



Instituto Politécnico Nacional

Centro de Educación Continua Unidad Morelia

Departamento de Servicios Técnicos y Administrativos

Manual Técnico

Sistema de Inventario de Bienes

SIB v1.0

Autor: Emotición

Contenido

1 Introducción	4
2 Objetivos del sistema	4
2.1 General	4
2.2 Específicos.....	4
3 Instalación	5
4 Metodologías utilizadas	6
5 Herramientas y programas aplicados.....	8
6 Especificación de Requerimientos	9
6.1 Introducción	10
6.2 Glosario de Términos.....	10
6.3 Definición de Requerimientos	12
6.4 Definición de Requerimientos Funcionales	12
6.5 Definición de Requerimientos No - Funcionales	13
6.6 Especificación Detallada de Requerimientos	13
6.6.1 Login	13
6.6.2 Altas.....	14
6.6.3 Consultas	15
6.6.4 Modificaciones	17
6.6.5 Bajas.....	18
6.7 Evolución del Sistema.....	18
7 Diseño de la aplicación	19
8 Diccionario de variables	33

9 EER Diagram	38
10 Algoritmos desarrollados	39
10.1 Correr aplicación	39
10.2 Crear cuenta de usuario	39
10.3 Iniciar sesión	41
10.4 Registrar nuevo bien	41
10.5 Búsqueda avanzada	43
10.6 Actualización de datos	53
10.7 Bajas	54
10.8 Recuperación de contraseña	55
10.9 Modificación de nombre de usuario y/o contraseña	58

1 Introducción

En el presente manual se describen las actividades que se realizaron para el desarrollo de este proyecto, detallando cada uno de sus componentes y la forma que interactúan entre sí para cumplir con una función específica relacionada a su razón de ser que es aportar una mejora al Centro de Educación Continua Unidad Morelia (CECUM), del Instituto Politécnico Nacional, en cuanto a la administración de los bienes muebles y equipo con los que cuenta.

2 Objetivos del sistema

2.1 General

Aportar una mejora al Centro de Educación Continua Unidad Morelia (CECUM), del Instituto Politécnico Nacional, en cuanto a la administración de los bienes muebles y equipo con los que cuenta.

2.2 Específicos

- Actualización pronta y oportuna de los datos.
- Control de bajas.
- Generación de inventarios generales y específicos.
- Acceso efectivo y más ágil al sistema.

3 Instalación

Para el correcto funcionamiento del sistema, deberá contar con los requerimientos mínimos o deseables tanto de software, como de hardware descritos en el manual de instalación.

A continuación se enlistan los pasos a seguir para una correcta instalación de cada uno de los programas necesarios para el funcionamiento del SIB.

1. Instalar Java JRE.
2. .NET Framework 4.0 (en caso de utilizar Windows XP)
3. Visual Studio Tools for Office Runtime 2010 (en caso de utilizar Windows XP)
4. MySQL Server 5.6 (la contraseña de la cuenta root debe ser “t1nocota+”, de lo contrario no funcionará el sistema).
5. MySQL Administrator.
Abra el programa MySQL Administrator y siga las indicaciones del manual de instalación para cargar los datos de los bienes.
6. Correr SetupSIB.exe

4 Metodologías utilizadas

Dada la naturaleza del proyecto y el tiempo con el que se contaba, se optó por seguir prácticas ágiles que permitieran la entrega de un software totalmente funcional en la fecha límite marcada, sin embargo, no se aplicó una sola metodología en todo el proceso, sino que se utilizaron diversas herramientas y artefactos que por las ventajas de cada uno, permitieron un correcto desarrollo en cada una de las fases del ciclo de vida del software, lo cual permitió la entrega de este producto dentro del tiempo establecido.

- **Tiempo para desarrollar el software:** Cuatro horas diarias durante 7 meses en días hábiles.
- **Participantes autores:** 1.

De acuerdo a las metodologías de la Familia Crystal, el desarrollo de este proyecto servirá para llevar un mejor control estrictamente interno de los bienes, sin embargo, se almacenarán los precios de los mismos. En base a esto se determina que la criticidad se ubica en el nivel D (Discretionary), ya que aunque se manejan los precios, el sistema llevará un control puramente interno, por lo que no afectará al respaldo de los datos con el que cuenta la Secretaría de Administración del Instituto Politécnico Nacional; la complejidad del sistema se ubica en 4, debido a la comunicación entre los distintos componentes al cumplir eficientemente con cada una de las funciones.

A continuación se describen las distintas herramientas y artefactos que se utilizaron con su debida justificación y metodología a la que pertenecen.

Herramienta	Metodología	justificación
Documento de Requerimientos	Tradicional	Esta herramienta garantiza que a futuro no se le permita al cliente dar nuevas especificaciones para el sistema, lo cual

		<p>resulta muy útil en caso de que desee que se implementen cambios que conlleven a una reestructuración de todo o de gran parte del diseño y/o del código de la aplicación o de la base de datos en un tiempo cercano a la fecha límite de entrega.</p> <p>Como se mencionó anteriormente, el principal uso de esta herramienta fue garantizar que no se le permitiera al cliente la petición de cambios grandes en los requerimientos en un tiempo cercano a la fecha límite, sin embargo, sí se le permitió especificar nuevas funciones dentro de un tiempo considerable, mismas que él mismo tuvo la oportunidad de ver cómo evolucionaban.</p>
CRC Cards	Extreme Programming (ágil)	Dado que se buscaba una mejora continua en el funcionamiento de la aplicación, constantemente se hacían cambios en el diseño, por esta razón las Tarjetas CRC permitieron incorporar nuevas clases, así como eliminar las obsoletas.
EER Diagram	Tradicional	Para la programación de las tablas se hizo indispensable una visualización tan completa de los atributos, como la que ofrecen estos diagramas.

5 Herramientas y programas aplicados

- **Microsoft Word 2013:** Elaboración del documento de requerimientos, manual de instalación, de usuario y técnico.
- **Microsoft Excel 2013:** Elaboración del diccionario de variables, conversión de los datos de bienes a formato CSV.
- **MySQL Workbench:** Diseño del Diagrama EER (Enhanced Entity Relationship).
- **MySQL 5.6 Command Line Client:** Programación de la base de datos.
- **MySQL Administrator 1.1.9:** Creación y restauración (o recuperación) de backups.
- **SQLyog Community:** Exportación de datos a partir de archivos CSV.
- **Java JDK 7:** Desarrollo de la aplicación (para un correcto funcionamiento en Windows XP).
- **NetBeans IDE 8.0.2:** Programación de la aplicación y compilación automática del código.
- **StarUML:** Ingeniería inversa aplicada al código Java para la verificación de un correcto diseño de las clases.
- **Apache Poi 3.12:** Librería utilizada para la conexión de Java con Microsoft Excel.
- **MySQL JDBC Driver:** Conexión de la base de datos con la aplicación.
- **PhotoScape:** Edición, conversión de imágenes y captura de pantallas.
- **Online-convert.com:** Conversión del icono de la aplicación a formato .ico
- **Caesium Image Compressor:** Compresión de imágenes.
- **Launch4j 3.8:** Creación de formato ejecutable para Windows® de la aplicación.
- **Inno Setup Compiler 5.5.2:** Creación del archivo instalador.

6 Especificación de Requerimientos

El día 09/03/15 fue firmado y aceptado el presente Documento de Requerimientos

Instituto Politécnico Nacional
Centro de Educación Continua Unidad Morelia
Departamento de Servicios Técnicos y Administrativos



6.1 Introducción

En este documento se mencionarán aspectos importantes sobre lo que se espera de este nuevo sistema.

La finalidad de este proyecto es llevar un mejor control estrictamente local de los bienes, realizando consultas más exactas de acuerdo a lo que el usuario desea visualizar.

El sistema podrá almacenar los datos y generar inventarios en formato de Microsoft Excel con los datos solicitados y sólo existirá un usuario con los permisos de almacenar, consultar, modificar o dar de baja algún bien.

6.2 Glosario de Términos

- **Aplicación:** Programa desarrollado para la automatización de un proceso específico.
- **Sistema:** Conjunto de componentes de hardware y software estructuralmente conectados entre sí para automatización de un proceso.
- **Software:** Conjunto de programas que a través del cálculo de diversas instrucciones lógico-matemáticas, hacen funcionar una computadora.
- **Usuario:** Persona o programa que utiliza las funciones del sistema.
- **Administrador:** En informática, un administrador es quien tiene derecho de utilizar absolutamente todas las funciones del sistema, excepto modificarlo internamente (acceder al código fuente, que es donde se especifican los detalles técnicos de alto nivel del software). Las funciones que puede manipular son: dar de alta, consultar y dar de baja.

- **Interfaz:** Se dice de un espacio de la computadora, (que aparece en la pantalla o monitor), en el que el usuario interactúa con el software. Ejemplo: en la interfaz de Microsoft Word se muestra una opción para guardar el documento, escribir palabras y cambiar su aspecto, agregar imágenes, etc.; en la interfaz de Gmail se muestra la opción de escribir un nuevo mensaje, ver los mensajes recibidos, etc.
- **Gráfico:** De acuerdo al léxico de la informática, se dice de las imágenes, vídeos, iconos, cursores, o cualquier cosa que ilustre alguna función de la aplicación y que en consecuencia facilite el uso de ésta y haga más agradable su uso ante el usuario.
- **GUI:** Interfaz Gráfica de Usuario (Graphical User Interface, por sus siglas en inglés), como su nombre lo dice, se trata de un espacio de la computadora que permite interactuar con el software de una manera fácil y atractiva, usando dibujos, iconos, vídeos, etc.
- **Permiso:** En informática al hablar de permisos, se trata de los derechos que tienen los distintos tipos de usuarios sobre el sistema. Ejemplo: en un sistema de administración escolar, un alumno no tiene derecho de modificar sus calificaciones, ya que eso sólo le corresponde a su profesor y al administrador, por lo cual, el alumno no tendrá derecho de modificar sus calificaciones, pero el maestro y el administrador sí.
- **Base de datos:** Se trata de un espacio en el que se almacenan los datos de ciertos usuarios. // Es un conjunto de datos estructurados de tal forma que sea posible almacenarlos de forma segura en grandes o enormes cantidades, así como también sean fáciles de manipular.
- **Login:** Dícese, del inicio de sesión al sistema.

- **ID:** Identificador. Dícese de un número de clave único, es decir, número que sirve para identificar a cada uno de los registros.

6.3 Definición de Requerimientos

- Se requiere un software con el que se puedan llevar a cabo altas, modificaciones, consultas y bajas de los bienes.
- Sólo un usuario tendrá acceso y la aplicación se encontrará instalada en la computadora correspondiente.
- Las consultas se deben realizar con mayor agilidad y visualizándose en Microsoft Excel.

6.4 Definición de Requerimientos Funcionales

- Llevar un control de los bienes con una base de datos.
- Sólo existirá un usuario con los permisos de altas, modificaciones, consultas y bajas. Éste contará con un nombre de usuario y contraseña para tener acceso.
- La aplicación debe contar con las opciones necesarias para que en primera instancia el usuario pueda visualizar los datos que necesite de forma concreta, además que también contará con la opción de realizar una búsqueda más detallada.
- La aplicación debe generar un nuevo archivo de Excel con los datos solicitados.

6.5 Definición de Requerimientos No - Funcionales

- En cuestión de la seguridad del sistema, fue especificado que no es necesario implementar métodos que protejan a los datos de cualquier ataque, pero lo que sí se le protegerá a la cuenta de usuario, ya que ninguna otra persona debe tener acceso al sistema.
- El software se encontrará únicamente en la computadora del administrador.
- La GUI debe ser atractiva y fácil de usar.

6.6 Especificación Detallada de Requerimientos

6.6.1 Login

Sólo existirá un usuario con acceso al sistema, éste tendrá derecho a cambiar su nombre de usuario y contraseña, o de recuperarla.

- En caso de que los recuerde pero sólo desee cambiarlos primero deberá iniciar sesión, al hacerlo contará con la opción de modificar el usuario y/o contraseña, al seleccionarla se desplegará una nueva ventana que le solicitará su contraseña actual y dos campos más, una lista desplegable para seleccionar una pregunta secreta y otro para introducir la respuesta, además del campo para la nueva contraseña. Si los todos los datos son correctos la contraseña habrá modificado exitosamente y se desplegará un mensaje que lo confirme, de lo contrario, si la contraseña anterior es incorrecta, la pregunta o respuesta, o la nueva contraseña introducida en el campo de confirmación no es idéntica a la del primer campo, la aplicación desplegará el mensaje de error correspondiente.
- Si el usuario extravió u olvido su contraseña, la ventana de inicio de sesión contará con una opción que le permitirá recuperarla, al seleccionarla se

desplegará una nueva ventana en que la podrá especificar la pregunta y respuesta de la que se habló anteriormente, además de 3 preguntas más, todas deberán ser seleccionadas correctamente para que pueda recuperar su cuenta, de lo contrario no se le permitirá. En caso de que las 4 preguntas y respuestas sean correctas podrá introducir su nueva contraseña y confirmarla, si coinciden se desplegará un mensaje que confirme que la contraseña ha sido cambiada exitosamente, de lo contrario se le informará que no coinciden y que las vuelva a introducir.

Es importante aclarar que por seguridad, el usuario no tendrá derecho de cambiar las preguntas y respuestas.

6.6.2 Altas

Para dar de alta algún bien, la ventana contendrá los siguientes campos de texto:

- No. de inventario.
- Descripción. (Computadora, silla, escritorio, etc.)
- Marca.
- Modelo.
- Número de serie.
- Edificio.
- Nivel.
- Local.
- Fecha de asignación.

- Persona asignada.
- Departamento.
- Valor. (\$)
- Detalles.

Además de un botón para guardar y otro para cancelar el registro.

Todos los campos serán obligatorios, excepto el número de inventario. En caso de que todos se hayan llenado correctamente el registro se almacenará en la base de datos, de lo contrario se desplegará un mensaje que indique los datos a corregir.

6.6.3 Consultas

Con la finalidad de hacer más ágil el proceso de consultas, se contará con un buscador en el que se introduzca la descripción del bien que se desea información, en el mismo se mostrarán algunos ejemplos como computadora, escritorio, etc., además de dos listas de opción, una para seleccionar la persona de asignación y otra para el departamento donde se encuentra (no será obligatorio elegir una opción de cualquiera de las dos listas). A un lado del buscador se encontrará una opción para realizar una búsqueda detallada, al seleccionarla se desplegarán los mismos campos que para dar de alta, sólo que esta vez servirán para realizar una búsqueda más exacta, como generar un inventario de todas las computadoras HP que hay en el laboratorio 1.

Ya sea realizando una búsqueda concreta o detallada, al seleccionar la opción de generar inventario, se creará un nuevo archivo de Microsoft Excel con los datos que el usuario solicitó. Todos los archivos generados se guardarán automáticamente en la misma carpeta.

Generalmente hablando, los datos que se mostrarán en el archivo serán los siguientes:

- ID del Bien
- No. de inventario.
- Descripción. (Computadora, silla, escritorio, etc.)
- Marca.
- Modelo.
- Número de serie.
- Edificio.
- Nivel.
- Local.
- Persona asignada.
- Fecha de asignación.
- Antigüedad (cuántos meses y años tiene desde que se le asignó a la persona)
- Departamento.
- Valor. (\$)
- Detalles.

También se podrán consultar los registros que fueron dados de baja y de la misma manera se generará un archivo de Microsoft Excel que contenga los siguientes datos:

- ID del Bien
- Descripción
- Marca
- Modelo
- Fecha de baja
- Detalles

6.6.4 Modificaciones

Para hacer modificaciones se contará con el mismo buscador y la lista desplegable donde aparezcan las personas a las que se les han asignado bienes, si existen registros que coincidan con la búsqueda, éstos se enlistarán, si el usuario especifica la persona designada, se mostrará el ID del bien, la descripción, la marca y modelo, pero si no lo especifica, en la lista aparecerá la persona designada junto con el ID, la descripción, marca y modelo.

Una vez que se haya desplegado la lista, el usuario podrá seleccionar al bien deseado, al hacer esto se ubicará cada dato en su respectivo campo. Una vez que se hayan especificado los cambios que se quieren realizar, el usuario podrá presionar el botón de guardar cambios, si los datos son correctos la aplicación lanzará un mensaje de que los cambios se han realizado exitosamente y regresará al buscador esperando una nueva búsqueda, de lo contrario indicará qué es lo que se tiene que corregir.

6.6.5 Bajas

Esta sección contendrá las mismas opciones de la opción *modificaciones*, y funcionará de la misma manera en cuanto a búsquedas, ya sea concretas o detalladas, sólo que esta vez al seleccionar el bien deseado la aplicación solicitará confirmación para continuar con la baja, si el usuario confirma, el "registro será borrado del sistema", o mejor dicho cambiará a estado de baja.

6.7 Evolución del Sistema

En un futuro se cambiarán y mejorarán los métodos de seguridad (acceso al sistema) para proteger a la cuenta de usuario, se rediseñará la base de datos y la aplicación con la finalidad de que las altas, bajas, consultas y modificaciones se realicen más ágil y eficazmente, además de que también la GUI mejorará su aspecto.

7 Diseño de la aplicación

- **Paquete ConexionDB:** Clases necesarias para almacenar nuevos datos de bienes, actualizarlos o consultarlos.

Las cuales son:

- Aconexion.
 - Selecciones
 - Updates.
-
- **Paquete Interacciones:** Clases necesarias para la obtención y despliegue de los datos de bienes, formato de la tabla donde se visualizan, así como la generación de inventarios en formato de Microsoft Excel.

Las cuales son:

- Bien.
 - BselectBA.
 - Fecha.
 - FormatoTablaBien.
 - GeneradorExcel.
-
- **Paquete Login:** Donde se encuentra la clase principal y pantalla de inicio de sesión, además de dos sub-paquetes que permiten el correcto funcionamiento de ésta.
-
- **Sub-paquete Login.RDU:** *Recuperación Datos de Usuario.* Contiene los siguientes diseños Swing de la pantallas:
 - CambNUC: Cambiar nombre de usuario y/o contraseña.
 - NvoUsuario: Registrar nueva cuenta.

- RecupContra: Recuperar contraseña.

Además de la clase Usuario para la obtención de los datos del mismo, así como de cada una de las preguntas y respuestas necesarias para la recuperación o modificación de la contraseña.

- **Sub-paquete Login.RDU.DB:** Contiene las siguientes clases:
 - Aconexion.
 - PR (Preguntas y Respuestas)
 - Selecciones
 - Updates
- **Paquete MovimientosSIB:** Contiene el diseño Swing de la pantalla de administración de bienes.

Librerías utilizadas:

- Apache Poi 3.12 (conexión Java – Microsoft Excel)
 - Poi-3.12-20150511.jar
 - Poi-excelant-3.12-20150511.jar
 - Poi-ooxml-3.12-20150511.jar
 - Poi-ooxml-schemas-3.12-20150511.jar
 - Poi-scratchpad-3.12-20150511.jar
 - Xmlbeans-2.6.0.jar
- MySQL JDBC Driver – mysql-connector-java-5.1.23-bin.jar
- JDK 1.8

A continuación se dará una descripción más detallada de las clases, apoyándonos en el uso del formato de CRC Cards.

ConexionDB.Aconexion	
Responsabilidades	Colaboradores
Conexión con la base de datos	

Esta clase contiene la siguiente función:

- **Conexión con la base de datos**
 - **Nombre del método:** conectar();
 - **Tipo:** void.
 - **Permisos:** public.

ConexionDB.Inserciones	
Responsabilidades	Colaboradores
Registrar nuevos bienes	Aconexion

Esta clase contiene la siguiente función:

- **Registrar nuevos bienes.**
 - **Nombre del método:** RegistrarBien();
 - **Tipo:** boolean.
 - **Permisos:** public.

Trabaja colaborativamente con la clase Aconexion.

ConexionDB.Selecciones	
Responsabilidades	Colaboradores
Verificar No de serie	Aconexion
Verificar clave CABM	Interacciones.Bien
Verificar nombre de usuario	
Desplegar bienes	

Esta clase contiene las siguientes funciones:

- Verificar No. de serie:** Este método verificará que el dato no se haya registrado previamente, en caso de almacenar un nuevo registro, para evitar ambigüedades.
 - **Nombre del método:** VNnoserie();
 - **Tipo:** boolean.
 - **Permisos:** public.
- Verificar Clave CABM:** La finalidad de este método es la misma del anterior, solo que esta vez se verificará que la clave CABM introducida no se haya registrado previamente.
 - **Nombre del método:** VNclaveCABM();
 - **Tipo:** boolean.
 - **Permisos:** public.
- Verificar nombre de usuario:** Sólo debe existir un usuario registrado en el sistema, por lo que éste método se encargará de comprobar que así sea, además de que en caso de que no lo exista, informará de la situación para que se permita crear una nueva cuenta.
 - **Nombre del método:** VnomUsuario();
 - **Tipo:** ArrayList.

- **Permisos:** public.
- **Desplegar bienes:** Se obtendrán los datos de los bienes con estado de alta.
 - **Nombre del método:** getBien();
 - **Tipo:** ArrayList.
 - **Permisos:** public.

Esta clase trabaja colaborativamente con:

- **Aconexion:** Consulta de los datos deseados.
- **Interacciones.Bien:** Obtención de los datos de cada columna del view correspondiente de la DB.

ConexionDB.Updates	
Responsabilidades	Colaboradores
Actualizar registros	Aconexion
Dar de baja	

Esta clase contiene las siguientes funciones:

- **Actualizar registros:** Actualizar o modificar datos de bienes.
 - **Nombre del método:** Actualizar();
 - **Tipo:** boolean.
 - **Permisos:** public.
- **Dar de baja:** Se cambia el estado de alta a baja en lugar de borrarse el registro.

- **Nombre del método:** DarDeBaja();
- **Tipo:** boolean.
- **Permisos:** public.

Esta clase trabaja colaborativamente con Aconexion para la actualización de los datos deseados.

Interacciones.Bien	
Responsabilidades	Colaboradores
Obtener datos	
Desplegar datos	

Esta clase contiene los siguientes métodos:

- **Obtener datos:** Obtener los datos de cada uno de los registros.
 - **Nombre:** “get” + nombreDeLaVariable
- **Desplegar datos:** Desplegar los datos de cada uno de los registros.
 - **Nombre:** “set” + nombreDeLaVariable

Interacciones.BselectBA	
Responsabilidades	Colaboradores
Búsqueda avanzada	

Esta clase contiene la siguiente función:

- **Búsqueda avanzada:** Este método al ser invocado, obtiene los valores que se hayan introducido en cada uno de los campos y determina cuáles fueron llenados, para que con base en esto la aplicación le envíe el comando SQL correcto al gestor y así puedan ser desplegados únicamente los datos solicitados.
 - **Nombre del método:** busqASelcSQL();
 - **Tipo:** String.
 - **Permisos:** public.

Interacciones.Fecha	
Responsabilidades	Colaboradores
Regresar formato SQL de fecha	
Obtener fecha actual	
Obtener fecha y hora actual	
Obtener año actual	

Esta clase contiene las siguientes funciones:

- **Regresar formato SQL de fecha:** MySQL maneja un formato estricto para las fechas, el cual es: 'AAAA-MM-DD', y la aplicación requiere cada dato por

separado, por lo que este método ayuda a regresarlos a su formato correspondiente para ser procesados por el gestor.

- **Nombre del método:** Fecha();
- **Tipo:** String.
- **Permisos:** public.

- **Obtener fecha actual.**
 - **Nombre del método:** mostrarFecha();
 - **Tipo:** String.
 - **Permisos:** public.

- **Obtener fecha y hora actual.**
 - **Nombre del método:** mostrarFechayHora();
 - **Tipo:** String.
 - **Permisos:** public.

- **Obtener año actual.**
 - **Nombre del método:** mostrarFechayHora();
 - **Tipo:** String.
 - **Permisos:** public.

Interacciones.FormatoTablaBien	
Responsabilidades	Colaboradores
Dar formato a la tabla	AbstractTableModel

Esta clase contiene la siguiente función:

- **Dar formato a la tabla:** Indicar el número de columnas, el nombre de cada una, así como insertar cada registro en su fila correspondiente.

Interacciones.GeneradorExcel	
Responsabilidades	Colaboradores
Generar inventario	
Generar inventario de bajas	

Esta clase contiene las siguientes funciones:

- **Generar inventario:** Crea un nuevo archivo de Microsoft Excel e inserta en él los datos de la consulta realizada.
 - **Nombre del método:** GenerarInventarioE();
 - **Tipo:** boolean.
 - **Permisos:** public.
- **Generar inventario de bajas:** Bienes que se han dado de baja.
 - **Nombre del método:** GenerarInventarioEBajas();
 - **Tipo:** boolean.
 - **Permisos:** public.

Login.RDU.Usuario	
Responsabilidades	Colaboradores
Obtener datos de usuario	
Desplegar datos de usuario	
Obtener preguntas y respuestas	
Desplegar preguntas y respuestas	

Esta clase contiene las siguientes funciones:

- **Obtener datos de usuario:** Obtener los datos de cada uno de los registros.
 - **Nombre:** “get” + nombreDeLaVariable + “();”
- **Desplegar datos de usuario:** Desplegar los datos de cada uno de los registros.
 - **Nombre:** “set” + nombreDeLaVariable + “();”
- **Obtener preguntas y respuestas:** Obtener los datos de cada uno de los registros.
 - **Nombre:** “get” + nombreDeLaVariable + “();”
- **Desplegar preguntas y respuestas:** Desplegar los datos de cada uno de los registros.
 - **Nombre:** “set” + nombreDeLaVariable + “();”

Login.RDU.DB.Aconexion	
Responsabilidades	Colaboradores
Conexión con la base de datos	
Registrar usuario	

Esta clase contiene las siguientes funciones:

- **Conexión con la base de datos.**
 - **Nombre del método:** conectar();
 - **Tipo:** void.
 - **Permisos:** public.
- **Registrar usuario:** Nueva cuenta de usuario.
 - **Nombre del método:** RegistrarUsuario();
 - **Tipo:** boolean.
 - **Permisos:** public.

Login.RDU.DB.PR	
Responsabilidades	Colaboradores
Obtener pregunta y respuesta	Aconexion
Desplegar pregunta y respuesta	

Esta clase contiene las siguientes funciones:

- **Obtener pregunta y respuesta.**
 - **Nombre:** "get" + nombreDeLaVariable

- **Desplegar pregunta y respuesta.**
 - Nombre: "set" + nombreDeLaVariable

Login.RDU.DB.Selecciones	
Responsabilidades	Colaboradores
Iniciar sesión	Aconexion
Desplegar preguntas	
Verificar nombre de usuario	
Verificar contraseña	
Verificar respuestas	

Esta clase contiene las siguientes funciones:

- **Iniciar sesión.**
 - **Nombre del método:** Login();
 - **Tipo:** boolean.
 - **Permisos:** public.
- **Desplegar preguntas:** Desplegar preguntas secretas en la pantalla de recuperación de contraseña.
 - **Nombre del método:** Preguntas();
 - **Tipo:** ArrayList.
 - **Permisos:** public.
- **Verificar nombre de usuario:** Al iniciar sesión, este método verifica que el nombre de usuario sea correcto sin necesidad de presionar el botón *Entrar*, además de otros usos de seguridad.

- **Nombre del método:** VnomUsuario();
 - **Tipo:** ArrayList.
 - **Permisos:** public.
- **Verificar contraseña:** Si es correcta al ser introducida en la pantalla para modificar el nombre de usuario y/o contraseña.
 - **Nombre del método:** Vcontraseña();
 - **Tipo:** boolean.
 - **Permisos:** public.
 - **Verificar respuestas:** Se verifica si el usuario respondió correctamente cada una de las respuestas secretas, el formato utilizado para cada uno de estos métodos es:
 - **Nombre del método:** "VRespuesta" + NumDeRespuesta + "();"
 - **Tipo:** boolean.
 - **Permisos:** public.

Login.RDU.DB.Updates	
Responsabilidades	Colaboradores
Actualizar contraseña	
Actualizar nombre de usuario y/o contraseña	

Esta clase contiene las siguientes funciones:

- **Actualizar contraseña.**
 - **Nombre del método:** ActualizarC();

- **Tipo:** boolean.
- **Permisos:** public.

- Actualizar nombre de usuario y/o contraseña.
 - **Nombre del método:** ActualizarNomUC();
 - **Tipo:** boolean.
 - **Permisos:** public.

8 Diccionario de variables

El diseño de la base de datos del Sistema de Inventario de Bienes fue realizado en base al formato establecido de la norma ISO-10015.

Tabla	Variable	Tipo de Dato Java	Nombre del Campo	Tipo de Dato MySQL
MBien	idBien	Integer	id_ben	MEDIUMINT
MBien	descripcion	String	des_ben	VARCHAR(45)
MBien	noSerie	String	nse_ben	VARCHAR(45)
MBien	claveCABM	String	cbm_ben	VARCHAR(45)
MBien	idEstado	Integer	id_est	MEDIUMINT
MBien	idDatosExtrasBien	Integer	id_mrc	MEDIUMINT
MBien	idDetalles	Integer	id_det	MEDIUMINT
MBien	fechaBaja	String	fbj_ben	DATE
CDetalles	idDetalles	Integer	id_det	MEDIUMINT
CDetalles	valor	Double	val_det	DECIMAL
CDetalles	detalles	String	dls_det	TEXT
CMarca	idMarca	Integer	id_mrc	MEDIUMINT
CMarca	marca	String	nom_mrc	VARCHAR(45)
CMarca	modelo	String	mlo_mrc	VARCHAR(45)
CEstado	idEstado	Integer	id_est	MEDIUMINT
CEstado	estado	Char	nom_est	CHAR
CAsignacion	idAsignacion	Integer	id_asg	MEDIUMINT
CAsignacion	personaDesig	String	prs_asg	VARCHAR(45)
CAsignacion	fechaAsig	String	fas_arg	DATE
CAsignacion	idBien	Integer	id_ben	MEDIUMINT
CAsignacion	idUbicacion	Integer	id_ubi	MEDIUMINT
CUbicacion	idUbicacion	Integer	id_ubi	MEDIUMINT
CUbicacion	edificio	String	edi_ubi	VARCHAR(45)
CUbicacion	nivel	Integer	nvl_ubi	INT
CUbicacion	local	Integer	lcl_ubi	INT
CUbicacion	departamento	String	dep_ubi	VARCHAR(45)
MUsuario	idUsuario	Integer	id_usu	TINYINT
MUsuario	nombreUsuario	String	nom_usu	VARCHAR(30)
MUsuario	contraseñaUsuario	String	psw_usu	VARCHAR(45)
MUsuario	idPrivilegio	Integer	id_prv	MEDIUMINT
CPrivilegio	idPrivilegio	Integer	id_prv	MEDIUMINT

CPrivilegio	nombrePrivilegio	Char	nom_prv	CHAR
CPregunta	idPregunta	Integer	id_prg	SMALLINT
CPregunta	pregunta	String	pre_prg	VARCHAR(45)
CPregunta	idRespuesta	Integer	id_rta	SMALLINT
CPregunta	idUsuario	Integer	id_usu	TINYINT
CRespuesta	idRespuesta	Integer	id_rta	SMALLINT
CRespuesta	respuesta	Integer	res_rta	VARCHAR(45)
DBienes	idBien	Integer	id_ben	MEDIUMINT
DBienes	claveCABM	String	cbm_ben	VARCHAR(45)
DBienes	descripcion	String	des_ben	VARCHAR(45)
DBienes	marca	String	nom_mrc	VARCHAR(45)
DBienes	modelo	String	mlo_mrc	VARCHAR(45)
DBienes	noSerie	String	nse_ben	VARCHAR(45)
DBienes	edificio	String	edi_ubi	VARCHAR(45)
DBienes	nivel	Integer	nvl_ubi	INT
DBienes	local	Integer	lcl_ubi	INT
DBienes	departamento	String	dep_ubi	VARCHAR(45)
DBienes	personaDesig	String	prs_asg	VARCHAR(45)
DBienes	fechaAsig	String	fas_arg	DATE
DBienes	antigüedad	Integer	ant_cbi	VARCHAR(11)
DBienes	valor	Double	val_det	DECIMAL
DBienes	detalles	String	dls_det	VARCHAR(45)
DBajas	id_ben	Integer	id_ben	MEDIUMINT
DBajas	claveCABM	String	cbm_ben	VARCHAR(45)
DBajas	descripcion	String	des_ben	VARCHAR(45)
DBajas	marca	String	nom_mrc	VARCHAR(45)
DBajas	modelo	String	mlo_mrc	VARCHAR(45)
DBajas	noSerie	String	nse_ben	VARCHAR(45)
DBajas	edificio	String	edi_ubi	VARCHAR(45)
DBajas	nivel	Integer	nvl_ubi	INT
DBajas	local	Integer	lcl_ubi	INT
DBajas	departamento	String	dep_ubi	VARCHAR(45)
DBajas	personaDesig	String	prs_asg	VARCHAR(45)
DBajas	fechaBaja	String	fbj_ben	DATE
DBajas	antigüedad	Integer	ant_cob	VARCHAR(11)
DBajas	valor	Double	val_det	DECIMAL
DBajas	detalles	String	dls_det	VARCHAR(45)
DPregResp	pregunta	String	pre_prg	VARCHAR(45)
DPregResp	respuesta	String	res_rta	VARCHAR(45)

DPrgRpasw	pregunta	String	pre_prg	VARCHAR(45)
DPrgRpasw	respuesta	String	res_rta	VARCHAR(45)
DPrgRpasw	contraseñaUsuario	String	psw_usu	VARCHAR(45)

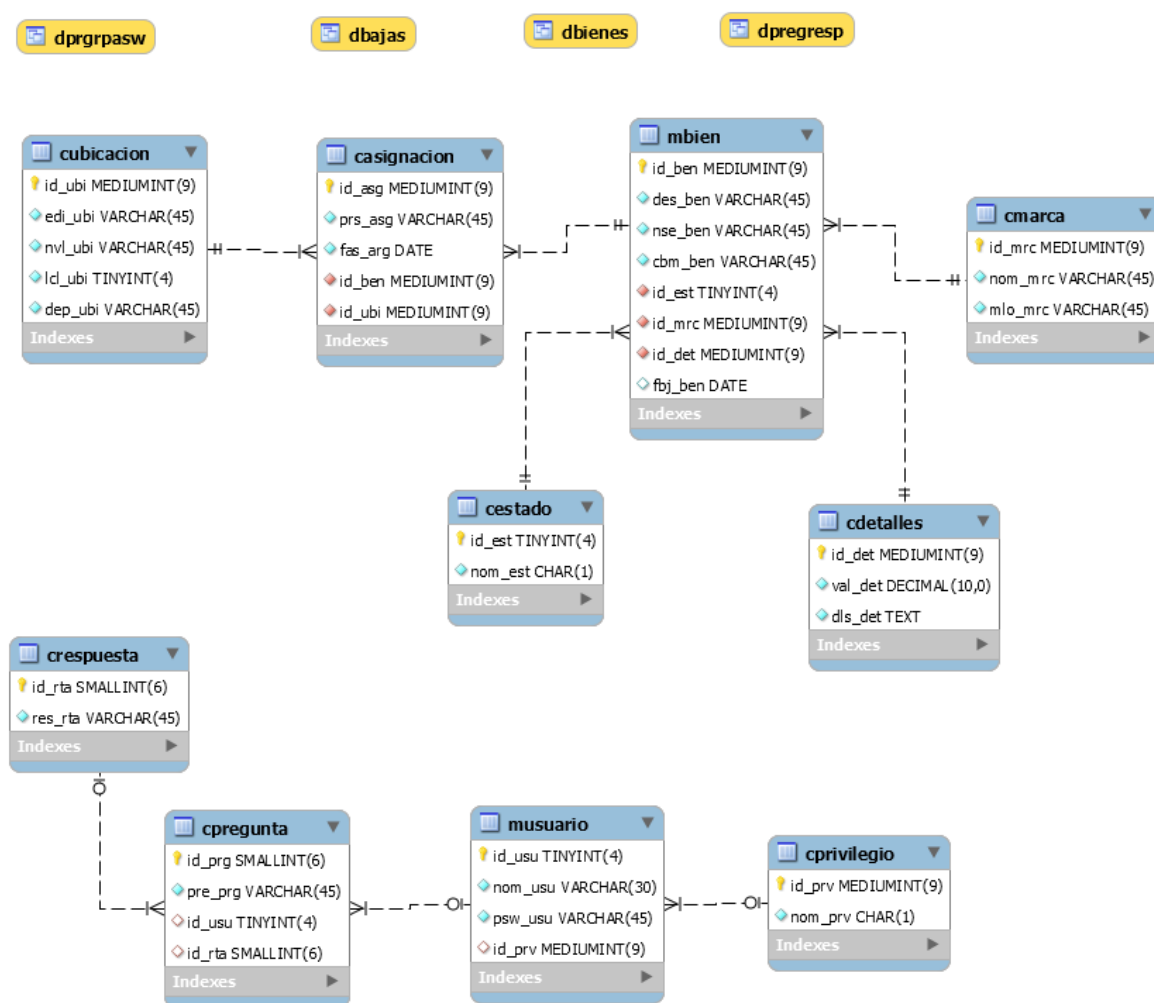
Tabla	Variable	Descripción
MBien	idBien	ID del bien
MBien	descripcion	Descripción del bien (si es una computadora, escritorio, etc.)
MBien	noSerie	No de serie del bien
MBien	claveCABM	Clave CABM del bien
MBien	idEstado	ID del estado, esto quiere decir si está dado de alta o de baja
MBien	idDatosExtrasBien	Este ID es de la tabla que contiene la marca y modelo del bien
MBien	idDetalles	Este ID es de la tabla que contiene el costo o valor del bien, además de otros detalles del mismo
MBien	fechaBaja	Este campo servirá para cuando se le dé de baja al bien, pero por default será nulo
CDetalles	idDetalles	ID de la tabla dDetalles
CDetalles	valor	Costo del bien
CDetalles	detalles	Detalles extras, como el color, tamaño, sistema operativo (en caso de que sea computadora), etc.
CMarca	idMarca	ID de la tabla CMarca
CMarca	marca	Nombre de la marca
CMarca	modelo	Modelo
CEstado	idEstado	ID de la tabla dEstado (si está dado de alta o baja)
CEstado	estado	Sólo habrá dos valores: "A" significa alta y "B" significa baja
CAsignacion	idAsignacion	ID Asignación
CAsignacion	personaDesig	Persona asignada, es decir, la persona responsable del bien
CAsignacion	fechaAsig	Fecha en que se le asignó
CAsignacion	idBien	ID del bien, esto para que podamos realizar consultas de todos los datos relacionamos las tablas a través de los ID's

CAsignacion	idUbicacion	ID de la ubicación.
CUbicacion	idUbicacion	ID de la tabla ubicación
CUbicacion	edificio	Edificio en que se encuentra (este dato se incluyó en caso de que sea requerido, pero en general será el mismo valor, el de CEC Morelia).
CUbicacion	nivel	Si el bien se encuentra ubicado en el piso 1, o en el 2
CUbicacion	local	Local, cuarto o lugar exacto donde se encuentra.
CUbicacion	departamento	Si pertenece al departamento de informática, recursos humanos, administración, etc.
MUsuario	idUsuario	ID del usuario
MUsuario	nombreUsuario	Nombre de usuario
MUsuario	contraseñaUsuario	Contraseña
MUsuario	idPrivilegio	ID del privilegio, esto significa si administrador o usuario común, aunque en este proyecto sólo existirá un usuario y éste será el propio administrador, sin embargo, se agrega este dato como medida de seguridad.
CPrivilegio	idPrivilegio	ID del privilegio.
CPrivilegio	nombrePrivilegio	Nombre del privilegio.
CPregunta	idPregunta	ID de la pregunta.
CPregunta	pregunta	En caso de olvido o extravío de la contraseña por parte del usuario, podrá cambiarla respondiendo 4 preguntas secretas, este campo sirve para almacenar las preguntas.
CPregunta	idRespuesta	ID Respuesta, este ID ayudará a relacionar a cada pregunta con su respuesta.
CPregunta	idUsuario	ID usuario, aunque sólo existe un usuario registrado, debemos relacionar a cada pregunta con el usuario por medio de este dato como medida de seguridad
CRespuesta	idRespuesta	ID de la respuesta
CRespuesta	respuesta	Respuesta a la pregunta secreta
DBienes	idBien	ID del bien, que servirá para identificarlo al realizar una consulta de los mismos
DBienes	claveCABM	Clave CABM del bien
DBienes	descripcion	Descripción
DBienes	marca	Nombre de la marca
DBienes	modelo	Modelo

DBienes	noSerie	No. de serie
DBienes	edificio	Edificio
DBienes	nivel	Nivel donde se encuentra el Bien (en el piso 1 o en el 2)
DBienes	local	Local, o lugar exacto donde se encuentra
DBienes	departamento	Departamento (si pertenece al departamento de Administración, Informática, etc.)
DBienes	personaDesig	Persona a quien se le asignó
DBienes	fechaAsig	Fecha en que se asignó
DBienes	antigüedad	Antigüedad del bien. (Años desde que se asignó)
DBienes	valor	Costo
DBienes	detalles	Detalles u observaciones
DBajas	id_ben	ID de la tabla Bien, que servirá para identificar a cada Bien ar realizar una consulta de los mismos
DBajas	claveCABM	Clave CABM del bien
DBajas	descripcion	Descripción
DBajas	marca	Marca
DBajas	modelo	Modelo
DBajas	noSerie	No. de serie
DBajas	edificio	Edificio
DBajas	nivel	Nivel
DBajas	local	Local
DBajas	departamento	Departamento
DBajas	personaDesig	Persona responsable
DBajas	fechaBaja	Fecha en que se dio de baja
DBajas	antigüedad	Años desde que se dio de baja
DBajas	valor	Costo
DBajas	detalles	Detalles u observaciones.
DPregResp	pregunta	Pregunta secreta. Este view servirá para la recuperación de la contraseña, en caso de olvido o extravío de la misma
DPregResp	respuesta	Respuesta secreta

DPrgrRpasw	pregunta	Pregunta secreta, este view servirá para la verificación de la contraseña actual y la respuesta secreta en caso que se desee modificar el nombre de usuario y/o contraseña
DPrgrRpasw	respuesta	Respuesta secreta
DPrgrRpasw	contraseñaUsuario	Contraseña

9 EER Diagram



10 Algoritmos desarrollados

Los algoritmos se presentan en forma de pseudocódigo.

10.1 Correr aplicación

```
1. Usuarios = null;
2. tUsuarios = 0;
3. Obtener usuarios;
4. tUsuarios = ContarUsuarios;
5. If (tUsuarios > 1) Then
    Desplegar: "Existe más de un usuario registrado en el sistema"
Else if (tUsuarios < 1) Then
    Abrir ventana: CrearNvoUsuario
    Desplegar: "¿Primera vez en el sistema?, cree su cuenta de usuario"
Else
    Abrir ventana: IniciarSesion
End
```

10.2 Crear cuenta de usuario

```
1. EditableFalse campoContraseña
2. EditableFalse campoContraseña2
3. In-visibilizar botonVerificarContra
4. Inhabilitar campoPregunta1
5. Inhabilitar campoPregunta2
6. Inhabilitar campoPregunta3
7. Inhabilitar campoPregunta4
8. Inhabilitar campoRespuesta1
9. Inhabilitar campoRespuesta2
10. Inhabilitar campoRespuesta3
11. Inhabilitar campoRespuesta4
12. In-visibilizar botonGuardar
13. In-visibilizar botonModDatosUsuario
14. nomUsuario = null;
15. contra = null;
16. contra2 = null;
17. pregunta1 = null;
18. pregunta2 = null;
19. pregunta3 = null;
20. pregunta4 = null;
```

```

21.respuesta1 = null;
22.respuesta2 = null;
23.respuesta3 = null;
24.respuesta4 = null;
25.Introducir nomUsuario
26.EditableTrue campoContraseña
27.Introducir contraseña
28.EditableTrue campoContraseña2
29.Introducir contraseña2
30.Visibilizar botonVerificarContra
31.Presionar botónVerificarContra
32.If (contra2 != contra) Then
    Desplegar: "Las contraseñas no coinciden"
Else
    Desplegar: "Todos los datos son correctos"
    Habilitar campoPregunta1
    In-visibilizar botonVerificarContra
    Visibilizar botonModDatosUsuario
    Introducir pregunta1
    Habilitar campoRespuesta1
    Introducir respuesta1
    Habilitar campoPregunta2
    Introducir pregunta2
    Habilitar campoRespuesta2
    Introducir respuesta2
    Habilitar campoPregunta3
    Introducir pregunta3
    Habilitar campoRespuesta3
    Introducir respuesta3
    Habilitar campoPregunta4
    Introducir pregunta4
    Habilitar campoRespuesta4
    Introducir respuesta4
    Visibilizar botonGuardar
    Presionar botonGuardar
    Desplegar: "¿Desea continuar?"
    If (respuesta == Aceptar) Then
        RegistrarUsuario
        Abrir ventana: IniciarSesion
        Desplegar: "Se ha registrado exitosamente, inicie sesión"

```


10.3 Iniciar sesión

```
1. username = null
2. Password = null
3. Vusername = false
4. Vpassword = false
5. Introducir username
6. Vusername = VerificarUsername
   // true significa que es correcto
7. If (Vusername == true) Then
   Inhabilitar campoNomUsuario
   Introducir password
   Vpassword = VerificarPassword
   If (password == true) Then
     Abrir ventana: AdministracionBienes
     Desplegar: "Bienvenido"
   Else
     Desplegar: "Su contraseña es incorrecta"
   Else
     Introducir password
     Presionar botonEntrar
     Desplegar: "El nombre de usuario y/o contraseña son incorrectos"
End
```

10.4 Registrar nuevo bien

```
1. Descripción = null
2. marca = null
3. modelo = null
4. claveCABM = null
5. noSerie = null
6. persona = null;
7. edificio = "CEC Morelia";
8. departamento = null;
9. año = null;
10. mes = 0;
11. dia = 0;
12. detalles = null
13. valor = 0.0
14. nivel = 0
15. local = 0
16. fechaAsignacion = null
17. Vregistro = false;
18. Introducir descripcion
19. Introducir marca
20. Introducir modelo
```

```

21. Introducir claveCABM
22. Introducir noSerie
23. Introducir persona
24. Introducir departamento
25. Introducir año
26. Introducir mes
27. Introducir día
28. Introducir detalles
29. Introducir valor
30. Introducir nivel
31. Introducir local
32. If (descripción==null || marca==null || modelo==null || noSerie==null
    || persona==null || edificio==null || departamento==null || año==null
    || mes==0 || día==0 || detalles==null || valor==0.0 || nivel==0 ||
    local==0) Then
    Desplegar: "Todos los campos son obligatorios, excepto la clave
    CABM"
Else
    tAño=0
    tAño = ConvertirAentero(año)
    if (tAño < 1996) Then
        Desplegar: "El CEC se fundó en 1996, no puede introducir un año
        inferior a éste"
    Else
        VnoSerie = false
        VclaveCABM = false
        VnoSerie = ComprobarExistencia
        // true significa que sí está registrado
        If (VnoSerie == true) Then
            Desplegar: "El número de serie introducido ya está registrado"
        Else
            VclaveCABM = ComprobarExistencia
            If (VclaveCABM == true) Then
                Desplegar: "La clave CABM ya se encuentra registrada"
            Else
                fechaAsignacion = año + "-" + mes + "-" + día
                Vregistro = RegistrarBien(descripción, marca, modelo,
                    claveCABM, noSerie, valor, persona, edificio, departamento,
                    nivel, local, fechaAsignacion, detalles)
                // true significa que sí se guardaron los datos correctamente
                If (vRegistro == true) Then
                    Desplegar: "El bien se ha registrado exitosamente"
                    Desplegar: Datos del registro en tabla
                Else
                    Desplegar: "Los datos no se pudieron almacenar"
            End
        End
    End
End

```

10.5 Búsqueda avanzada

```

1. Descripción = null
2. marca = null
3. modelo = null
4. claveCABM = null
5. noSerie = null
6. persona = null;
7. edificio = "CEC Morelia";
8. departamento = null;
9. año = null;
10. mes = 0;
11. día = 0;
12. detalles = null
13. valor = 0.0
14. nivel = 0
15. local = 0
16. Vregistro = false;
17. Introducir datosNecesarios
    // Si continua siendo nulo un campo significa que no se introdujo nada
    en su campo correspondiente
18. If (descripción==null && marca=null && modelo==null && claveCABM==null
    && noSerie==null && persona==null && edificio!=null &&
    departamento==null && año==null && mes==0 && día==0 && detalles==null
    && valor==0.0 && nivel==0 && local==0) Then
    Desplegar: InformeCompletoDeBienes en tabla
    Desplegar: "¿Desea generar un inventario en Excel?"
    If(respuesta==Aceptar) Then
        GenerarInventarioGeneral
Else
    If (año!=null && día!=0 || mes!=0) Then
        Desplegar: "Si desea consultar por fecha introduzca sólo el año"
    Else
        // Para determinar qué campos fueron llenados y por
        coinsiguiente, qué es lo que el usuario espera que se
        despliegue, se utilizarán instrucciones de tipo else if()
        // no continuarán buscando una vez que encuentren la
        combinación correcta de campos llenados
        // lo cual optimiza al código
        // Acontinuación se mostrarán las posibles combinaciones que
        sí se podrían presentar (previstas)
        tipoConsulta = null;
        if (descripción!=null && marca!=null && modelo!=null &&
        claveCABM==null && noSerie==null && persona!=null && edificio!=null
        && departamento!=null && año!=null && mes!=0 && día!=0 &&
        detalles!=null && valor!=0.0 && nivel!=0 && local!=0) Then
            // Combinación de campos llenados
            tipoConsulta = "TodosLosCamposExptoNoSerieyClaveCABM"

```

```

Else if(descripcion ==null && marca==null && modelo==null &&
claveCABM==" " || claveCABM=="vacia" || claveCABM=="vacio" ||
claveCABM=="sin clave" || claveCABM=="sin clave cabm" ||
claveCABM=="VACIA" || claveCABM=="VACIO" || claveCABM=="SIN CLAVE"
|| claveCABM=="SIN CLAVE CABM" || claveCABM=="sin clave CABM" &&
noSerie==null && persona==null && edificio!=null &&
departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0)
    tipoConsulta = "BienesSinClaveCABM"

Else if (descripcion ==null && marca==null && modelo==null &&
claveCABM!=null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "claveCABM"

Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion"

Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca"

Else if (descripcion==null && marca==null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "modelo"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie!=null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "noSerie"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "nivel"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null

```

```

&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "local"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "personaAsignada"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "departamento"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "valor"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "detalles"

Else if (descripcion==null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año!=null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "fechaAsignacion"

Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripción,marca"

Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripción,marca,modelo"

Else if (descripcion==null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null

```

```

&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,modelo,departamento"

Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripción,departamento"

Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripción,personaAsignada"

Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripción,marca,modelo,persona,departamento"

Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripción,persona,departamento"

Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripción,nivel"

Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripción,marca,modelo,departamento"

Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta =
"descripción,marca,modelo,departamento,detalles"

```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,persona,marca"
```

```
Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,detalles,persona"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,departamento"
```

```
Else if (descripcion!=null && marca==null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento!=null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,modelo,departamento"
```

```
Else if (descripcion!=null && marca==null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,modelo"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,valor"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,valor"
```

```
Else if (descripcion ==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,detalles"
```

```
Else if (descripcion==null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,modelo"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento!=null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,departamento"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles!=null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,valor,detalles"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "marca,nivel"
```

```
Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "descripcion,local"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,persona"
```

```
Else if (descripcion==null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,modelo,persona"
```

```
Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,valor"
```



```
Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,valor"
```

```
Else if (descripcion ==null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,modelo,valor"
```

```
Else if (descripcion==null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,modelo,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,detalles"
```

```
Else if (descripcion==null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "marca,modelo,valor,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelovalor,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "descripcion,marca,modelo,local"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles!=null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "descripcion,marca,modelo,local,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "descripcion,marca,local"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles!=null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "descripcion,marca,local,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles!=null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,nivel,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,nivel"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles!=null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,nivel,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && día==0 &&
detalles==null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,nivel"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,valor,nivel"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,valor,nivel"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor!=0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "marca,nivel,detalles,valor"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "marca,nivel,detalles"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "marca,local,valor"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor!=0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "marca,local,detalles,valor"
```

```
Else if (descripcion==null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "marca,nivel,valor"
```

```
Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año!=null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,persona,fechaAsignacion"
```

```
Else if (descripcion!=null && marca!=null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor!=0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,nivel,detalles,valor"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor!=0.0 && nivel!=0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,nivel,valor,detalles"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor!=0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,valor,persona"
```

```
Else if (descripcion!=null && marca==null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,modelo,detalles"
```

```
Else if (descripcion!=null && marca==null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "descripcion,modelo,local"
```

```
Else if (descripcion!=null && marca==null && modelo!=null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles!=null && valor==0.0 && nivel==0 && local!=0) Then
    tipoConsulta = "descripcion,modelo,local,detalles"
```

```
Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel!=0 && local!=0) Then
    tipoConsulta = "descripcion,nivel,local"
```

```
Else if (descripcion!=null && marca!=null && modelo!=null &&
claveCABM==null && noSerie==null && persona!=null && edificio!=null
&& departamento==null && año==null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,marca,modelo,persona"
```

```

Else if (descripcion!=null && marca==null && modelo==null &&
claveCABM==null && noSerie==null && persona==null && edificio!=null
&& departamento==null && año!=null && mes==0 && dia==0 &&
detalles==null && valor==0.0 && nivel==0 && local==0) Then
    tipoConsulta = "descripcion,fecha"

Else
    tipoConsulta = "combinacionInesperada"

if (tipoConsulta == "combinacionInesperada") Then
    Desplegar: "Su búsqueda no puede ser realizada, contáctese al
desarrollador de la aplicación"
Else
    Desplegar: tipoConsulta en tabla
    Desplegar: "¿Desea generar un inventario?"
    If (respuesta == Aceptar) Then
        // Se genera un inventario con los datos solicitados
        GenerarInventarioExcel.(tipoConsulta)

End

```

10.6 Actualización de datos

1. Inhabilitar botonActualizar
2. Descripción = null
3. marca = null
4. modelo = null
5. claveCABM = null
6. noSerie = null
7. persona = null;
8. edificio = "CEC Morelia";
9. departamento = null;
10. año = null;
11. mes = 0;
12. dia = 0;
13. detalles = null
14. valor = 0.0
15. nivel = 0
16. local = 0
17. fechaAsignacion = null
18. Vactualizacion = false
19. IDbien = 0
20. Seleccionar fila
21. IDbien = getIDbien
22. Habilitar BotonActualizar
23. Introducir descripcion
24. Introducir marca

```

25. Introducir modelo
26. Introducir claveCABM
27. Introducir noSerie
28. Introducir persona
29. Introducir departamento
30. Introducir año
31. Introducir mes
32. Introducir día
33. Introducir detalles
34. Introducir valor
35. Introducir nivel
36. Introducir local
37. Editar campos
38. fechaAsignacion = año + "-" + mes + "-" + día
39. Presionar botonActualizar
40. If (descripcion==null || marca==null || modelo==null || noSerie==null
    || persona==null || edificio==null || departamento==null || año==null
    || mes==0 || día==0 || detalles==null || valor==0.0 || nivel==0 ||
    local==0) Then
    Desplegar: "Todos los campos son obligatorios, excepto la clave
    CABM"
Else
    // El IDbien sirve para modificar únicamente los datos del registro
    seleccionado
    Vactualizacion=ActualizarBien(IDbien,descripcion,marca,modelo,clave
    CABM,noSerie,valor,persona,edificio,departamento,nivel,local,fechaA
    signacion,detalles)
    If (Vactualizacion == true) Then
        Desplegar: "El registro de ha actualizado correctamente"
        Desplegar: DatosActualizados en tabla
        Inhabilitar botonActualizar
    Else
        Desplegar: "Los datos no se pudieron actualizar"

End

```

10.7 Bajas

```

1. Inhabilitar botonBaja
2. Descripción = null
3. marca = null
4. modelo = null
5. claveCABM = null
6. noSerie = null
7. persona = null;
8. edificio = "CEC Morelia";
9. departamento = null;

```

```

10.año = null;
11.mes = 0;
12.día = 0;
13.detalles = null
14.valor = 0.0
15.nivel = 0
16.local = 0
17.fechaAsignacion = null
18.Vbaja = false
19.IDbien = 0
20.Seleccionar fila
21.IDbien = getIDbien
22.Habilitar BotonBaja
23.Presionar BotonJaba
24.Desplegar: "¿Desea dar de baja al bien?"
    If (respuesta == Aceptar) Then
        // Se lleva al método correspondiente pasándole como parámetro el
        ID del registro para únicamente "afectar" al seleccionado
        Vbaja = DarDeBaja.(IDbien)
        If (Vbaja == true) Then
            Desplegar: "El registro se ha dado de baja exitosamente"
            Desplegar: BienesDadosDeBaja en tabla
        Else
            Desplegar: "El registro no se pudo dar de baja"

End

```

10.8 Recuperación de contraseña

1. Inhabilitar botonVerificarRespuestas
2. Pregunta1 = obtenerPregunta1
3. Pregunta2 = obtenerPregunta2
4. Pregunta3 = obtenerPregunta3
5. pregunta4 = obtenerPregunta4
6. respuesta1 = null
7. Respuesta2 = null
8. Respuesta3 = null
9. Respuesta4 = null
- 10.Desplegar: pregunta1
- 11.Desplegar: pregunta2
- 12.Desplegar: pregunta3

```
13.Desplegar: pregunta4
    // Verificar si la respuesta es correcta
14.vRespuesta1 = false
15.vRespuesta2 = false
16.vRespuesta3 = false
17.vRespuesta4 = false
18.Introducir respuesta1
19.Pregunta1 = getPregunta1
20.vRespuesta1 = Verificar(pregunta1,respuesta1)
    if(vRespuesta1 == true) Then
        Inhabilitar campoRespuesta1
        Desplegar: "Respuesta correcta"
21.Introducir respuesta2
22.Pregunta2 = getPregunta2
23.vRespuesta2 = Verificar(pregunta2,respuesta2)
    if(vRespuesta2 == true) Then
        Inhabilitar campoRespuesta2
        Desplegar: "Respuesta correcta"
24.Introducir respuesta3
25.Pregunta3 = getPregunta3
26.vRespuesta3 = Verificar(pregunta3,respuesta3)
    if(vRespuesta3 == true) Then
        Inhabilitar campoRespuesta3
        Desplegar: "Respuesta correcta"
27.Introducir respuesta4
28.Pregunta4 = getPregunta4
29.vRespuesta4 = Verificar(pregunta4,respuesta4)
    if(vRespuesta4 == true) Then
        Inhabilitar campoRespuesta4
        Desplegar: "Respuesta correcta"
30. if(vRespuesta1==true && vRespuesta2==true && vRespuesta3==true &&
    vRespuesta4==true) Then
```


// Si todas las respuestas son correctas, automáticamente nos permitirá la entrada

```

In-visibilizar pregunta1
In-visibilizar pregunta2
In-visibilizar pregunta3
In-visibilizar pregunta4
In-visibilizar campoRespuesta1
In-visibilizar campoRespuesta2
In-visibilizar campoRespuesta3
In-visibilizar campoRespuesta4
In-visibilizar botonVerificarRespuestas
Visibilizar campoContraseña
Visibilizar campoContraseña2
Visibilizar botonActualizarContraseña
Desplegar: "Bienvenido usuario"
contraseña = null
contraseña2 = null
Introducir contraseña
Introducir contraseña2
If (contraseña2 != contraseña) Then
    Desplegar: "Las contraseñas no coinciden"
Else
    Desplegar "¿Desea modificar su contraseña?"
    If (respuesta == Aceptar) Then
        ActualizarContraseña(contraseña)
        Abrir ventana: IniciarSesion
        Desplegar: "Su contraseña se ha modificado
        correctamente"
// alguna de las respuestas fue incorrecta
Else
    If (respuesta1!=null && respuesta2!=null && respuesta3!=null &&
    respuesta4!="") &&) Then
        Habilitar botonVerificar

```

```
Presionar botonVerificar
If (vRespuesta1 != true) Then
    Desplegar: Respuesta incorrecta
Else
    Desplegar: Respuesta correcta

If (vRespuesta2 != true) Then
    Desplegar: Respuesta incorrecta
Else
    Desplegar: Respuesta correcta

If (vRespuesta3 != true) Then
    Desplegar: Respuesta incorrecta
Else
    Desplegar: Respuesta correcta

If (vRespuesta4 != true) Then
    Desplegar: Respuesta incorrecta
Else
    Desplegar: Respuesta correcta

End
```

10.9 Modificación de nombre de usuario y/o contraseña

1. contraseñaActual = null
2. contraseñaActual2 = null
3. preguntaSecreta = null
4. preguntaSecreta = ObtenerPreguntaSecreta
5. respuestaSecreta = null
6. vContraseña = false
7. vRespuesta = false
8. Inhabilitar botonVerificarDatos

9. Desplegar: preguntaSecreta

10. Introducir contraseña

11. Introducir contraseña2

```

If (contraseña2 == contraseña) Then
    vContraseña = VerificarPassw(contraseña)
    if (vContraseña == true) Then
        Inhabilitar campoContraseña
        Inhabilitar campoContraseña2
        Desplegar: "Contraseña correcta"
        Introducir RespuestaSecreta
        vRespuesta = VerificarRespuesta(respuestaSecreta)
        If (vRespuesta == true) then
            // Si todos los datos son correctos, entramos
            automáticamente
                In-visibilizar preguntaSecreta
                In-visibilizar campoRespuestaSecreta
                In-visibilizar botonVerificar
                nomUsuario = null
                nomUsuario = ObtenerNomUsuario
                nuevaCaontraseña = null
                nuevaContraseña2 = null
                Desplegar: nomUsuario en campoNomUsuario
                Desplegar: "Nueva contraseña:"
                Visibilizar botonActualizar
                Inhabilitar botonActualizar
                Editar nomUsuario
                Introducir nuevaContraseña
                Introducir nuevaContraseña2
                If (nomUsuario!=null && nuevaContraseña!=null &&
nuevaContraseña2!=null) Then
                    Habilitar botonActualizar
                    Presionar botonActualizar
                    If (contraseña2 != contraseña) Then

```

```
Desplegar: "¿Sus nuevos datos son  
correctos?"  
  
If (Respuesta == Aceptar) Then  
    ModificarDatos(nomUsuario,contraseña)  
    Abrir ventana: IniciarSesion  
    Desplegar: "Sus datos se actualizaron  
correctamente, inicie sesión de nuevo"  
  
End
```