.2 es la siguiente:



Donde ROOT indica la raíz de la estructura y contiene todos los archivos PHP del código. A su vez, se encuentra sub-dividido en tres carpetas:

* **CONF**: Contiene los archivos PHP que se encargan de la configuración y la conexión a la base de datos. Estos archivos son Conexión, Configuración y config.
* **DOCUMENTS**: Contiene las carpetas donde se almacenan de forma local todos los archivos procesados por el sistema.
  + **Cartera**: PDF garantías de tipo cartera.
  + **Fianza**: PDF garantías de tipo fianza.
  + **Hipoteca**: PDF garantías de tipo hipoteca.
  + **Leasing**: PDF garantías de tipo leasing.
  + **Prenda**: PDF garantías de tipo RTE.
  + **RTE**: Archivos XML de RTE en PF.
* **LIB**: Este directorio se encuentra dividido en cuatro partes:
  + **JQUERY**: Contiene la librería JQUERY junto con los archivos JavaScript que permiten el funcionamiento de la aplicación.
  + **CSS**: Contiene los archivos necesarios para dar estilos a la página web. En ella se encuentran los CSS correspondientes a Bootstrap y DataTables.
  + **IMG**: Contiene las imágenes utilizadas en el proyecto.
  + **JS**: Contiene los archivos JavaScript para la utilización de Bootstrap y DataTables.

## Nómina de archivos

En esta sección se detallan cada uno de los archivos que son parte del ROOT del proyecto.

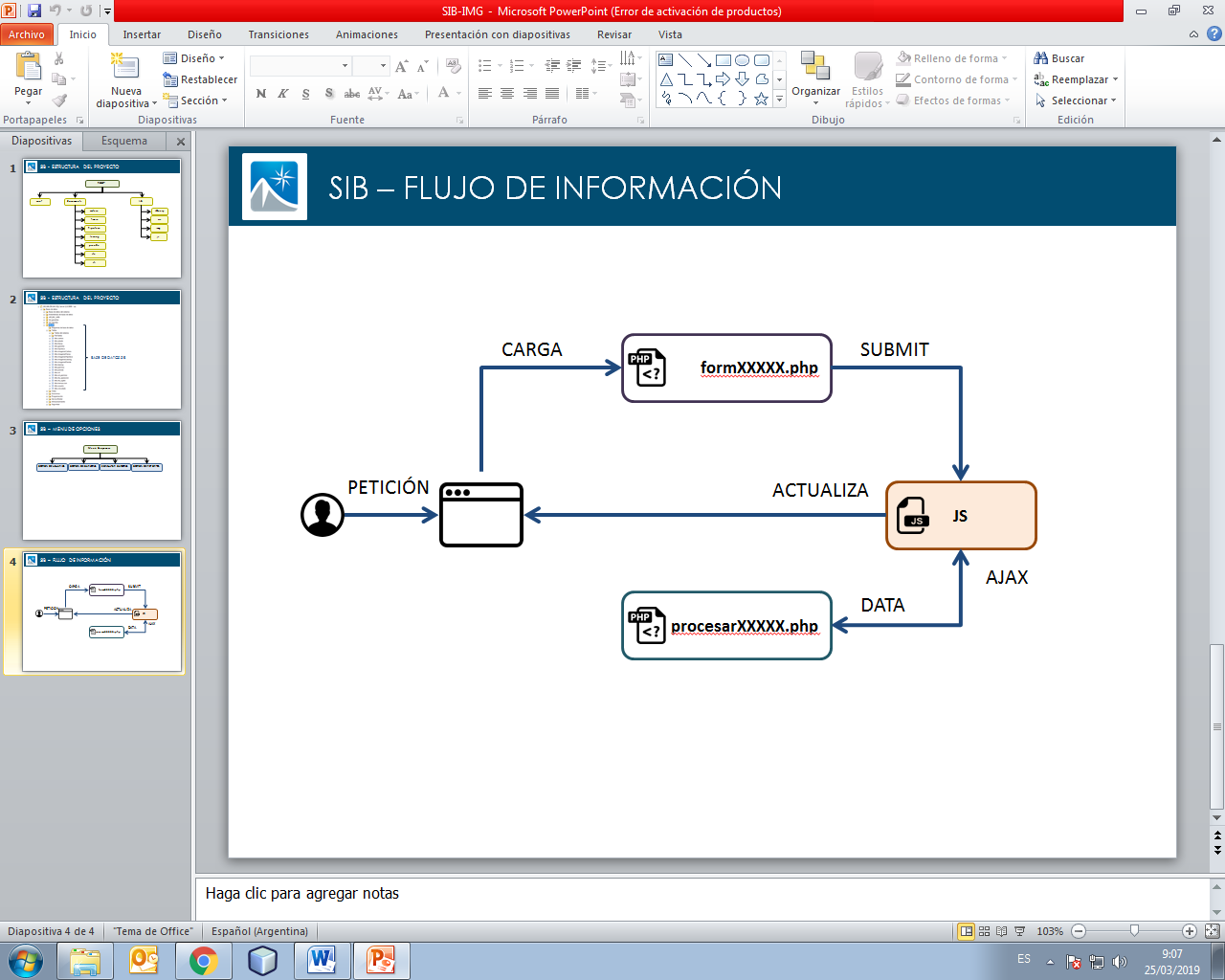
|  |  |
| --- | --- |
| **Archivo** | **Descripción** |
| formBuscarEstados.php | Permite realizar la búsqueda de los estados cargados en la BD y mostrarlos en una tabla. También permite crear estado, borrar o modificar. |
| formBuscarGaratia.php | Permite realizar la búsqueda de garantías según el filtro ingresado. |
| formBuscarPermisos.php | Permite realizar la búsqueda de los permisos cargados en la BD y mostrarlos en una tabla. También permite crear permiso, borrar o modificar. |
| formBuscarRTE.php | Permite realizar la búsqueda de los RTE cargados en la BD y mostrarlos en una tabla. También permite borrar, modificar y generar XML. |
| formBuscarRoles.php | Permite realizar la búsqueda de los roles cargados en la BD y mostrarlos en una tabla. También permite crear rol. |
| formBuscarUsuario.php | Permite realizar la búsqueda de los usuarios cargados en la BD y mostrarlos en una tabla. También permite la creación, eliminación y modificación. |
| formConsultarGarantia.php | Permite realizar la consulta de garantías según el filtro ingresado. |
| formCrearRTE.php | Permite realizar la creación de un RTE. |
| formDetallesGarantia.php | Permite mostrar el detalle de una garantía seleccionada durante una búsqueda. |
| formModificarEstado.php | Permite mostrar los campos modificables de un estado seleccionado previamente durante una búsqueda. |
| formModificarGarantia.php | Permite mostrar los campos modificables de una garantía seleccionada previamente durante una búsqueda. |
| formModificarRTE.php | Permite mostrar los campos modificables de un RTE seleccionado previamente durante una búsqueda. |
| formModificarRol.php | Permite mostrar los campos modificables de un rol seleccionado previamente durante la búsqueda. |
| formModificarUsuario.php | Permite mostrar los campos modificables de un usuario previamente seleccionado durante una búsqueda. |
| guardarFormulario.php | Genera la consulta SQL para almacenar una garantía en la BD. |
| header.php |  |
| index.php |  |
| inicio.php |  |
| modalBorar.php | Modal de confirmación. |
| modalCargarEstado.php | Modal de alta estado. |
| modalCargarPermiso.php | Modal de alta permiso. |
| modalCargarRol.php | Modal de alta rol. |
| modalCargarUsuario.php | Modal de alta usuario. |
| principal3.php | Permite realizar la creación de una nueva garantía. |
| procesarBorrarEstado.php | Realiza el proceso para llevar a cabo la eliminación de un estado, previamente seleccionado, de la BD. |
| procesarBorrarPermiso.php | Realiza el proceso para llevar a cabo la eliminación de un permiso, previamente seleccionado, de la BD. |
| procesarBorrarRol.php | Realiza el proceso para llevar a cabo la eliminación de un rol, previamente seleccionado, de la BD. |
| procesarBorrarUsuario.php | Realiza el proceso para llevar a cabo la eliminación de un usuario, previamente seleccionado, de la BD. |
| procesarBuscarGarantia.php | Realiza el proceso para llevar a cabo la búsqueda de una garantía en la BD según el filtro ingresado. |
| procesarCargarEstado.php | Realiza el proceso para llevar a cabo la creación de un estado en la BD. |
| procesarCargarPermiso.php | Realiza el proceso para llevar a cabo la creación de un permiso en la BD. |
| procesarCargarRol.php | Realiza el proceso para llevar a cabo la creación de un rol en la BD. |
| procesarCargarUsuario.php | Realiza el proceso para llevar a cabo la creación de un usuario en la BD. |
| procesarConsultarGarantia.php | Realiza el proceso para llevar a cabo la búsqueda de una garantía en la BD según el filtro ingresado. |
| procesarCrearRTE.php | Realiza el proceso para llevar a cabo la creación de un RTE en la BD. |
| procesarDescargarRTE.php | Realiza el proceso para descargar un archivo XML almacenado en el directorio local. |
| procesarGenerarRTE.php | Realiza la generación de un archivo XML para un RTE previamente seleccionado durante la búsqueda. |
| procesarLogin.php | Realiza el proceso de búsqueda del usuario a través de LDAP y luego en la BD del sistema. |
| procesarModificarEstado.php | Realiza el proceso de modificación de un estado, previamente seleccionado, en la BD. |
| procesarModificarGarantia.php | Realiza el proceso de modificación de una garantia, previamente seleccionado, en la BD. |
| procesarModificarPermiso.php | Realiza el proceso de modificación de un permiso, previamente seleccionado, en la BD. |
| procesarModificarRTE.php | Realiza el proceso de modificación de un RTE, previamente seleccionado, en la BD. |
| procesarModificarRol.php | Realiza el proceso de modificación de un rol, previamente seleccionado, en la BD. |
| procesarModificarUsuario.php | Realiza el proceso de modificación de un usuario, previamente seleccionado, en la BD. |
| procesarSalir.php | Realiza la eliminación de los datos almacenados en la sesión, elimina la sesión y redirecciona a la página de ingreso. |

## Flujo de información

El funcionamiento básico de la aplicación (en forma general) es el siguiente:

1. Un usuario realiza una petición en el navegador.
2. Se carga el formulario solicitado donde el usuario trabaja. Luego hace la confirmación y se envía el mismo (SUBMIT).
3. El evento es capturado por el archivo JavaScript correspondiente haciendo uso de JQuery. Este archivo realiza las validaciones que correspondan y lanza una petición usando AJAX al archivo de procesamiento.
4. El archivo procesa la peticion solicitada y devuelve el resultado que AJAX procesa para actualizar la porción de la pantalla que corresponda.
5. El usuario visualiza el resultado de la operación.
6. El flujo se repite.

Este procesa se observa a continuación:

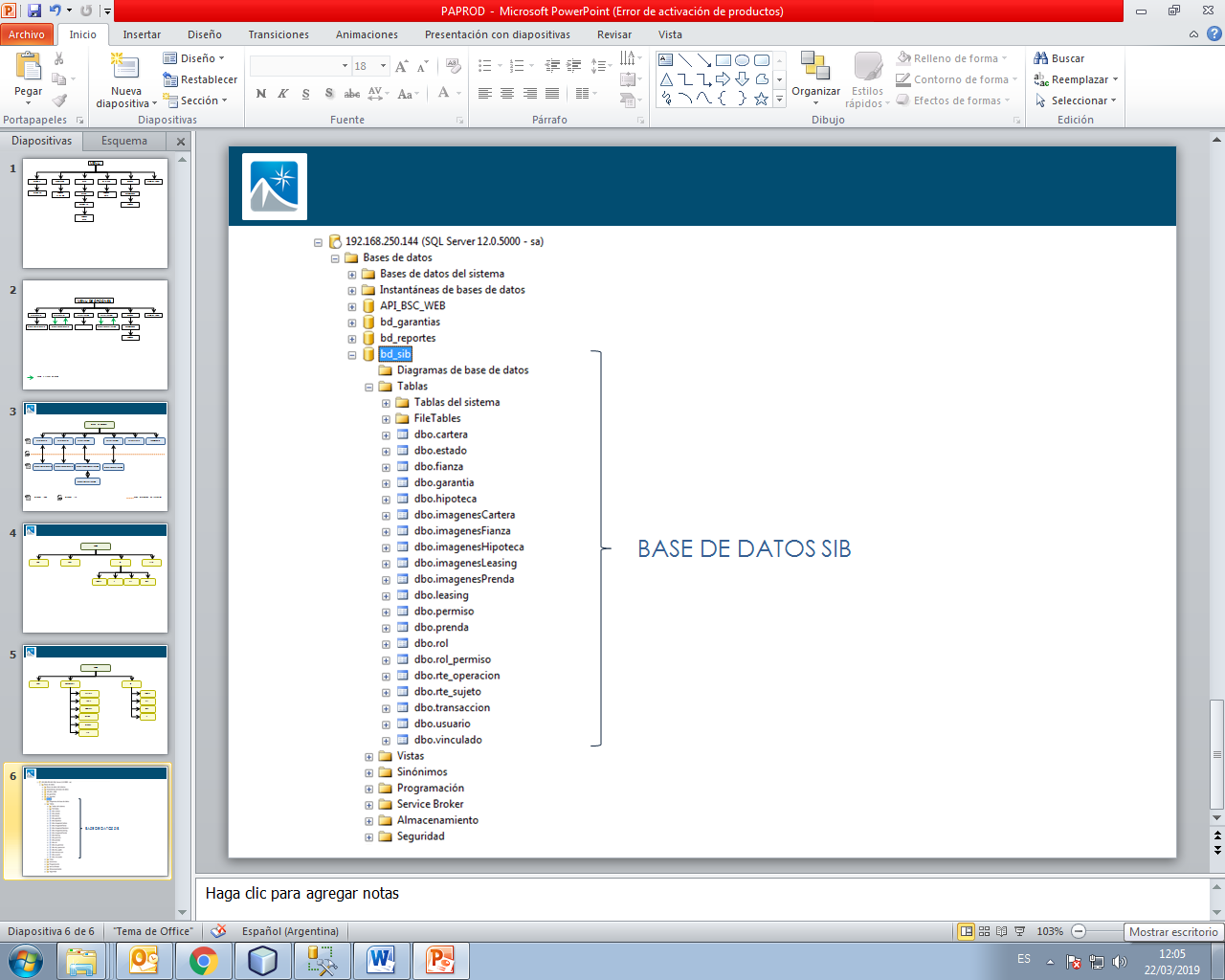


Por esta característica de comunicación es que el sistema requiere tener habilitadas las opciones JavaScript en el navegador.

# Base de datos

La base de datos de SIB se creó haciendo uso de Microsoft SQL SERVER Management Studio. La misma se denomina “SB\_SIB” y contiene las tablas correspondientes a los tres módulos iniciales del sistema.

Base de datos en desarrollo:



**Base de datos en 192.168.250.144**

## Modulo de Usuarios

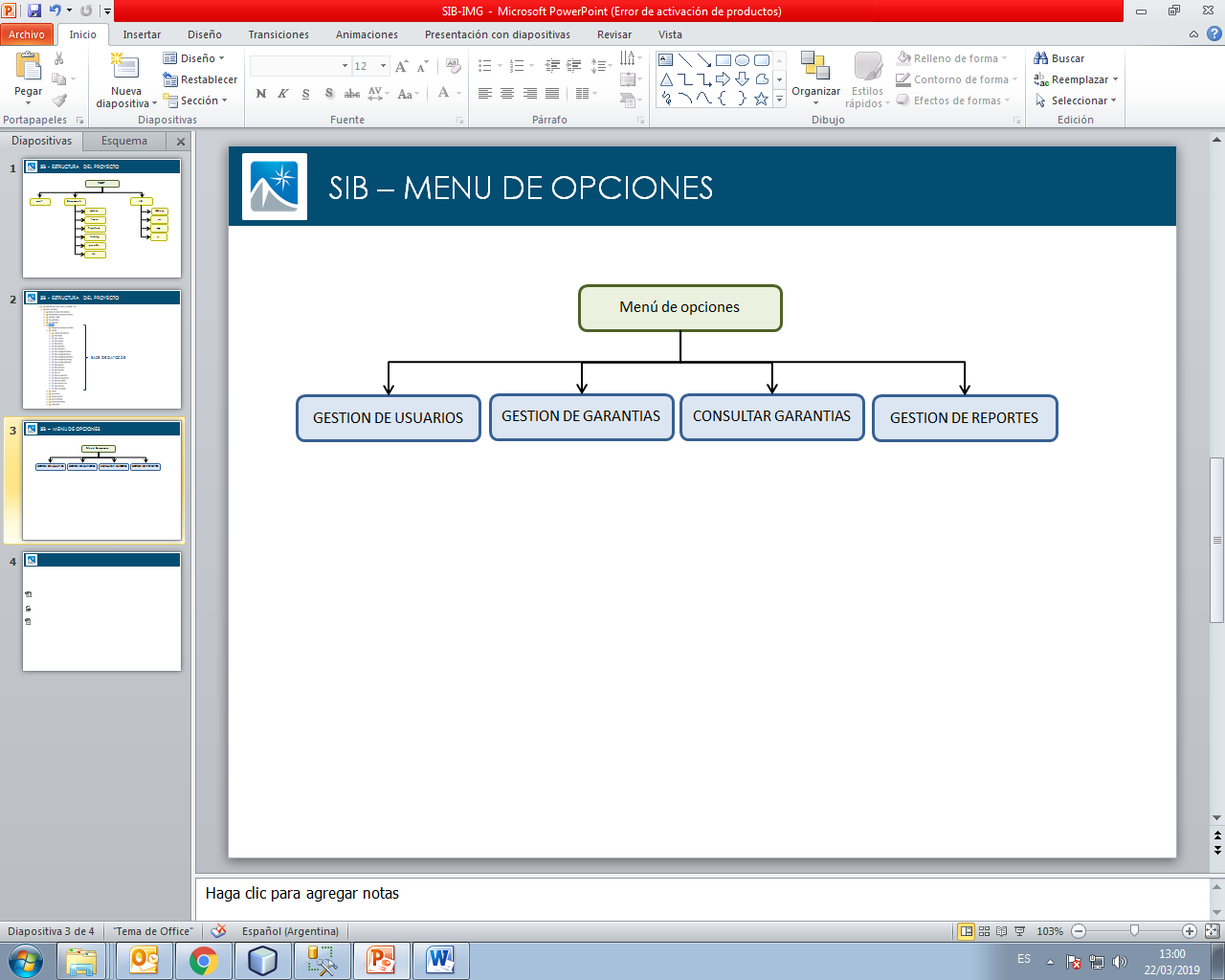
Detalle de las tablas que corresponden al módulo de Usuarios:

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| usuario | Guarda la información del usuario. Tiene una relación 1:N con rol donde un usuario cumple un rol y un rol puede ser asignado a N usuarios. Se utiliza para brindar acceso y dividir las tareas que puede realizar. |
| Rol | El rol define las tareas que puede realizar un usuario. Se relación con usuario y con permisos. La relación con permisos es N:N donde un rol puede tener múltiples permisos y un permiso puede pertenecer a múltiples roles. |
| permiso | Guarda los permisos que puede tener un rol. Representa las tareas que tiene permitido realizar el rol que se asocie a esta. |
| rol\_permiso | Representa la relación muchos a muchos. |

Inicialmente SIB requiere de los siguientes datos:

* PERMISOS:
  + GESTIONAR USUARIOS.
  + GESTIONAR GARANTIAS.
  + CONSULTAR GARANTIAS.
  + GESTIONAR REPORTES.

Estos permisos son los que permiten al sistema desplegar el menú de opciones.



## Modulo de Garantias

Detalle de las tablas que corresponden al módulo de Garantías:

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| garantía | Guarda los datos generales de la garantía. |
| cartera | Guarda los datos específicos de una cartera. La cartera está asociada a una garantía. Es una relación 1:1 con garantía. |
| estado | Guarda los estados que puede poseer una garantía. Sirve para parametrizar los estados. Es una relación 1:N con garantía. |
| fianza | Guarda los datos específicos de una fianza. La fianza está asociada a una garantía. Es una relación 1:1 con garantía. |
| hipoteca | Guarda los datos específicos de una hipoteca. La hipoteca está asociada a una garantía. Es una relación 1:1 con garantía. |
| leasing | Guarda los datos específicos de un leasing. El leasing está asociado a una garantía. Es una relación 1:1 con garantía. |
| prenda | Guarda los datos específicos de una prenda. La prenda está asociada a una garantía. Es una relación 1:1 con garantía. |
| imagenesCartera | Guarda la ruta de todos los documentos asociados a una cartera. Es una relación 1:N con cartera. |
| imagenesFianza | Guarda la ruta de todos los documentos asociados a una fianza. Es una relación 1:N con fianza. |
| imagenesHipoteca | Guarda la ruta de todos los documentos asociados a una hipoteca. Es una relación 1:N con hipoteca. |
| imagenesLeasing | Guarda la ruta de todos los documentos asociados a un leasing. Es una relación 1:N con leasing. |
| imagenesPrenda | Guarda la ruta de todos los documentos asociados a una prenda. Es una relación 1:N con prenda. |

## Modulo de reportes

Detalle de las tablas que corresponden al módulo de reportes:

|  |  |
| --- | --- |
| **Nombre** | **Descripción** |
| rte\_operacion | Guarda la información general de un Reporte de Transacciones en Efectivo. Se relación con rte\_sujeto dado que una operación puede tener asociado múltiples sujetos. |
| rte\_sujeto | Guarda la información del sujeto vinculado a un Reporte de Transacción en Efectivo. |
| transaccion | Guarda la información general de un Reporte de Transacción Internacional. Se relaciona con la tabla vinculado. |
| vinculado | Guarda la información del sujeto vinculado al Reporte de Transacción Internacional. |

# Archivos de configuración

El Sistema Integral Bancario cuenta con un archivo de configuración para agregar y modificar la información necesaria para la conexión con la base de datos. Dicho archivo se encuentra dentro de la carpeta “conf” del proyecto.

Este archivo es un XML que contiene la información de la conexión.

* En la etiqueta “**serverName**” se coloca la IP donde se encuentra la BD.
* En la etiqueta “**dataBase**” se coloca el nombre de la base de datos a usar.
* En la etiqueta denominada “**uid**” se indica el usuario.
* En la etiqueta “**pwd**” se indica la contraseña del usuario.

